

I D E N T I F I C A T I O N

SEQ 0001

PRODUCT CODE: MAINDEC-11-DOKUB-B-D
 VERSION /101/

PRODUCT NAME: KD11-K Microdiagnostic

MAINTAINER: Diagnostic Engineering

AUTHOR: Don North

DATE CREATED: 18-Jan-1977

LAST REVISION: 22-Jun-1977

COPYRIGHT (C) 1977; Digital Equipment Corporation
 146 Main Street
 Maynard, Massachusetts, USA
 01754 617-897-5111

This software is furnished to the purchaser under a license for use on a single computer system, and can be copied (with inclusion of DIGITAL's copyright notice) only for use in such system, except as may otherwise be provided in writing by DIGITAL.

The information in this document is subject to change without notice, and should not be construed as a commitment by DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION assumes no responsibility for any errors that may appear in this document.

DIGITAL assumes no responsibility for the use or reliability of its software on equipment not supplied by DIGITAL.

- TABLE OF CONTENTS -

1.0	ABSTRACT
2.0	REQUIREMENTS
2.1	Hardware
2.2	Documentation / Listings
3.0	LOADING PROCEDURE
4.0	STARTING PROCEDURE
4.1	Operator's Console
4.2	DCS Switches
4.3	Console "INIT"
5.0	OPERATING PROCEDURE
5.1	DCS Indications While Executing
5.2	DCS Execution Time
5.3	DCS 'End of Pass' and 'Error' Indications
6.0	RESTRICTIONS
6.1	Hardware
6.2	Software
7.0	TEST DESCRIPTION
7.1	Test Structure
7.2	DCS Microcode Conventions
7.3	What is/is-not tested
7.4	Global Test Order
8.0	ERROR HANDLING
8.1	What is an 'ERROR' ?
8.2	'FAULT DIRECTORY' Format and Use
8.3	'SCOPE Loop' Facility
9.0	REVISION HISTORY
9.1	Revision Number
9.2	Revisions to DCS Code
10.0	DCS VERIFICATION / SELF TEST
10.1	Requirements
10.2	Method / Algorithm
10.3	Procedure and Indications
11.0	MISCELLANEOUS
11.1	ACT/APT/XXDP
11.2	Macro Instruction Interface

1.0 ABSTRACT

The 'DCS' (Diagnostic Control Store) Module is a diagnostic tool specifically designed for the PDP-11/60 [KD11-K] Central Processor. Functioning as an alternate 2048 word control store, microcoded tests are executed to detect and isolate errors within the internals of the processor control and datapath hardware. Error indication information is provided by the DCS module; coupled with an indexed FAULT DIRECTORY, errors are resolved to the module level, when possible. Additional information is also provided resolving the error to a specific functional logic block. Significant benefits of this micro diagnostic approach are seen to be:

- Memory/I-O Device/UNIBUS independence
- Direct hardware microcontrol and visibility
- Extremely fast load and execution times
- Excellent coverage and resolution

2.0 REQUIREMENTS

A Preliminary note:

Throughout this DCS User's Guide, the two terms "FAULT" and "ERROR" have been used interchangeably. They both are to indicate "malfunctioning logic elements" (eg, busted IC's) and open/shorted ETCH runs, etc., in the unit under test. No distinction between the two terms is implied.

2.1 Hardware

To use the DCS, the following hardware is required:

1. DCS (Diagnostic Control Store) Module M7871 (KU116-BB)
2. PDP-11/60 [KD11-K] Central Processor
3. DL11-W Line Clock (required)
4. First 4k memory bank (required minimum)

2.2 Documentation / Listings

The available documentation for the DCS module user comprises the following items:

- DCS User's Guide (this document)
- FAULT DIRECTORY Listing; for module replacement information
- DCS Microcode Listing; when IC level debug is necessary
- DCS Maintenance Manual, Print Set; for detailed information on DCS hardware operation
- KD11-K Processor Maintenance Manual, Print Set; for IC level debug, base machine hardware specific information

Specific MAINDEC component part numbers for the DCS documentation are as follows:

- MD-11-DQKUB--D;
User's Guide, FAULT DIRECTORY (PAPER, 80 pages)
- MD-11-DQKUB--LA;
DCS Microcode Listing (PAPER, 450 pages)
- MD-11-DQKUB--FA;
User's Guide, FAULT DIRECTORY, DCS Microcode Listing (FICHE, 4 cards)

3.0 LOADING PROCEDURE

The DCS occupies slot 1 in the KD11-K processor backplane; thus if an ECS or UCS option is present, it must be removed. To load the DCS, the following sequence should be employed:

1. Power down the CPU
2. Remove the ECS/UCS option from processor slot #1, if present
3. Insert the DCS module into slot #1. Use caution while inserting the DCS module, as a slightly bowed board may require some gentle maneuvering to seat it in place.
4. Orient the DCS 'RUN/STOP' switch to 'STOP', and the 'NORMAL/VERIFY' switch to 'NORMAL'
5. Now power up the CPU

4.0 STARTING PROCEDURE

DCS execution can be initiated by two distinct methods:

4.1 Startup via the Operator's Console (KY11-P)

With the KD11-K CPU in "CONSOLE" ("HALTe4") mode, simultaneously depress the KY11-P operator's console "CNTRL/DIAG" keys to start the DCS. (If no DCS/ECS/UCS happens to be present, there should be no effect). Section 5.1 (below) interprets the display on the operator's console while the DCS is executing, and after it stops. If this method fails to start the DCS, proceed to the next paragraph.

4.2 Startup via the DCS Switches

If it is desired for some reason to bypass use of the KY11-P operator's console to initiate the DCS, an alternative means is provided. This method would be used, for example, in the event that starting from the operator's console was not possible (using "CNTRL/DIAG") due to some hardware malfunction. The procedure is:

1. On the DCS module, set the 'NORMAL/VERIFY' switch to 'NORMAL'
2. Now set the 'RUN/STOP' switch (on DCS module also) to the 'RUN' position, or flip it from 'RUN' -> 'STOP' -> 'RUN'

The DCS should now assume control of the KD11-K CPU, irregardless of the previous state of the CPU ("CONSOLE", "RUN", or whatever).

NOTE

If the DCS 'RUN/STOP' switch was already in the 'RUN' position, then it is expected that the operator's console keys have no effect - the DCS is already enabled to execute, and is controlling the CPU. The DCS microcode does not monitor the console keypad for operator input.

If neither of the above methods produce a 'RUNNING' indication of the DCS (as per Section 5.1), then proceed to the next paragraph.

4.3 Using Console "INIT"

Set the DCS switches as detailed in Section 4.2 (ie, 'NORMAL' and 'RUN'). Now generate a processor "INIT" signal by simultaneously depressing the "HALT" and "START" keys on the operator's console. This should now initialize the CPU logic and restart the DCS, producing a 'RUNNING' indication (see Section 5.1).

If no 'RUNNING' indication is now present, then a problem exists in the power supply, system clocks, DCS module, or ???.

5.0 OPERATING PROCEDURE

5.1 DCS Indications While Executing

5.1.1 On the KD11-K Operator's Console

While the DCS is executing, the operator's console display should be approximately as follows:

"RUN"	-	ON Continuously
"PROC"	-	BLINKing
"USER"	-	BLINKing
"CONSOLE"	-	OFF Continuously
"BATTERY"	-	<indeterminate>

The 6 digit octal display should read:

(0 0 0 0 0 0) - In DCS pass 1;

(.0.0.0.0.0.0.) - In DCS passes 2-64.

If any of the above conditions are not met, then either:

1. The console hardware is inoperable, or
2. The DCS is hung in an 'error/scope loop'

(Not necessarily in that order)

5.1.2 On the DCS Module

On the DCS module, the two important indicators to watch at this time are the 'ERROR' and 'EOP' LEDs:

-----LEDS-----			
'ERROR'	'EOP'	State	Comment
-----	-----	-----	-----
OFF	OFF	Probable ERROR	DCS and/or processor HUNG
OFF	ON	* EOP *	Successful EOP
ON	OFF	* ERROR *	Genuine ERROR
ON	ON	Probable ERROR	DCS 'VERIFY MODE' indication

Note the 'Probable ---' notation. Either 'EOP' or 'ERROR' on the DCS should be lit; both OFF or both ON indicate non-standard conditions that require further investigation. Neither 'ERROR' nor 'EOP' indicates that the DCS and/or processor are in a HUNG condition (eg, clocks suppressed); see section 8 (Error Handling). Both 'ERROR' and 'EOP' is the standard DCS 'VERIFY MODE' indication. Make sure the DCS 'NORMAL/VERIFY' switch is set to 'NORMAL' (if this is intended), and then restart the DCS. See section 10 (Verify Mode) for further information.

5.2 DCS Execution Time

The execution time of the DCS will vary depending upon its mode of initiation.

5.2.1 "CNTRL/DIAG" Start From Operator's Console

When started via "CNTRL/DIAG" from the operator's console, and assuming no errors, DCS will execute with console display as detailed above for approximately 6 seconds (64 passes at 100 milliseconds/pass, about 350,000 microcycles/pass). After this time has elapsed, control should return to the KD11-K microcode, as if the CPU had just been powered up with the glide switch set to "HALT". Note that the machine will end up in console mode irregardless of the actual glide switch settings (RUN/BOOT/HALT). Section 5.3 interprets the console and DCS displays after this delay time, for successful 'END-OF-PASS' and 'ERROR' indications.

5.2.2 Start From DCS Switches

When started via setting 'RUN/STOP' to 'RUN' on the DCS module, the DCS will execute continuously, not returning to the KD11-K microcode until the 'RUN/STOP' switch is reset to the 'STOP' position. Throughout this time, the execution indications will be as previously detailed. When the switch is returned to 'STOP', the DCS may execute for a maximum of

approximately 6 more seconds; and then proceed as described directly above, entering "CONSOLE" mode.

5.3 DCS 'END-OF-PASS' and 'ERROR' Indications

5.3.1 On The DCS Module

Successful 'END-OF-PASS' and 'ERROR' conditions detected by the DCS are indicated most directly on the DCS module by the two LEDs labeled 'EOP' and 'ERROR'. 'EOP' is turned on at the end of each successful pass thru the DCS code - assuming 'ERROR' has not yet been turned on. In an error-free running situation, 'EOP' will be on; 'ERROR' off. Note that 'EOP' comes on at the end of the first pass (assuming no errors), and thus will appear to be on continuously.

In the event the DCS detects an error, the 'ERROR' LED will be turned on, and the 'EOP' LED turned off. The 'ERROR' LED will be latched in such a way that it cannot be turned off if the error disappears (i.e., a 'flaky' timing error), thus retaining the indication of an error. See section 8 (Error Handling) for a full treatment of the various error conditions: Detection, Indication, Scope-looping.

5.3.2 On the Operator's Console

Successful 'END-OF-PASS' and 'ERROR' conditions are indicated on the operator's console as follows:

1. A successful 64 passes and return to KD11-K processor microcode control, by a

(.1,2,3,3,2,1.)

in the console display, and the KD11-K processor "HALTed" in "CONSOLE" mode.

2. An error in pass 1 will usually (but not always) be indicated by a

(0 0 0 0 0 0)

in the console display.

3. An error in pass 2 thru -- will usually (but not always) be indicated by a

(.0,0,0,0,0,0.)

in the console display.

Note the qualifications in the previous statements:

For a successful 'END-OF-PASS' and exit back to KD11-K microcode control, only one display is valid:

(.1.2.3.3.2.1.)

For any other display except "(.1.2.3.3.2.1.)", check the DCS module 'EOP' and 'ERROR' LEDs for the most reliable indication of the result of execution.

See Section 8 for a full explanation of error processing.

6.0 RESTRICTIONS

6.1 Hardware

6.1.1 Cache and Memory Management (KT)

The DCS executes with both Cache and KT disabled:

1. Cache, by setting both the "Force Miss" bits
2. KT, by clearing the "Enable" bit

The DCS checks the most basic path from the UNIBUS to/from internal data paths. Further macro diagnostic programs are available for Cache (MD-11-DQKKA-*) and Memory Management (MD-11-DQKTA-*) fault diagnosis.

6.1.2 MOS Memory Battery Backup

The MOS memory battery backup (if present) must either be "Good", or else disabled. Otherwise, the micro diagnostic code WILL NOT execute without detecting an error.

Indications are: ERCD=5621/TNUA=7400, in TEST620C.

6.2 Software

6.2.1 Return To Console / DCS End-of-pass Processing

Console "HALT" mode is the ONLY exit provided from DCS back to base machine microcontrol. This action is due to:

- 1) The DCS, upon detecting an error, locks itself (and the processor) up in such a way that manual intervention by the operator is required for return to base machine microcontrol.

- 2) The DCS completely alters the internal microstate of the processor, destroying its previous contents, and leaving "garbage" in its place. Thus the full base machine power-up "INIT" sequence is generated by DCS to "clean-up" the processor prior to returning control.

7.0 TEST DESCRIPTION

7.1 Test Structure

7.1.1 Philosophy

The testing philosophy used in the design of the DCS microcode centers around two major points:

1. Start with as minimal a HARDCORE as possible.
2. Use only TESTED and VERIFIED logic elements to diagnose UNTESTED elements; add TESTED elements to the arsenal of logic available for further testing.

This method of test organization and construction presents the best approach for building a high resolution and high coverage diagnostic program. Section 7.4 (below) summarizes the actual testing order present in the DCS microcode, as designed using the above philosophy. Section 8.1 presents more specific details on the actual method of ERROR detection employed by the DCS microcode and hardware.

7.1.2 Mechanics

The manner in which DCS tests are setup is depicted in the following diagram:

```

      \ /
      .
<setup ENUA/DCS-CNTR as test requires>
      .
<perform the actual test function>
      .
      -may be inline microcode
      -may call subroutine
      .
<jump to BUT area, perform BUT(test) into TARGET>
      .
<enter TARGET area from BUT, ENUA::TNUA compare>
      -the DCS-CNTR was loaded above to enable compare here
      -the ENUA was setup at correct TARGET point
      -exit via BUTA(RETURN)
      .
      \ /
<...next test now begins...>
```

7.2 DCS Microcode Conventions

7.2.1 Microdiagnostic Code Listing Organization

The format of the DCS microcode listing is presented in the following table, along with a brief description of each section (when necessary):

pages	contents
1-5	IDENTIFICATION, TABLE-OF-CONTENTS
6-9	DCS REGISTER LAYOUTS, MICROWORD BIT TABLE a number of graphic tables
10-24	MICROWORD FIELD DEFINITIONS, BASE MACHINE defines U<47:00> and their functions
25-26	DCS FIELD DEFINITIONS defines U<54:48> and their functions
27-37	"SIMPLE" MACROS these macros are combinations of FIELD definitions only
38-52	"ADVANCED" MACROS these macros are combinations of SIMPLE macros
53-380	---DCS-MICRODIAGNOSTIC-TESTS---
381-384	EOP/VERIFY-MODE MICROCODE microcode used in END-OF-PASS and VERIFY-MODE processing
385-393	GENERAL SUBROUTINES, COMMON CODE this section contains some VERY commonly used subroutines
394-395	JAMUPP MICROCODE all JAMUPP conditions enter here ...
396-402	BUT(---) TAKEOFF MICROWORDS most all DCS tests start their microbranch from this list of BUT's
403-417	BUT(---) TARGET MICROWORDS and most all DCS tests end up here, where they compare their ENUA;ITNUA in this 256. word table
418	BIT MAP OF DCS ADDRESS SPACE

- BIT MAP of the entire DCS address space, 1=USED, 0=FREE
- X1-X10 SYMBOL/LINE-NUMBER/LOCATION CROSS REFERENCE
a very useful reference to find the location in the DCS listing of a particular symbolic label
- X11-X18 LOCATION/SYMBOL CROSS REFERENCE
an expansion of the above BIT MAP, substituting the symbolic label for the I/O's present in the above
- X19 FREE/USED LOCATION SUMMARY
a quantitative summary of FREE/USED microlocations, by PAGE and TOTAL

7.3 What is/is-not tested

The DCS micro diagnostic code has been designed to detect and isolate errors within the "internals" of the KD11-K processor hardware. As such, it does not attempt (nor is it possible) to detect certain errors, which are processor related, but require devices external to the processor. Errors related to NPR/BR arbitration sequences, multiple BR priority level interrupt sequences, POWER FAIL / RESTART, etc., are a few of the elements of this general class. The following sections enumerate both classes of logic: TESTED by DCS, and UNTESTABLE by DCS.

7.3.1 TESTED by DCS

Essentially, the DCS is designed to test the "heart" of the KD11-K processor - those elements that must be functional to "bring up" the processor to a level whereupon further macro diagnostic programs could then be loaded and started (ie, M9301-YH boot/diagnostic, processor and peripheral diagnostics, etc.) to successfully isolate more complex processor and system related errors. Processor operation with either KT/KJ or CACHE enabled was not considered as part of this "heart" - as the processor will run perfectly fine without either of these facilities (albeit at a degraded performance level). Thus the DCS is able to concentrate on more thorough coverage and resolution in those portions of the hardware that are least "visible and testable" from a macro diagnostic, and very suitable to micro diagnosis.

Rather than present a full module by module list of which logic the DCS tests, refer to section 7.4 (below), which is an itemized "execution-time" summary of the DCS micro diagnostic tests. The following section lists those areas that the DCS cannot test (ie, uncontrollable or unobservable logic) or

would have required a prohibitive number of microwords to test effectively.

7.3.2 UNTESTABLE by DCS

From a functional point of view, the DCS operates on the KD11-K processor from the "inside out". Both the CACHE and KT/KJ are turned off (disabled); there is no FP11-E HFP unit assumed to be present; likewise no ECS/WCS options. No external I/O options are assumed on the UNIBUS, except the standard DL11-W console interface / line clock. The lowest 4K memory bank is assumed to be present. These restrictions impose constraints on the logic that can be exercised by DCS. The following list attempts to detail as specifically as possible, on a module by module basis, those functional areas of the processor that the DCS cannot fully diagnose:

- K2 UWORD-
 - full effects of processor "INIT"
 - NUA<11> (DCS can't "see" it)
 - CROM contents and address drivers
 - FP11-E related (FLPGO, HFP(CC), etc.)
 - UCON's HFP/KT/WCS
 - HFP FLAG-ROM contents
- K3 IR-DECODE-
 - CROM contents and address drivers
 - CROM extension roms
 - full effects of processor "INIT"
 - full rom contents (location by location) off:
 - BYTE/CC, INSTR-5, FLTPT, CC(V/C), PS(CC)-BRANCH,
 - BYTE-DFCODE, KT-DECODE roms
 - KT/KJ enable/select logic
- K4 DATAPATH-
 - full processor "INIT" effects
 - data I/O validity in some A/B/C-SPAD locations
 - SP/BY/KT selection
 - CACHE INVALIDATE logic
- K5 CACHE/KT-
 - processor UNIBUS operation with CACHE enabled
 - processor UNIBUS operation with KT/KJ enabled
 - status conditions of above, including
 - RED/YELLOW ZONE, MM-ABORT, CACHE errors, etc.
- K6 TIMING-
 - ECL clock logic (must be "clocking" ...)
 - 74S37 etc CLOCK DRIVERS - mostly HARDWARE
 - MAINTENANCE clock
 - some JAMUPP/PULSE-SUPPRESS conditions (see K7)
 - UNIBUS master arbitration, NPR/BR/PROCESSOR
 - CACHE control (NPR track, HIT, etc.)
 - remember DCS turns CACHE OFF

- UNIBUS address drivers - full test of same
- full check of INTRNL=ADDR rom contents
- full processor "INIT" effects
- K7 STATUS-
 - full processor "INIT" generation/effects
 - POWER FAIL/RESTART logic
 - BR-4/5/7 requests, BG arbitration
 - SACK timeout (BG/NPR)
 - NPR/NPG logic
 - SERVICE conditions:
 - FP11-E/YELLOW-ZONE/CONSOLE/PWRFAIL/CACHE
 - JAMUPP conditions:
 - K7-ABORT/RED-ZONE/CACHE+MEMORY+WCS-parity-error
 - CONSOLE interface:
 - KEYCODE input/DISPLAY output/LOCAL and REMOTE
 - (see 11/60 microcode listing for "CONSOLE MICRO TEST", which tests these functions)
 - STATUS mux bits:
 - bits related to above conditions not asserted
 - BATTERY BACKUP for MOS MEMORY OK

7.4 Global Test Order

The following list provides a summary of the major functional blocks of the DCS micro diagnostic code, presented in execution-time order. Note the progression from the innermost portions of the processor logic (microsequencing, IR decode); through the intermediate areas (ALU, SHIFT TREE, etc.); out to the external interface logic (UNIBUS cycles, INTR sequences).

NOTE

Most of the capitalized terms refer to specific hardware elements in the KD11-K processor. No attempt is made to explain their meaning - the unfamiliarized reader is referred to the processor logic block diagrams, print set, maintenance manual, and the micro diagnostic code listing for their definition.

The notation "[xxx/yyy]" signifies the micro tests in the range "TESTxxx" to "TESTyyy". All DCS tests are numbered octally, and are executed in ascending order. Some test numbers are further broken down into "TESTxxxA", "TESTxxxB", etc., when their functions are logically similar. Certain test numbers are non-existent (eg, there are no tests with numbers in the range 200-277).

- [001/007] - NUA sequencing logic
- UPF sequencing, page changing

[010/011] - microsubroutine operation
-RETURN register, BUTA(SUBR-B)/BUTA(RFTURN) decode

[012A/050B] - IR decode logic and microbranch
-INSTP -1, -5, -FLTPT decode
-microbranch selection / execution
-misc IR decode related microbranch logic
-processor UCON IR load, EMIT constant generation

[101] - D-REGISTER/DBUF/BUSDIN/IR datapath
-D-REGISTER load, ALU control (zeroes)
-DBUF load, BUSDIN enable (UCON)

[102A/104B] - C-SCRATCHPAD
-address modes (2-bit/4-bit), address lines
-BUSDIN/CSP/ALU-B/D-REGISTER/DBUF/IR datapath, [1s/0s]

[105A/105E] - SR load/store
-SR load/store/XMUX-enable
-BUT(SR<3:0>) microbranch
-SR/ALU-A/D-REGISTER/DBUF/IR datapath, [1s/0s]

[114A/122A4] - ALU logic functions, D[C] sources
-ALU function/mode decode
-ALU logic function execution
-D[C] sources (ALUxx, CIN, save); D[C] microbranch

[130A1/136B2] - ALU arithmetic functions
-ALU arithmetic function/mode decode
-ALU arithmetic function execution
-ALU carry logic (in, out, lookahead)
-D[C] sources (COUTxx, CIN)

[320A/320F] - D[C] select logic, D=ZERO logic
-D[C] 1/8 addressing
-AUT(D=ZERO) decode logic

[350A/352D] - A/B-SCRATCHPADS
-addressing, lines and mode (SF, DF, RIF)
-data patterns [1s/0s]

[361A/371B] - SR/GUARD/XMUX, RES control
-SR shift (left, right, nop)
-GUARD register (shift, enable/disable, test)
-RES/SR control

-FLTPT assemble port

[372A/372B] - CUA(PROC mux) / BUTA(SUBR-A)
-BUTA(SUBR-A) decode/execution, RETURN register

[373A/373B] - JAMUPP and BUTA(DIAGNOSE)
-active BUTA(DIAGNOSE) decode
-JAMUPP clock suppress logic via external JAM

[374A1/376A] - A/B-SCRATCHPADS
-rewrite modes test (A/B, HI/LO, etc.)
-BYTE WRITE, DAD control
-R-IOR-1 / FLTPT-INHIBIT addressing

[410A/410E] - BYTE / BYTE-CONSTANT / D=ZERO (loop)
-BYTE/WORD rom decode, microbranch
-BYTE-CONSTANT CSP addressing, DAD control
-BUTR(D=ZERO) decode (full test)

[500C/500F] - PREFETCH / OVERLAP / SP-DEFEAT (loop)
-PREFETCH rom decode, microbranch
-OVERLAP rom decode
-SP-WRITE-DEFEAT decode/control

[503A/510F] - processor UCON registers / control
-FLAGS/EXFLAGS - read/write/microbranch
-FPS - read/write/microbranch
-PS - read/write/microbranch
-MULTIPLE BUT - input/select/output
-SERVICE/INTR decode logic, microbranch

[511A1/511B4] - MOVE FROM SAME STACK (MFSS) logic
-decode / microbranch

[512A1/512E2] - KT SRC/DST addressing logic for ASP/BSP
-rom decode / control

[520A/520E] - INSTR BRANCH rom
-IR/PS inputs, microbranch output

[533A/537] - SHIFT TREE (AMUX/BMUX/CMUX/SENDMUX)
-data path (is/os)
-function decode / mux select
-RES control / select
-COUNTER load / read

[551A/551C] - base machine COUNTER
-active BUTA(COUNT) decode/microbranch
-COUNTER count execution

[610A1/610D2] - PS condition code NZVC generation
-INSTP CLASS decode
-BYTE/WORD CC mux select
-CC rom addressing/data

[620A/624D] - microbreak and JAMUPP
-MICROBREAK REGISTER load/enable/compare
-JAMUPP via microbreak, JAM register & STATUS
-CUA TRACKING, lock/unlock
-JAMUPP inhibit micro-operation logic
-JAMUPP CLEAR

[701A/701D] - BA register

- 18. BIT load/read via STATUS mux
- microbranch conditions

[710A/722C] - UNIBUS function decode, error conditions

- ODD ADDR/INTERNAL ADDR/SSYN TIMEOUT errors
- 18./16. bit BA modes, I/O PAGE decode, CONSOLE mode
- SERVICE/JAM register inputs (STATUS mux)
- bus function decode (DATI, DATO, etc.)

[730A/731E] - UNIBUS cycles to/from memory

- DATI(B)(P)/DATO(B) execution, side effects
- UNIBUS data lines, control lines C0, C1
- ALLOW ODD ADDR, BYTE/WORD operations
- DBUF, UNIBUS data latches load/enable
- DATI-CLKIR decode/execution
- DATIP/PROC-BBSY/BUTA(LAST) logic
- clock suppress / restart logic

[740A/740D] - UNIBUS cycle function modification

- BUTA(INSTR-1)/PREFETCH alteration of BUS CODE
- bus cycle YANK (SP DEFEAT) decode

[761A/763D] - UNIBUS interrupt (BR INTR) logic

- bus reset, microbranch on status
- line clock INTR enable, at BR6
- PS PRIORITY level/INTR PRIORITY level interaction
- SERVICE port conditions
- ALLOW BUS GRANT / VECTOR LOAD logic

8.0 ERROR HANDLING

8.1 What is an 'ERROR' ?

The concept of an 'ERROR' in DCS terms is very simple. It involves the use of the ENUA (Expected NUA), TNUA (Tracking NUA), and DCS COUNTER registers; all of which are local to the DCS module. The 11/60 processor itself has no control over the setting/clearing of 'ERROR'; in fact, it cannot directly determine whether 'ERROR' is set or clear.

The ENUA register (12 bits) is loaded from the EMIT field of the microword, under control of a DCS rom extension bit. It is setup at the beginning of a test to reflect the "EXPECTED" micro address after the test microbranch ("BUT") is executed.

The TNUA register (12 bits) is loaded continuously as the DCS microcode executes, TRACKING the progress of the microaddress field. This register contains the value of the "RECEIVED" micro address after the test microbranch is executed.

The DCS-CNTR is loaded with a value from (00)-(17) (octal), from the EMIT field of the microword, under DCS rom extension control. This register continuously counts up every microcycle. When the contents of this register is (17), the DCS hardware compares ENUA and TNUA, and does the following:

```
Set 'ERROR'="1" if DCS-CNTR=(17) and ENUA<>TNUA
else leave 'ERROR' unchanged from its previous value.
```

This is the manner by which DCS is able to set 'ERROR'. All DCS tests use this method.

Note also that the DCS hardware "locks up" the loading of the ENUA and TNUA registers after 'ERROR' is set, preserving their contents. Thus only the FIRST 'ERROR' will be recorded. There is no provision to detect subsequent errors until the previous ones are eliminated. See the DCS Maintenance Manual and Print Set for more detailed information.

8.2 FAULT DIRECTORY Format and Use

8.2.1 Basic Structure

The FAULT DIRECTORY is essentially a tabular summary of all ERROR codes the DCS is able to generate - a total of 432 entries occupying 52 pages. Each individual ERROR code entry in the FAULT DIRECTORY contains a short description of the test, and the module replacement information pertaining to that test. For ease of reference, the ERROR codes have been organized into ascending numerical order, in the range 4000(8)-6777(8).

8.2.2 Basic Use - with an example

This section describes how to use the FAULT DIRECTORY after the DCS has been run, and has indicated an error is present in the KD11-K processor.

Assume for the purpose of explanation that the DCS was started, and has returned the following values:

ERCD = 4616 (Error Code)
TNUA = 7405 (Tracking NUA)

with EOP=<OFF>, and ERROR=<ON>

1. Going to the FAULT DIRECTORY, we find the entry for ERROR code 4616 to be on page 9, entry number 73.
2. Some general information about the failing test is first obtained:
 - a) 'Symbolic label' - A reference to the DCS microtest which failed, in this instance TEST-115-A2.
 - b) 'Line number' - A reference to the line number in the DCS microcode listing where the failing test is located (here, line number 5983).
 - c) 'ENUA' - The Expected NUA of this test, in this case 7412. Note that the obtained TNUA (7405) is not the same as the test's ENUA (7412); thus the ERROR.
 - d) The remainder of the line contains a short description of the function performed by this test; in this instance we note the test was diagnosing the ALU portion of the DATA-PATH module.
3. We now note that the TNUA we obtained was 7405. Scanning downward in the column of TNUA entries for this test, we find it listed as the fourth entry. More information, specific to this particular error, can now be obtained:
 - a) 'Module sequence' - These 3+ columns contain (scanning left to right) the top 3+ choices of processor modules to inspect/replace, in order to locate and correct the fault(s). The module choices are listed using "slot" notation (ie, K#, where #=the slot), and a "confidence factor" to indicate the percent confidence that replacing this particular module will eliminate the fault(s). The best choice is the module called out in the first column ("s1"); then "s2" etc. Note that the percentages are rounded to the nearest 5%, and may therefore not always add up to exactly 100%.
 - b) To the right of the module choices is summarized the

IC information obtained from the FAULT INSERTION effort of the DCS/KD11-K (signified by "FI"). IC information is referenced to a particular module by the notation:

K4=E23,E33-36,E89-E90,E101; K2=E12,E15,E69;

- N O T E -
CALLOUT OF SPECIFIC IC'S ON A MODULE IS --NOT-- INTENDED INTENDED TO BE AN "EXHAUSTIVE-ONLY-THOSE-ARE-THE-ONES" LIST. IT IS INTENDED TO PROVIDE REFERENCE TO A SPECIFIC FUNCTIONAL AREA OF A MODULE, AND GIVE REFERENCE TO THOSE IC'S WHICH CAUSED THE FAILURE DURING THE FAULT INSERTION EFFORT OF HARD STUCK-HIGH/-LOW AND ADJACENT-PIN-SHORT TYPE FAULTS. AGAIN, DO NOT ASSUME THIS LIST TO BE ALL INCLUSIVE OF THE POSSIBLE CHOICES FOR FAULTY IC'S.

Another type of entry is of the format:

K404=ALU/CARRY-LOOKAHEAD;
or Kmp=functional-description-of-logic-block

which references a particular module (#m) and page (#pp) in the KD11-K Processor Print set. This notation is used when specific fault insertion data is not available for a test.

4. In the instance when the TNUA obtained does not match any of those provided under a given test/ERROR code entry, a wild-card character ("?") has been used to allow a match with any octal digit. Thus 740? matches 7400, 7401, ..., 7407. These entries should be used for further information or when a specific TNUA is not present.
5. If there is no TNUA listed which matches the obtained TNUA, and also no wildcard entry is present; then the information about the functional nature of the test (from above), along with an intelligent interpretation of the obtained TNUA, will be required. The following table lists some TNUA's that might be obtained in such a case:

(see table on following page)

TNUA	Cause
----	-----
4000	DCS forced to its starting address
4777\	
4756 >	an unexpected JAMUPP condition occurred
4747/	
7361	in UNIBUS function tests, a JAMUPP did not occur when expected
7400-	the "standard" BUT() target area for DCS
7777	micro-tests

8.3 'SCOPE Loop' Facility

8.3.1 General Information

The 'SCOPE loop' implementation provided by DCS is almost identical to that provided in the standard MAINDEC macro diagnostic program. What the 'SCOPE loop' does is to repeatedly execute the same sequence of diagnostic test code; this allows the technician to 'scope' appropriate logic signals in an effort to zero-in on the fault.

The DCS 'SCOPE loop' occurs ONLY and ALWAYS when 'ERROR' is set. There are no user options to change the size or range of the loop - all these parameters have been fixed in microcode and hardware. The loops have been setup to be as tight, and as useful, as was possible. Most are in the range of 10-30 microwords, although some (three, in particular) are larger.

8.3.2 Implementation and Use

A DCS extension rom control bit is used to enable the 'SCOPE loop' check at selected points in DCS code execution. These points are recognized by the following:

```
SCOPE123:
<possible some other functions>
NEXT, BUTD(SCOPE), !NO ERROR: "TEST124" [+1, WORDS]
J/TEST124 ! ERROR: "LOOP123" [-5, WORDS]
```

The two comments "EPROR/NO=ERROR" tell the user where the DCS code will branch, depending upon the current state of 'ERROR'. Usually, the 'NO=ERROR' condition falls thru to the next word (eq, +1, words). For the 'EPROR' case, the loop is ALWAYS backwards (ie, up the page, toward the point where the error was detected). The "-number" notation gives a relative count of the number (approximately) of micro words backwards in the jump.

This facility can be used in two modes - dynamic and static. Either mode must be entered via the use of the DCS 'RUN/STOP' switch set to the 'PUN' position, as this then enables the DCS code to execute continuously. The results are generally undefined if the switch is set to 'STOP', and the "DIAG" button was used to enter the loop.

Dynamic mode requires the use of an oscilloscope, logic analyzer, etc., and the determination of an appropriate logic signal on which to sync. The DCS microcode then automatically remains in this tight loop to allow observation of the suspected faulty signals, at processor cycle speed.

Static mode is entered in the same manner as dynamic; but afterwards the "SINGLE-MICROSTEP/MAINTENANCE-CLOCK" feature of the 11/60 processor (on K6 TIMING module, the two switches - see prints) is enabled. This allows the processor to be single micro - stepped, under user control. The additional debug features of the DCS can now be employed: the BUSDIN/DOOUT display LEDs (16), and the (2) "free" LEDs. See the DCS Maintenance Manual for further details. Also available are the NUA (Next-U-Address) LEDs on the 11/60 processor "UWORD" module (K2). Note that these 'point' at the NEXT microword to be executed, not the current.

9.0 REVISION HISTORY

9.1 Revision Number

After a successful 'END-OF-PASS' indication, console internal exam functions can be used to obtain the 'DCS Microcode Revision Number', stored in the macro machine general register R5. BIT<15> of this number will also be set, indicating successful 'END-OF-PASS' was reached. The initial version of the microcode will display:

(100101)

with subsequent versions to be:

(100102)

(100103) etc

If, however, an 'ERROR' is obtained, one might still be able to obtain the DCS microcode revision number. In this instance BIT<15> will/should be clear, and the lower bits the revision code:

(000101)

(000102) etc

However, this number must be taken with caution, as the error may or may not have influenced the storing of the revision number in the register.

9.2 Revisions To DCS Code

Note that ALL revisions to the DCS microcode / FAULT DIRECTORY are to be documented in this section, with the following information supplied:

- a) A short description of the problem(s) found, and how they were corrected.
- b) Updated MAINDEC (MD) and DCS (uCODE) revision information.
- c) Date of fix, and person responsible for fix.
- d) The test/ERROR codes affected by the changes.

Note that the changes MUST also be incorporated into the DCS microcode listing, and/or the FAULT DIPECTORY listing at the appropriate points. Actual microcode changes will be entered as ECO's to individual ROM patterns on the DCS module (M7871). Contact PDP-11/60 Support Engineering for the procedure.

REVISION MD/UCODE	Date	Who	Explanation
.. ..		..	
A/101	18-Jan-77	DNN	Initial Release
B/101	22-Jun-77	DNN	No microcode changes; documentation added/updated.

10.0 DCS VERIFICATION / SELF-TEST

'VERIFY MODE' is a self-check mode designed to verify the operation of the DCS module and its associated error detection/indication support logic.

10.1 Requirements

This mode of operation requires that a known good PDP-11/60 system (as described in Section 2) be used to test/verify a DCS module, so that errors detected by the DCS are due to the DCS module under test, and not due to the other system components. The set of PDP-11/60 processor macro diagnostics, or a known good DCS module, can be used to perform such a verification of the host system.

10.2 Verification Method

The method (or algorithm) used to perform the DCS self-verification is as follows:

Hardware on the DCS module is conditioned to execute a single pass thru the DCS microcode, via setting the DCS 'NORMAL/VERIFY' switch to the 'VERIFY' position. This also alters the 12 bit hardware counter on the DCS module from a 'Pass Counter' to the 'Verify Counter'.

At the start of a 'Verify Pass', this counter is preset to a specific value, predetermined so that when 'END-OF-PASS' is signaled by the DCS microcode, this counter will have a value of octal (7777), or be at the point of overflow (carry out) enabled.

As the DCS executes in 'VERIFY MODE', this counter is incremented whenever:

1. A microword is executed from page 7, or
2. A microword is executed with the 'VERIFY' bit (a page 4-6 only DCS ROM extension bit) asserted. These 'VERIFY' bits have been scattered, more or less at random, throughout the DCS microcode. Thus this counter will be incremented at random intervals during a 'Verify Pass'.

The DCS code executes approximately 350,000 microwords per pass; thus the counter will overflow between 2-85 times (depending upon the number of 'VERIFY' bits and page 7 references encountered) before the 'END-OF-PASS' / 'Verify Counter' overflow match. Physically, the verify count is retained modulo 4096 (12 bits), with only the low order bits of the count used in the comparison.

A verification will be considered successful only if a verify counter overflow point exactly matches the microword which signals 'END-OF-PASS' (done only once) in the DCS microcode.

10.3 Procedure And Indications

10.3.1 Procedure

To run the DCS in verify mode, the following procedure is followed;

1. Install the DCS in the PDP-11/60 as detailed in Section 3
2. Set the DCS switches:
 'RUN/STOP' = 'STOP' and
 'NORMAL/VERIFY' = 'VERIFY'
3. Now set:
 'RUN/STOP' = 'RUN'
4. The DCS now executes a single 'Verify Pass'
5. At the end of the 'Verify Pass' the DCS enters a microcode loop, in which:
 - An error is forced with specific 'ENUA', 'TNUA', and 'ERROR code' values
 - 'END-OF-PASS' is repeatedly signaled
 - A 'Scope Loop' branch is executed

See the DCS microcode listing, under 'Verify Mode Code' for the exact sequence of operations.

6. At this time examine the DCS module LEDs for comparison with their expected contents, as noted below.
7. To return control to the PDP-11/60 after a 'Verify Pass', position:
 'RUN/STOP' = 'STOP' and
 'NORMAL/VERIFY' = 'NORMAL'

And then generate a "CONSOLE INIT" ("START/HALT") on the operator's console

10.3.2 Indicators

Only the status described below is acceptable to signal a successful DCS verification. Assuming a known good PDP-11/60 system, any deviation from the description (below) should be considered an indication of a fault in the DCS module under

test.

After a 'Verify Pass', indications on the DCS module will be:

'TNUA' = (7522)

'ERROR' = (4255)

(Note the alternating ON/OFF pattern)

'ENUA' = (7523) was loaded to force
an 'ERROR' indication

'EOP' LED = ON, Approx. 1/2 brilliance

'ERROR' LED = ON, continuously

This will be the only instance when both
the 'EOP' and 'ERROR' LEDs should be on
simultaneously.

Indications on the PDP-11/60 console should be:

"RUN" LED = OFF continuously

"PROC" LED = OFF continuously

"USER" LED = OFF continuously

"CONSOLE" LED = OFF/ dimly lit

"BATTERY" LED = <indeterminate>

Octal display = (212121), with the
decimal points either on or off.

11.0 MISCELLANEOUS

11.1 ACT/APT/XXDP

The DCS module is not directly supported by ACT/APT/XXDP software at this time.

11.2 Macro Instruction Interface

11.2.1 DCS Presence

Presence of a DCS module in slot 1 is indicated by a bit in the "WHAMI" register:

NOTE

See PDP-11/60 documentation for a full description of the "MED" instruction.

```
MED      ,022      ,READ WHAMI => R0
BIT      #BIT08,R0  ,BIT<08>=1, DCS PRESENT
                        ,BIT<08>=0, NO DCS IN SLOT-1
```

11.2.2 DCS Register Access

Access to several of the internal registers and status bits is also possible via the "MED" instruction:

```
MED      ,152      ,READ DCS 'TNUA'
```

After execution, R0's contents is as follows:

```
0000_#TNUA<11:00>
```

Similarly:

```
MED      ,153      ,READ DCS 'EOP/ERROR'
```

and R0's contents:

```
ERROR#01#EOP#EPRCOD<11:00>
```

```
BIT<15> = ERROR(1)H
BIT<14:13> = "01", code for DCS module
BIT<12> = EOP(1)H
```

11.2.3 Macro Instruction Startup Of DCS

The DCS may also be started via a "MED" instruction. Note, however, that this is a one-way transfer of control; there is

no means to re-enter an executing macroinstruction program without operator intervention at the operator's console.

This method simulates the operator depression of the "CNTRL/DIAG" keys via loading the KD11-K microaddress pointer (NUA) with the starting address of the "DIAGNOSE" key service routine.

```
MOV    #011410,R0      ,(1141) = "DIAGNOSE"  
MED    ,347            ,WRITE NUA  
;NEVER COME BACK TO HERE  
;DCS ALWAYS EXITS TO "CONSOLE" MODE
```

##	ERROR code	Symbolic label	Line number	ENUA	TNUA	Module sequence #1/#2/#3	Test summary - Print reference - Chip information
1)	4000	TEST001	2633	6252	NUA SEQUENCING, PAGE (4) -> (2)	[2=6], UBF=(34)	
				4000	K3/99	../..	FI; K3=E6,E15
				4203	K2/99	../..	FI; K3=E41,E47
				4252	K2/90	K3/5	K4/6 FI; K2=E13-E14,E34-E38,E40-E41,E46-E47,E58-E59,E62, E64-E68,E67-E68,E72-E74,E78-E80,E91; K3=E34, E111; K4=E3
				4253	K2/99	../..	FI; K2=E14
				4256	K2/99	../..	FI; K2=E34,E46,E59,E68,E74,E80
				4257	K2/99	../..	FI; K2=E40,E68
				4747	K2/75	K3/25	../.. FI; K2=E2,E29,E52-E53,E79; K3=E6,E26
				4756	K2/99	../..	FI; K2=E70
				4757	K3/99	../..	FI; K3=E4,E7
				4776	K2/99	../..	FI; K2=E60
				4777	K2/75	K3/10	K6/5 K4/5 K7/5 FI; K2=E1-E7,E10-E18,E21-E25,E27-E39, E41-E54,E57-E59,E61-E61,E84,E90,E93-E96,E100, E104; K3=E1-E20,E22,E24-E26,E32,E39,E41,E48, E101; K4=E40,E43,E60-E61,E66,E78-E81,E83,E92; K4=E18,E26,E29-E30,E38,E71,E87; K7=E36,E52,E74
				5130	K2/99	../..	FI; K2=E94
				5247	K2/99	../..	FI; K2=E47
				5276	K2/99	../..	FI; K3=E6,E33,E60
				6000	K2/99	../..	FI; K2=E7-E8,E19,E40
				6052	K2/55	K3/45	../.. FI; K2=E1,E7,E117; K3=E58,E91
				6201	K2/99	../..	FI; K2=E70-E71
				6212	K3/55	K2/45	../.. FI; K3=E58,E91; K2=E2,E7
				6231	K2/99	../..	FI; K2=E62
				6235	K2/99	../..	FI; K2=E62
				6242	K3/55	K2/45	../.. FI; K3=E21,E81; K2=E2,E8
				6250	K3/55	K2/45	../.. FI; K3=E21,E81; K2=E2,E8
				6253	K2/55	K3/45	../.. FI; K2=E2,E60,E66,E69,E75,E81; K3=E21,E23,E44,E68
				6256	K2/55	K3/45	../.. FI; K2=E2,E8-E9,E21,E27,E60,E66,E69,E75,E81; K3=E21-E22,E37-E39,E54,E56,E74-E75
				6257	K3/80	K2/20	../.. FI; K3=E37-E38,E74; K2=E8
				6272	K2/95	K3/5	../.. FI; K2=E2-E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E52
				6277	K2/99	../..	FI; K2=E2,E13
				6382	K2/80	K3/20	../.. FI; K2=E1,E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E51, E58,E70
				6372	K2/99	../..	FI; K3=E7
				6682	K2/80	K3/20	../.. FI; K3=E1,E19,E42,E59,E65,E68,E74,E80; K3=E23,E63
				7077	K2/99	../..	FI; K2=E117
				7231	K2/99	../..	FI; K2=E14
				7252	K2/99	../..	FI; K2=E35,E47,E58
				7752	K2/99	../..	FI; K2=E48
				7777	K2/99	../..	FI; K2=E48

Module codes: K1/DCS K2/UNWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##	ERROR code	Symbolic label	Line number	ENUA	TNUA	Module sequence #1/#2/#3	Test summary - Print reference - Chip information
2)	4255	VFY003	18313	7523	THIS ERROR-CODE EXPECTED IN DCS "VERIFY MODE"		
				7522	/..	/..	THIS TNUA EXPECTED IN DCS "VERIFY MODE"
				7522	K6/80	K7/40 K3/5	K2/5 K4/5 FI; K6=E3-E8,E12,E14,E22,E26,E31-E32, E34,E37-E40,E42-E43,E46,E54,E56,E61-E62,E64,E69, E71-E72,E76,E78-E81,E86-E88,E90,E94-E96,E98-E102, E104-E105,E108; K7=E2,E4,E8,E12,E14-E15,E18-E23, E25-E27,E34-E38,E37-E40,E43,E45-E47,E49,E52,E54, E59,E62-E63,E66,E68,E70-E71,E87,E98,E97-E98; K3=E2,E6-E8,E10,E12,E18,E21-E26,E34,E52,E54,E64; K2=E2-E3,E6-E9,E21,E36,E50,E60,E66,E75,E81; K4=E8,E16
				????	K1/99	/..	DCS VERIFY MODE, BAD TNUA LATCHED
3)	4177	TFST007	2749	7303	NUA SEQUENCING, PAGE (4) => (3)	[3=7], UBF=(35)	
				7301	K2/99	/..	FI; K2=E61
				7303	K2/65	K4/35 K3/5	K6/5 FI; K2=E4,E10,E16,E22,E25-E26,E28,E31,E37, E43,E49,E52,E70-E71,E76-E77,E97; K4=E1,E12-E15, E29,E38,E70; K3=E33; K6=E34
				7307	K2/60	K3/40	/.. FI; K2=E9,E21,E27; K3=E21-E22
				7343	K2/99	/..	FI; K2=E7
				7376	K3/99	/..	FI; K3=E34
				7777	K2/99	/..	FI; K2=E23,E29
4)	4450	TEST730C1	17211	7402	BUTA(LAST) CLEARS PROC BBSY (DATIP); EMIT=(052525) ON SUBDIN		
				7400	K5/99	/..	FI; K5=E94
				7401	K6/50	K7/35 K5/15	FI; K6=E4,E61,E70,E79,E103; K7=E31,E39,E46,E49; K5=E95,E97,E105
5)	4451	TFST740D	17653	7402	INSTRI=OVERLAP, BUS CYCLE YANKED		
				4747	K5/65	K6/35	/.. FI; K5=E95; K6=E40
				7401	K6/99	/..	FI; K6=E43
6)	4452	TEST740C	17619	7402	-INSTRI=PREFETCH, BC<0>=BC<0>		
				4747	K2/99	/..	FI; K2=E41
				7401	K2/99	/..	FI; K2=E105
7)	4453	TEST740B	17591	7402	INSTRI=PREFETCH, BC<0>=0		
				7401	K2/65	K3/35	/.. FI; K2=E82,E105; K3=E46
8)	4454	TFST731F	17467	7432	BUS DATI=CLKIR, BA=(0000000); IN=(000125) E88/(432) LOADED		
				4747	K3/99	/..	FI; K3=E1
				7402	K6/99	/..	FI; K6=E5
				7407	K6/99	/..	FI; K6=E4,E108
				7412	K6/85	K3/15	/.. FI; K6=E3,E5; K3=F45

Module codes: K1/DCS K2/UNWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERRR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence-> #1/% #2/% #3/%	Test summary - Print reference - Chip information
9)	4455	TEST731D	17428	7402		BUS DATI=CLKIR, BA=(000000); SERVICE=(000340) 4747 K6/99/.. FI; K6=E19 7400 K6/99/.. FI; K6=E108	
10)	4456	TEST731C	17401	7432		BUS DATI, BA=(000000); MEM(000000)=(000125) VIA DATO/DATOB/DATI 7400 K6/99/.. FI; K6=E93 7402 K6/99/.. FI; K6=E16,E43,E85,E90,E93 7405 K4/99/.. FI; K4=E28 7420 K5/99/.. FI; K5=E94,E105 7434 K6/99 K7/15/.. FI; K6=E11,E55,E71; K7=E49	
11)	4457	TEST731B	17363	7412		IR=(128200) E78/(412); NO BUS FUNCTIONS CLOCK-IR EXCEPT DATI=CLKIR 7400 K2/65 K3/35/.. FI; K2=E34,E84; K3=E48	
12)	4460	TEST730C	17188	7402		DATIP HOLDING BUS (PROC BBSY); DU=(125252) ON BUSDIN 7401 K6/65 K7/35/.. FI; K6=E70,E79,E103; K7=E17,E62	
13)	4461	TEST730D	17233	7402		BUS DATO, BA=(000000); DATIB=BYTE+ODD, BA=(000001); SERVICE=(000340) 4747 K6/50 K2/50/.. FI; K6=E47; K2=E105 7400 K6/90 K7/15/.. FI; K6=E18-E39,E54; K7=E49 7401 K6/99/.. FI; K6=E18	
14)	4462	TEST730F	17286	7402		BUS DATO, BA=(000000); DATIB=BYTE+ODD, BA=(000001); DU=BUSDIN=(052525) 7400 K6/90 K5/10 K4/5 FI; K6=E5,E46,E66-E67,E72,E74-E75,E80,E83-E84,E86, E88,E90,E92-E93,E108; K5=E53,E63,E99-E101,E109, E112-E113,E115,E122; K4=E28 7403 K6/99/.. FI; K6=E92-E93	
15)	4464	TEST722B	17004	7402		INVALIDATE, ODD ADDR JAM; JAM=(101004) ???? K6/99/.. K605=UNIBUS-FUNCTION-DECODE	
16)	4465	TEST722C	17029	7402		INVALIDATE, 16, BIT PBA, -I/O PAGE(6); SERVICE=(002340) 7400 K6/99/.. FI; K6=E59 ???? K6/99/.. K605=UNIBUS-FUNCTION-DECODE	
17)	4466	TEST730A	17149	7402		BUS DATIP, BA=(000001); DU=(128252) AFTER ON BUSDIN 4747 K7/65 K6/35/.. FI; K7=E16,E25,E32; K6=E55,E76,E88 7400 K6/95 K5/5/.. FI; K6=E66-E67,E74-E75,E83-E84,E92-E93; K5=E53,E63, E66 7401 K6/65 K7/35/.. FI; K6=E11,E19,E47,E55,E63,E66-E67,E69,E71,E74-E76, E79,E83-E84,E92-E93,E98-E99,E101,E103,E105; K7=E10,E17,E21,E32-E33,E49,E101 7403 K6/99/.. FI; K6=E92-E93	

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERRR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence-> #1/% #2/% #3/%	Test summary - Print reference - Chip information
18)	4471	TEST730A	17084	7402		BUS DATO, BA=(000000); D=(128252); CHECK DBUF=D AFTER 4747 K6/65 K7/25 K5/10 FI; K6=E2-E3,E19-E20,E26,E40,E76,E85,E87-E88, E96-E98; K7=E2,E10,E32-E33,E36,E57-E58,E60; K5=E35,E42-E43,E62 7400 K5/99/.. FI; K5=E90,E107-E108,E114,E120-E121 7401 K6/65 K7/20 K4/10 K3/5 FI; K6=E11,E56,E64,E69,E72,E85,E101,E103; K7=E10,E33; K4=E28,E55; K3=E42 7403 K6/50 K5/25 K2/25 FI; K6=E101; K5=E91; K2=E100	
19)	4472	TEST132P2	7572	7417		BUTR(X=D[C]XX), D[C]=COUT15="1"; A(0)+B(1)+C(1)=D(0)+CO(1) 7413 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
20)	4473	TEST136P2	8010	7402		(122645)+(132264)+(0)=(170360); CHECK COUT15=(0) 7403 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
21)	4474	TEST006	2730	4377		NVA SEQUENCING, NO 'BUT' 4777 K2/99/.. FI; K2=E17,E63	
22)	4475	TEST136B1	7983	7434		(122645)+(132264)+(0)=(170360); CARRY LOOKAHEAD LOGIC 7400 K4/99/.. FI; K4=E47 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
23)	4476	TEST136A2	7961	7403		(055132)+(132264)+(1)=(007417); CHECK COUT15=(1) 7286 K2/80 K3/20/.. FI; K2=E8-E9,E21,E27; K3=E21 7402 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
24)	4477	TEST135B2	7904	7413		(055132)+(132264)+(0)=(170360); CHECK COUT15=(0) 7417 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
25)	4500	TEST133A2	7630	7403		(103607)+(103607)+(1)=(007417); CHECK COUT15=(1) 7402 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
26)	4501	TEST133B1	7652	7434		(074170)+(074170)+(0)=(170360); CARRY LOOKAHEAD LOGIC 7402 K4/80 K2/50/.. FI; K4=E49; K2=E102 7420 K4/99/.. FI; K4=E49 7421 K3/99/.. FI; K3=E71 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
27)	4502	TEST133P2	7679	7402		(074170)+(074170)+(0)=(170360); CHECK COUT15=(0) 7403 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
28)	4503	TEST134A2	7742	7403		(045513)+(141703)+(1)=(007417); CHECK COUT15=(1) 7402 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
29)	4504	TEST134B1	7764	7434	(132264)+(036074)+(0)=(170360); CARRY LOOKAHEAD LOGIC 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
30)	4505	TEST134P2	7791	7401	(132264)+(036074)+(0)=(170360); CHECK COUT15=(0) 7403 K3/99 ../.. FI; K5=E95 7407 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
31)	4506	TEST135A2	7855	7417	(122645)+(064551)+(1)=(007417); CHECK COUT15=(1) 7413 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
32)	4507	TEST135B1	7877	7434	(055132)+(113226)+(0)=(170360); CARRY LOOKAHEAD LOGIC 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
33)	4511	TEST134A1	7705	7434	(045513)+(141703)+(1)=(007417); CARRY LOOKAHEAD LOGIC 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
34)	4513	TEST135P1	7818	7434	(122645)+(064551)+(1)=(007417); CARRY LOOKAHEAD LOGIC 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
35)	4515	TEST136A1	7931	7434	(055132)+(132264)+(1)=(007417); CARRY LOOKAHEAD LOGIC 7421 K4/75 K3/25 ../.. FI; K4=E64; K3=E74 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
36)	4516	TEST132A2	7527	7403	BUTR(D(C18A00), D(C)=COUT15="1"; A(1)+B(0)+CI(1)=D(0)+CO(1) 7402 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
37)	4517	TEST132B1	7547	7434	ALU=DIVIDE=A-MINUS-B; A(052525), B(052525), D(000000), D(C)="1" 7400 K3/80 K4/20 ../.. FI; K3=E41,E53,E61; K4=E16 7402 K3/99 ../.. FI; K3=E51 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
38)	4521	TEST132A1	7502	7434	ALU=A-MINUS-B; A(125252), B(125252), D(000000) 7400 K4/70 K3/30 ../.. FI; K4=E49; K3=E42,E92 7402 K4/85 K3/45 ../.. FI; K4=E6,E11,E17,E26,E96; K3=E41,E82 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
39)	4523	TEST133A1	7600	7434	(103607)+(103607)+(1)=(007417); CARRY LOOKAHEAD LOGIC 7400 K4/75 K3/25 ../.. FI; K4=E8,E49,E71; K3=E53 7402 K3/99 ../.. FI; K3=E43,E84 7420 K4/99 ../.. FI; K4=E49,E88 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
40)	4530	TEST130R1	7334	7434	ALU=A-PLUS-B-PLUS-1; A(125252), B(125252), D(052525) 7400 K4/70 K3/15 K2/15 FI; K4=E49,E101-E102,E110; K3=E53; K2=E50 7402 K4/80 K3/20 ../.. FI; K4=E49,E98,E101; K3=E41				

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
	4530	TFST130R1	[Continued]		ALU=A-PLUS-B-PLUS-1; A(125252), B(125252), D(052525) 7406 K4/99/.. FI; K4=E110 7420 K4/99/.. FI; K4=E49,E102,E112 7421 K4/99/.. FI; K4=E71 7436 K4/99/.. FI; K4=E112 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
41)	4532	TFST130B2	7365	7403	BUTR(COUNT=D(C)), D(C)=COUT15="1"; A(1)+B(1)+CI(0)=D(0)+CO(1) 7402 K2/50 K3/30 K4/25 FI; K2=E35,E41,E61,E77,E85; K3=E4,E6-E7,E11,E15, E43; K4=E63-E64				
42)	4537	TFST320D	8197	7403	BUTR(COUT07=DOUT07:XX), TARGET="01" 7401 K3/88 K4/15/.. FI; K3=E53,E55-E56,E95; K4=E10 7407 K3/99/.. FI; K3=E74				
43)	4541	TFST131A1	7392	7434	ALU=A-PLUS-B-PLUS-PS(C); A(125252), B(052525), D(177777), PS(C)=(0) 7400 K3/75 K4/25/.. FI; K3=E35,E41,E46,E71,E84; K4=E49 7402 K3/99/.. FI; K3=E82 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
44)	4543	TEST320A	8098	7403	D(C) ADDR SELECT, CODE="010"=ALU00="1" ONLY ONE SET 7402 K4/99/.. K404=D(C)-SELECT-LOGIC				
45)	4544	TFST131A2	7423	7401	BUTR(D(C18A00), D(C)=COUT15="0"; A(1)+B(0)+CI(0)=D(1)+CO(0) 7403 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
46)	4546	TEST131R1	7443	7434	ALU=DIVIDE=A-PLUS-B; A(052525), B(125252), D(177777), D(C)="0" 7400 K3/99/.. FI; K3=E82 7402 K3/99/.. FI; K3=E61,E84 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
47)	4550	TEST130A2	7314	7402	BUTR(COUNT=D(C)), D(C)=COUT15="0"; A(0)+B(0)+CI(1)=D(1)+CO(0) 7403 K4/99/.. FI; K4=E63-E64				
48)	4552	TFST320B	8138	7400	BUT(D<14:00>=08D15), D=(000001), TARGET="00" 7402 K2/99/.. FI; K2=E40,E88,E94,E99,E112				
49)	4553	TEST762A1	17899	NONE	DL11-W ENABLED FOR BR6 INTR; DID NOT RESPOND W/ BG-SERVICE-L WITHIN 22.45. 4553 K7/85 K6/10 K5/5 K3/5 FI; K7=E3,E5-E6,E11,E13,E16-E17,E19-E20, E25-E26,E28,E35-E36,E39,E76,E80; K6=E28,E34,E62; K5=E1; K3=E74				
50)	4554	TEST320E	8217	7403	D(C) ADDR SELECT, CODE="101"=COUT07="1" ONLY ONE SET 7401 K4/99/.. FI; K4=E63				
51)	4556	TEST320F	8256	7405	BUTR(COUT07=DOUT07:XX), TARGET="10" 7407 K3/99/.. FI; K3=E56,E95				

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERROR	Code	Symbolic label	Line number	ENVA	TNUA	--Module sequence--	Test summary - Print reference - Chip information
---	---	-----	-----	-----	-----	#1/% #2/% #3/%	-----
52)	4#62	TEST320C	8157	7413		BUTR(X#D(C)X#XX), D(C)=COUNT7="0", COUNT15="1" 7417 K4/99/.. FI; K4#E63-E64	
53)	4#66	TEST551C	14539	NONE	B.M. DATAPATH COUNTER, BUTA(COUNT-18-377#D(C)), COUNT FROM (000):(377) 4531 K3/99/.. FI; K3#E56 ???? K2/80 K4/20/.. K207=ACTIVE-BUT; K409=CNTR		
54)	4#70	TEST122A2	7172	7403		BUTR(COUNT#D(C)), D(C)=CINMUX="1", ALU=A-IOR-B 7402 K3/99/.. FI; K3#E43,E61,E71	
55)	4#71	TEST721A	16840	4777		DATOB=BYTE, ODD ADDR JAM, BA=(120001) 7341 K6/99/.. K608=UNIBUS-FUNCTION-DECODE	
56)	4#72	TEST537A	14413	7402		SHIFTER, 'SENDMUX' CMUX#000 OUTPUT ON CMUX/LEFT-1 7400 K4/90 K5/5 K3/5 FI; K4#E77-E78,E80; K5#E97; K3#E61	
57)	4#73	TEST721B	16894	7402		DATOB=BYTE, ODD ADDR JAM; JAM=(101004) 7407 K7/99/.. K705=JAM=FLAGS,K708=STATUS=MUX	
58)	4#75	TEST451A	14469	NONE	B.M. DATAPATH COUNTER, BUTA(BR1-0#COUNT-18-377), COUNT FROM (000):(377) 4531 K3/75 K4/25/.. FI; K3#E36,E54; K4#E91 4547 K3/90 K4/10/.. FI; K3#E36,E54,E56,E64; K4#E91 4555 K4/90 K2/10/.. FI; K4#E20,E54,E87,E90-E91; K2#E55 ???? K2/80 K4/20/.. K207=ACTIVE-BUT; K409=CNTR		
59)	4#76	TEST551B	14498	7400	B.M. DATAPATH COUNTER, BUTA(COUNT-18-377), COUNT FROM (000):(377) 7401 K3/99/.. FI; K3#E64 7402 K3/99/.. FI; K3#E44 ???? K2/80 K4/20/.. K207=ACTIVE-BUT; K409=CNTR		
60)	4#77	TEST721C	16918	7402		DATOB=BYTE, 16. BIT PBA, -I/O PAGE(5); SERVICE=(002340) 7400 K6/99/.. FI; K6#E59	
61)	4#01	TEST115A1	5939	7412		ALU=NOT-A; A(052525), B(177777), D(125252), BITS<15:12>=(12) 7400 K4/85 K7/10 K2/8 FI; K4#E1,E4-E5,E7,E15-E18,E22,E24,E34,E40-E42, E98; K7#E41,E70; K2#E10 7402 K4/99/.. FI; K4#E33-E34 7403 K4/75 K2/25/.. FI; K4#E261; K3#E39 7405 K4/85 K3/45/.. FI; K4#E14,E21,E23; K3#E43,E52-E53,E84 7407 K4/99/.. FI; K4#E4 7410/.. FI; K4#E34 7413 K4/99/.. FI; K4#E26,E34,E62 7416 K4/99/.. FI; K4#E17,E34,E54 7417 K4/99/.. FI; K4#E1,E6,E8,E10,F14-E18,E20-E21,E23-E24, E26-E27,E29,E37,E44,E46	

```
-----
```

Module codes:	K1/DCS	K2/UWORD	K3/IRDECODE	K4/DATAPATH	K5/KTCACHE	K6/TIMING	K7/STATUS
---------------	--------	----------	-------------	-------------	------------	-----------	-----------

```
-----
```

PROR	code	Symbolic label	Line number	ENUA	TNUA	=>Module sequence=>	Test summary = Print reference = Chip information
---	---	-----	-----	-----	-----	#1/% #2/% #3/%	-----
62)	4602	TFST115B4	6044	7402	BUTR(COUNT#D[C]), D(C)=PS(C)="0"		
					7403 K4/50 K3/45 K2/10 FI; K4=E2-E3,E56,E63; K3=E36,E57,E63; K2=E70		
63)	4603	TFST114A2	5906	7402	BUTR(COUNT#D[C]), D(C)=CINMU#PS(C)="0", ALU=ZERO		
					7403 K4/50 K3/50 ../.. FI; K4=E56,E63,E72,E78; K3=E22,E41,E54,E71-E73		
					7406 K3/99 ../.. FI; K3=E74		
64)	4604	TFST115A4	5967	7403	BUTR(COUNT#D[C]), D(C)=CINMU#="1", ALU=NOT-A		
					7402 K4/55 K3/45 ../.. FI; K4=E2-E3,E56,E63,E71-E72,E78,E86; K3=E41,E43,E46,E54,E61-E62,E71,E75		
65)	4605	TEST121C2	7009	7412	ALU=A-XOR-B; A(000000), B(125252), D(125252), BITS<11:06>=(52)		
					7434 K7/99 ../.. FI; K7=E20		
					74?? K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
66)	4606	TEST116B	6326	7434	ALU=ZERO; A(125252), B(052525), D(000000)		
					7402 K3/99 ../.. FI; K3=E102		
					74?? K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
67)	4607	TFST114A	5874	7434	ALU=ZERO; A(177777), B(177777), D(000000)		
					7400 K3/99 ../.. FI; K3=E43,E82		
					74?? K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
68)	4611	TFST115C1	6104	7412	ALU=NOT-A; A(052525), B(000000), D(125252), BITS<15:12>=(12)		
					7400 K3/99 ../.. FI; K3=E51,E82		
					7405 K4/99 ../.. FI; K4=E6,E17,E23,E31-E34,E40-E43		
					74?? K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
69)	4612	TFST115B3	6077	7432	ALU=NOT-A; A(125252), B(177777), D(052525), BITS<05:00>=(25)		
					7420 K4/90 K5/10 ../.. FI; K4=E28,E37,E40-E41, K5=E66		
					74?? K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
70)	4613	TFST115B2	6060	7405	ALU=NOT-A; A(515252), B(177777), D(052525), BITS<11:06>=(25)		
					7400 K4/90 K2/10 ../.. FI; K4=E37,E41-E42,E89,E108; K2=E26		
					7402 K4/99 ../.. FI; K4=E98		
					74?? K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
71)	4614	TEST115B1	6021	7405	ALU=NOT-A; A(125252), B(177777), D(052525), BITS<15:12>=(05)		
					7400 K4/99 ../.. FI; K4=E16		
					7401 K4/99 ../.. FI; K4=E34		
					7404 K4/99 ../.. FI; K4=E34		
					7407 K4/99 ../.. FI; K4=E34		
					7412 K4/99 ../.. FI; K4=E14,E23		
					7415 K4/99 ../.. FI; K4=E34		
					7417 K4/90 K6/10 ../.. FI; K4=E4,E12-E13,E21-E24; K6=E34		

```
-----
```

Module codes:	K1/DCS	K2/UWORD	K3/TRDFCODE	K4/DATAPATH	K5/KTCACHE	K6/TIMING	K7/STATUS
---------------	--------	----------	-------------	-------------	------------	-----------	-----------

FRPDR ##	code	Symbolic label	Line number	ENUA	TNUA	Module sequence #1/% #2/% #3/%	Test summary - Print reference - Chip information
72)	4615	TEST115A3	6000	7425	ALU=NOT-A; A(052525), B(177777), D(125252), BITS<05:00>=(52) 7420 K4/90 K5/10 ../.. FI; K4=E31,E37,E40-E41,E44,E61,E75-E76; K5=E63,E66 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
73)	4616	TEST115A2	5983	7412	ALU=NOT-A; A(052525), B(177777), D(125252), BITS<11:06>=(52) 7400 K4/80 K2/15 K3/8 FI; K4=E5-E7,E11,E17-E18,E20,E24,E26,E28,E32, E41-E43,E60,E66-E68,E88-E89,E108; K2=E32-E33, E38-E39,E44-E46,E50-E51,E53-E54; K3=E27,E41,E82 7402 K3/99 ../.. FI; K3=E53,E61 7403 K5/99 ../.. F2; K5=E83 7405 K4/90 K3/15 ../.. FI; K4=E10,E22,E24,E98; K3=E84 7434 K4/80 K2/50 ../.. FI; K4=E71; K2=E117 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
74)	4617	TEST115C2	6126	7412	ALU=NOT-A; A(052525), B(000000), D(125252), BITS<11:06>=(52) 7400 K4/99 ../.. FI; K4=E41-E42 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
75)	4620	TEST116A3	6271	7412	ALU=NOT-A-AND-B; A(000000), B(125252), D(125252), BITS<11:06>=(52) 7400 K4/99 ../.. FI; K4=E41-E42,E89 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
76)	4621	TEST116A2	6257	7403	BUTR(COUNT=D(C)), D(C)=ALU15="1", D=(125252) 7402 K4/99 ../.. FI; K4=E2-E3,E56,E63-E64		
77)	4623	TEST116A1	6235	7412	ALU=NOT-A-AND-B; A(000000), B(125252), D(125252), BITS<15:12>=(12) 7400 K4/99 ../.. FI; K4=E15,E24,E31-E34,E40-E43 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
78)	4624	TEST115D3	6203	7432	ALU=NOT-A; A(125252), B(000000), D(052525), BITS<05:00>=(25) 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
79)	4625	TEST115D2	6186	7405	ALU=NOT-A; A(125252), B(000000), D(052525), BITS<11:06>=(25) 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
80)	4626	TEST115D1	6164	7405	ALU=NOT-A; A(125252), B(000000), D(052525), BITS<15:12>=(05) 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
81)	4627	TEST115C3	6143	7425	ALU=NOT-A; A(052525), B(000000), D(125252), BITS<05:00>=(52) 7420 K4/80 K3/50 ../.. FI; K4=E40; K3=E11 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
82)	4630	TEST116C5	6429	7401	BUTR(DIC)BA00, D(C)=D(C)="0" 7403 K4/99 ../.. FI; K4=E63-E64,E70		

Module codes: K1/DCS K2/WORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRDRP ##	code	Symbolic label	Line number	ENUA	TNUA	Module sequence #1/% #2/% #3/%	Test summary - Print reference - Chip information
83)	4631	TEST116C4	6413	7432	ALU=NOT-A-AND-B; A(000000), B(052525), D(052525), BITS<05:00>=(25) 4631 K7/99/.. FI; K7=E43 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
84)	4632	TEST116C3	6396	7405	ALU=NOT-A-AND-B; A(000000), B(052525), D(052525), BITS<11:06>=(25) 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
85)	4633	TEST116C2	6381	7402	BUTR(COUNT=D(C)), D(C)=ALU15="0", D=(052525) 7403 K4/99/.. FI; K4=E63-E64		
86)	4635	TEST116C1	6359	7405	ALU=NOT-A-AND-B; A(000000), B(052525), D(052525), BITS<15:12>=(05) 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
87)	4636	TEST116A5	6305	7403	BUTR(DIC)BA00, D(C)=D(C)="1" 4747 K2/99/.. FI; K2=E46,E55 7401 K4/90 K3/10/.. FI; K4=E2-E3,E56,E63-E64,E93,E104,E113-E114; K3=E44,E62		
88)	4637	TEST116A4	6288	7425	ALU=NOT-A-AND-B; A(000000), B(125252), D(125252), BITS<05:00>=(52) 7420 K4/99/.. FI; K4=E40,E88 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
89)	4640	TEST117B2	6592	7412	ALU=A-AND-NOT-B; A(177777), B(052525), D(125252), BITS<11:06>=(52) 7400 K4/99/.. FI; K4=E108 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
90)	4641	TEST117B1	6570	7412	ALU=A-AND-NOT-B; A(177777), B(052525), D(125252), BITS<15:12>=(12) 7400 K4/99/.. FI; K4=E4,E10,E23-E24 7410 K4/99/.. FI; K4=E80 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
91)	4642	TEST117A4	6549	7413	BUTR(X=D(C)XX), D(C)=CINMUX=D(C)="0", ALU=A-AND-NOT-B 7417 K3/85 K4/35/.. FI; K3=E71,E75; K4=E63 7433 K3/99/.. FI; K3=E68 7453 K3/99/.. FI; K3=E70 7513 K3/99/.. FI; K3=E70 7613 K3/99/.. FI; K3=E70 7653 K3/99/.. FI; K3=E70		
92)	4643	TEST117A3	6534	7432	ALU=A-AND-NOT-B; A(177777), B(125252), D(052525), BITS<05:00>=(25) 7420 K4/85 K3/35/.. FI; K4=E88-E89; K3=E61 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
93)	4644	TEST117A2	6517	7405	ALU=A-AND-NOT-B; A(177777), B(125252), D(052525), BITS<11:06>=(25) 7402 K7/99/.. FI; K7=E37		

[Continued]

Module codes: K1/DCS K2/WORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

MAINDEC-11-DQMUR-RO			PDP-11/60 (KD11-K)		DCS FAULT DIRECTORY			Page 12			SFO 0042
##	PROR code	Symbolic label	Line number	ENUA	TNUA	-->Module sequence-->			Test summary - Print reference - Chip information		
						81/99	82/99	83/99			
107)	4663	TEST121B1	6906	7412	ALU=A-XOR-B;	A(177777), B(082525), D(125252), BITS<15:12>=(12)					
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
108)	4665	TEST121A1	6832	7405	ALU=A-XOR-B;	A(000000), B(052525), D(052525), BITS<15:12>=(05)					
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
109)	4667	TEST120A1	6685	7412	ALU=A-AND-B;	A(125252), B(177777), D(125252), BITS<15:12>=(12)					
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
110)	4671	TEST122A1	7148	7434	ALU=A-TOR-B;	A(000000), B(000000), D(000000)					
					7400	K4/99	../..	../..	FI; K4=E1,E7-E8,E11,E13,E22,E92,E96,E100-E102, E109-E112		
					7402	K4/99	../..	../..	FI; K4=E7,E10,E21-E22,E101		
					7417	K4/99	../..	../..	FI; K4=E110		
					7420	K4/99	../..	../..	FI; K4=E22,E102,E112		
					7421	K4/99	../..	../..	FI; K4=E22,E112		
					7422	K4/99	../..	../..	FI; K4=E112		
					7424	K4/99	../..	../..	FI; K4=E21,E112		
					7426	K4/99	../..	../..	FI; K4=E102		
					7430	K4/99	../..	../..	FI; K4=E102		
					7433	K4/99	../..	../..	FI; K4=E102		
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
111)	4673	TEST130A1	7283	7434	ALU=A-PLUS-B-PLUS-0;	A(052525), B(052525), D(125252)					
					4777	K8/99	../..	../..	FI; K2=E79		
					7400	K4/60	K3/40	../..	FI; K4=E1,E6,E49,F64,F88-E89,E98,E101,E108,E110; K3=E35,E42,E82,E92		
					7402	K4/70	K3/30	../..	FI; K4=E21,E49,E98,E100-E102,E109-E112; K3=E41,E43, E52-E53,E84		
					7420	K4/99	../..	../..	FI; K4=E49,E102,E112		
					7421	K4/85	K3/15	../..	FI; K4=E49,E71; K3=E71		
					7423	K4/99	../..	../..	FI; K4=E112		
					7427	K4/99	../..	../..	FI; K4=E112		
					7431	K4/99	../..	../..	FI; K4=E102		
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
112)	4674	TEST121A4	6885	7403	BUTR(DIC)BA00;	DIC=ALU00="1", D=(052525)					
					7401	K4/99	../..	../..	FI; K4=E56,E63,E114		
113)	4675	TEST121A3	6871	7432	ALU=A-XOR-B;	A(000000), B(052525), D(052525), BITS<05:00>=(25)					
					7420	K4/99	../..	../..	FI; K4=E10		
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		
114)	4676	TEST121A2	6854	7405	ALU=A-XOR-B;	A(000000), B(052525), D(052525), BITS<11:06>=(25)					
					7477	K4/90	K3/10	../..	K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE		

Module codes: K1/DCS K2/UNORD K3/TRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS											

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	--Module sequence-- #1/% #2/% #3/%			Test summary - Print reference - Chip information
115	4677	TEST117C2	6655	7417	BUTR(X#D(C)XX), D(C)=CINMUX=D(C)=1", ALU=A-AND-NOT-B 4756 K3/50 K2/50 ../.. FI; K3=E12,E18; K2=E80-E81 7413 K4/90 K2/10 ../.. FI; K4=E2-E3,E56,E63,E104; K2=E1				
116	4701	TFST350	8337	NONE	(NUA SEQUENCING LOGIC ERROR) 7777 K1/99 ../.. ../.. INTERNAL DCS ERROR				
117	4705	TEST352A	8670	7434	BSPL0 ADDRESSING, USING BSP/RIF ADDRESS=(02) 7403 K4/99 ../.. ../.. FI; K4=E13 7417 K4/99 ../.. ../.. FI; K4=E13 7477 K4/99 ../.. ../.. K407=BSF-ADDRS-MUX/REG(RIF)				
118	4707	TEST121C1	6987	7412	ALU=A-XOR-B; A(000000), B(125252), D(125252), BITS<15:12>=(12) 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
119	4710	TEST763D	18148	7407	BUTR(BG-SERVICE) NEGATED; AFTER INTR DETECTED/SERVED 4747 K5/99 ../.. ../.. FI; K5=E81-E82 7403 K7/99 ../.. ../.. K703=BR-REQUEST-INTR,K704=BR-GRANT-INTR				
120	4711	TEST762F	18031	7403	BUTR(SERVICE) ASSERTED; DL11-W BR6 INTR PRESENT 7402 K2/99 ../.. ../.. FI; K2=E100				
121	4712	TEST351D	8634	7434	ASPHI ADDRESSING, USING ASP/RIF ADDRESS=(05) 4747 K7/99 ../.. ../.. FI; K2=E1 7477 K4/99 ../.. ../.. K406=ASP-ADDRS-MUX/REG(RIF)				
122	4713	TEST352B	8692	7434	BSPL0 ADDRESSING, USING BSP/RIF ADDRESS=(03) 7477 K4/99 ../.. ../.. K407=BSF-ADDRS-MUX/REG(RIF)				
123	4716	TEST121C4	7040	7417	BUTR(X#D(C)XX), D(C)=ALU07="1", D(125252) 7413 K4/99 ../.. ../.. FI; K4=E56,E63				
124	4717	TEST121C3	7025	7425	ALU=A-XOR-B; A(000000), B(125252), D(125252), BITS<05:00>=(52) 7420 K4/99 ../.. ../.. FI; K4=E88 7477 K4/90 K3/10 ../.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
125	4720	TEST763B	18108	7402	VECTOR LOADED FROM DL11-W LINE CLOCK; CHECK=(000100) 7400 K7/85 K6/15 ../.. FI; K7=E13,E17,E20-E21,E28; K6=E34				
126	4721	TFST763C	18128	7401	BUTR(VECTOR-LOAD) NEGATED; AFTER INTR DETECTED/SERVED 7403 K7/80 K6/20 ../.. FI; K7=E10,E21,E94; K6=E26				
127	4722	TEST351B	8591	7434	ASPL0 ADDRESSING, USING ASP/RIF ADDRESS=(03) 7477 K4/99 ../.. ../.. K406=ASP-ADDRS-MUX/REG(RIF)				

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	--Module sequence-- #1/% #2/% #3/%			Test summary - Print reference - Chip information
128	4723	TFST351C	8612	7434	ASPHI ADDRESSING, USING ASP/RIF ADDRESS=(04) 4777 K2/99/../.. FI; K2=E1 7271 K2/99/../.. FI; K2=E28 7400 K4/99/../.. FI; K4=E14 7477 K4/99/../.. K406=ASP-ADDRS-MUX/REG(RIF)				
129	4724	TFST352C	8713	7434	BSPHI ADDRESSING, USING BSP/RIF ADDRESS=(04) 7400 K4/99/../.. FI; K4=E12 7477 K4/99/../.. K407=BSP-ADDRS-MUX/REG(RIF)				
130	4725	TFST352D	8735	7434	BSPHI ADDRESSING, USING BSP/RIF ADDRESS=(05) 7477 K4/99/../.. K407=BSP-ADDRS-MUX/REG(RIF)				
131	4726	TFST121D2	7083	7405	ALU=A-XOR-B; A(177777), B(125252), D(052525), BITS<11:06>=(25) 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
132	4727	TEST121D1	7061	7405	ALU=A-XOR-B; A(177777), B(125252), D(052525), BITS<15:12>=(05) 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
133	4730	TEST762E	18006	7402	DL11-W BR6 INTR PENDING, LAST BUS DATO; SERVICE=(043740) 7400 K3/99/../.. FI; K3=E80 7401 K2/99/../.. FI; K2=E100 7407 K7/99/../.. K708=STATUS-MUX				
134	4732	TEST351A	8569	7434	ASPL0 ADDRESSING, USING ASP/RIF ADDRESS=(02) 7400 K4/99/../.. FI; K4=E9,E14 7402 K7/99/../.. FI; K7=E52 7403 K4/65 K2/35/.. FI; K4=E15,E56; K2=E52 7417 K4/99/../.. FI; K4=E4,E15 7477 K4/99/../.. K406=ASP-ADDRS-MUX/REG(RIF)				
135	4734	TEST122A3	7188	7402	BUT(D<14:00>=00D15), D=(000000), TARGET="10" 7400 K2/95 K3/5/.. FI; K2=E40,E56,E88-E89,E94-E96,E104-E105; K3=E56, E66 7403 K3/99/../.. FI; K3=E54,E95 7406 K3/99/../.. FI; K3=E74				
136	4735	TFST122A4	7202	7401	BUT(D<14:00>=00D15), D=(125252), TARGET="01" 7400 K3/99/../.. FI; K3=E54-E55,E63,E95 7403 K2/85 K3/15/.. FI; K2=E40,E56,E90; K3=E56				
137	4736	TFST121D4	7115	7413	BUTR(X#D(C)XX), D(C)=ALU07="0", D=(052525) 7417 K4/99/../.. FI; K4=E56,E63				

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRROP ##) Code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
138)	4737 TEST121D3	7100	7432	ALU=A-XOR-B; A(177777); B(125252); D(052525); BITS<05:00>=(25) 7434 K2/99/.. FI; K2=E2 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
139)	4740 TEST762C	17946	7403	BUTR(BG=SERVICE) ASSERTED; BR6 INTR PENDING, PROC PRI0=(5) 7407 K7/99/.. FI; K7=E3,E5,E11,E20,E25				
140)	4741 TEST762D	17980	7417	BUTR(BG=SERVICE+FP=SERVICE) ASSERTED; BR6 INTR PRESENT 7407 K3/80 K7/20/.. FI; K3=E50,E76; K7=E80				
141)	4742 TEST350C	8471	7434	ASPLO ADDRESSING, USING ASP/DF ADDRESS FROM (7):(0) 7400 K4/80 K3/20 K5/5 K2/5 FI; K4=E1,E5,E7,E10,E15-E16,E23,E25,E31-E34, E40-E43,E99,E110; K3=E45-E47,E59; K5=E63; K2=E55 7402 K4/65 K3/35 K6/5 FI; K4=E1,E6,E15,E17,E22,E33,E86,E92,E99-E102, E109-E112; K3=E19,E24-E26,E45,E47; K6=E40 7403 K3/99/.. FI; K3=E59 7406 K4/60 K3/40/.. FI; K4=E6; K3=E46 7417 K4/99/.. FI; K4=E43 7420 K4/99/.. FI; K4=E28,E31-E32,E102,E112 7421 K4/99/.. FI; K4=E31 7422 K4/99/.. FI; K4=E31 7424 K4/99/.. FI; K4=E31 7426 K4/99/.. FI; K4=E32 7430 K4/99/.. FI; K4=E32				
142)	4743 TEST350D	8499	7434	ASPHI ADDRESSING, USING ASP/DF ADDRESS FROM (0):(7) 7400 K4/99/.. FI; K4=E6-E7,E14,E17,E42,E99-E100,E109 7402 K4/75 K3/25/.. FI; K4=E17,E34,E99; K3=E45 7417 K4/99/.. FI; K4=E15,E109 7420 K4/99/.. FI; K4=E40-E41,E74,E92,E99,E111 7421 K4/99/.. FI; K4=E111 7422 K4/99/.. FI; K4=E111 7424 K4/99/.. FI; K4=E111 7426 K4/99/.. FI; K4=E92 7433 K4/99/.. FI; K4=E99				
143)	4744 TEST740A	17556	7402	INTR1=-PREFETCH, BC<0>=BC<0> 4747 K2/99/.. FI; K2=E24 7401 K2/99/.. FI; K2=E105 7403 K5/99/.. FI; K5=E97				
144)	4745 TEST761A	17726	7402	BUTR(SERVICE) NEGATED; AFTER CLEAR ALL SERVICE CONDITIONS, UNIBUS INIT 4747 K7/99/.. FI; K7=E57 7403 K7/99/.. K707/9=UNIBUS=UCON=INIT,K704=BG=GRANT=INTR				

Module codes: K1/DCS K2/UNWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRROP ##) Code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
145)	4750 TEST131B2	7474	7413	BUTR(X#D(C)=XX); D(C)=COUT18=0; A(0)+B(1)+C(0)=D(1)+CO(0) 7403 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE				
146)	4751 TEST762R	17924	7407	BUTR(BG=SERVICE) NEGATED; BR6 INTR PENDING, PROC PRI0=(6) 7403 K7/99/.. FI; K7=E11,E76 7407 K7/99/.. K703=BR=REQUEST=INTR				
147)	4752 TEST350A	8417	7434	BSPLO ADDRESSING, USING BSP/DF ADDRESS FROM (0):(7) 7400 K4/90 K2/10/.. FI; K4=E1,E5,E7-E9,E12,E14,E17,E21,E26,E35,E92, E99-E102,E109-E112; K2=E38,E55 7402 K4/99/.. FI; K4=E22,E100 7403 K4/99/.. FI; K4=E47 7407 K4/99/.. FI; K4=E20 7410 K4/99/.. FI; K4=E6,E13 7413 K7/99/.. FI; K7=E25 7417 K7/99/.. FI; K4=E4,E13,E15,E17,E22,E26,E109 7420 K4/99/.. FI; K4=E92,E111 7421 K4/99/.. FI; K4=E111 7422 K4/99/.. FI; K4=E111 7424 K4/99/.. FI; K4=E111 7426 K4/99/.. FI; K4=E92 7430 K4/99/.. FI; K4=E92 7431 K4/99/.. FI; K4=E92,E99,E111				
148)	4753 TEST350B	8444	7434	BSPHI ADDRESSING, USING BSP/DF ADDRESS FROM (7):(0) 7400 K4/80 K2/15 K7/5 K5/5 FI; K4=E6-E7,E11-E15,E17,E21-E22,E26,E55, E87,E94,E101,E110; K2=E32,E44,E50,E53; K7=E66; K5=E66 7417 K4/85 K5/15/.. FI; K4=E13,E15,E22; K5=E66 7420 K4/99/.. FI; K4=E102,E112				
149)	4755 TEST731A	17321	7402	BUS DATOB=RYTE=ODD, BA=(000001); DBUF=D=(000000) 4415 K2/99/.. FI; K2=E30 4747 K6/68 K2/35/.. FI; K6=E5; K2=E18 7777 K6/99/.. K605=UNIBUS=FUNCTION=DECODE				
150)	4760 TEST720C	16813	7402	DATIB=-BYTE, 16, BIT PBA, -I/O PAGE(3); SERVICE=(100340) 7400 K6/99/.. FI; K6=E59				
151)	4761 TEST763A	18062	4733	ALLOW-BG(11)H GIVEN TO BR6 INTR; BUTR (VECTOR-LOAD) ASSERTED 8888/../.. II ALSO CHECK GRANT CONTINUITY CARD II 4731 K7/95 K2/5 K6/5 K5/5 FI; K7=E4-E5,E8,E10,E12-E13,E17-E18,E20-E22, E25,E29,E33,E35,E37,E41,E51,E59,E61,E76-E77,E79, E94,E101; K2=F70; K6=E56; K5=E53 4756 K7/99/.. FI; K7=E6-E7				

Module codes: K1/DCS K2/UNWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERRR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	---	-----	-----	-----	-----	-----	-----
152)	4771	TEST761B	17765	7401	BUTR(VECTOR-LOAD) NEGATED; AFTER UNIBUS INIT	7403 K7/99	K707/9=UNIBUS-UCON-INIT,K704=BG-GRANT-INTR
					7403 K3/99	FI; K3=E44	
153)	4773	TEST762A	17829	4343	(NUA SEQUENCING LOGIC ERROR)	4747 K6/75 K5/20 K7/5 K4/5	FI; K6=E11,E19,E25,E29,E33,E49-E50; K5=E2,E10,E33,E80,E82,E97; K7=E66; K4=E48
154)	4774	TEST761C	17786	7407	BUTR(BG-SERVICE) NEGATED; AFTER UNIBUS INIT	7403 K7/99	K707/9=UNIBUS-UCON-INIT,K704=BG-GRANT-INTR
155)	4775	TEST010	2802	5245	BUTA(SUBR-B) -> BUTA(RETURN) SEQUENCE, RETURN="1010 1010 0101"=(S245)	4245 K2/85 K3/35	FI; K2=E35,E41,E61,E77,E85; K3=E4,E7,E15,E43
					4376 K2/90 K3/15	FI; K2=E33,E35,E39-E40,E48-E46,E51,E54; K3=E34	
					4777 K2/99	FI; K2=E5,E13-E14,E41,E57,E63,E82,E85-E86,E91-E92,E97,E106,E110,E116,E118	
					5005 K2/99	FI; K2=E25	
					5045 K2/85 K4/15	FI; K2=E7,E13,E25,E31,E37,E43,E49,E52,E70,E97; K4=E12-E13	
					5105 K2/99	FI; K2=E13	
					5125 K2/99	FI; K2=E13	
					5205 K2/90 K4/15	FI; K2=E4,E7,E10,E13,E16,E22,E25,E28,E71,E97; K4=E1	
					5240 K2/99	FI; K2=E26	
					5241 K2/85 K4/15	FI; K2=E4,E8,E10,E14,E16,E22,E26,E28,E62,E91; K4=E20,E86	
					5242 K2/99	FI; K2=E14	
					5244 K2/70 K4/30	FI; K2=E8,E14,E26,E58,E64,E67,E71,E73,E79,E91; K4=E12,E14	
					5245 K2/85 K4/15	FI; K2=E25,E31,E35,E37,E43,E49,E52,E58,E62,E64,E67,E73,E77,E79,E91,E97; K4=E3,E70	
					5247 K2/99	FI; K2=E8,E14,E26	
					5252 K2/99	FI; K2=E14	
					5255 K2/99	FI; K2=E8,E14,E26	
					5256 K2/99	FI; K2=E8	
					5257 K2/99	FI; K2=E14	
					5258 K2/99	FI; K2=E7,E13,E25	
					5345 K2/99	FI; K2=E7,E13,E25	
					5365 K2/99	FI; K2=E7,E13	
					5376 K2/99	FI; K2=E40	
					7245 K2/99	FI; K2=E35,E41,E77	
					7247 K2/99	FI; K2=E40	
					7273 K3/85 K2/45	FI; K3=E4,E6-E7,E11,E15,E43; K2=E61,E77,E85	
					7401 K2/99	FI; K2=E41	
					7777 K2/99	FI; K2=E11	

Module codes: K1/DCS K2/UNWORD K3/IPDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERRR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	---	-----	-----	-----	-----	-----	-----
156)	5146	TEST005	2711	4474	NUA SEQUENCING, PAGE (3) -> (4), USF=(34)	4574 K3/50 K2/50	FI; K3=E58; K2=E7
					5441 K3/60 K2/40	FI; K3=E10,E18,E20; K2=E30,E54	
157)	5463	TEST722A	16951	4777	INVALIDATE, ODD ADDR JAM, BA=(140001)	7341 K6/75 K4/15 K3/15	FI; K6=E71,E76,E81,E88,E96; K4=E51; K3=E11
					7777 K6/99	K605=UNIBUS-FUNCTION-DECODE	
158)	5465	TEST720A	16730	4777	DATIB=BYTE, ODD ADDR JAM, BA=(080001)	7341 K5/80 K3/50	FI; K5=E67,E94; K3=E34
					7777 K6/99	K605=UNIBUS-FUNCTION-DECODE	
159)	5467	TEST713A	16594	4777	DATI=NOINT, ILLEGAL INTERNAL ADDR JAM, BA=(177776)	7777 K6/99	K605=UNIBUS-FUNCTION-DECODE,INTERNAL-ADDR-DETECT
160)	5471	TEST712A	16460	4777	DATI, INTERNAL ADDRESS JAM, BA=(177776)	7777 K6/99	K605=UNIBUS-FUNCTION-DECODE,INTERNAL-ADDR-DETECT
161)	5473	TEST711A	16328	4777	DATOB=BYTE, 88YN TIMEOUT JAM, BA=(160001)	7341 K6/70 K4/20 K7/5 K5/5	FI; K6=E4,E11,E28,E62,E64,E78-E79,F98-E99,E101-E102; K4=E18,E28; K7=E60; K5=E55
					7777 K6/99	K605=UNIBUS-FUNCTION-DECODE	
162)	5475	TEST710A	16157	4777	DATO, ODD ADDRESS ERROR JAM, BA=(160001)	4747 K6/99	FI; K6=E60
					7341 K6/75 K7/15 K4/10 K5/5	FI; K3/5 FI; K6=E2,E4,E10,E38,E40,E42-E43,E46-E47,E54-E55,E60,E64,E69,E71-E72,E76,E79,E81,E89,E98-E99,E103,E108; K7=E22,E25,E39,E43,E45-E46,E65; K4=E9,E18,E26,E30,E71; K5=E44,E64; K3=E2	
					7401 K6/99	FI; K5=E97	
					7445 K6/99	FI; K6=E43	
					7777 K6/99	K605=UNIBUS-FUNCTION-DECODE	
163)	5477	TEST702A	16040	7402	LOAD BA(1716)=01, 18, BIT MODE, READ THRU STATUS=MUX(SERVICE)<918>	4747 K2/99	FI; K2=E34
					7400 K6/40 K5/25 K4/20 K7/10 K2/5	FI; K6=E11,E51-E54,E59; K5=E4,E13,E15,E26,E54,E84,E97; K4=E20,E55; K7=E48,E56; K2=E61	
164)	5400	TEST374D2	10178	7434	A/R SP WRITE FCN WR(B,HI,A-ADDR) AND WR(B,HI,B-ADDR)	7477 K4/99	K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS
165)	5502	TEST374F2	10251	7434	A/B SP WRITE FCN WR(AB,LO,A-ADDR) AND WR(AB,LO,B-ADDR)	7400 K2/99	FI; K2=E117

[Continued]

Module codes: K1/DCS K2/UNWORD K3/IRDFCODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	FPROR code	Symbolic label	Line number	ENUA	TNUA	-->Module sequence-->			Test summary - Print reference - Chip information
---	---	---	---	---	---	01/00	02/00	03/00	-----
	5502	TEST374E2	[Continued]		A/B SP WRITE FCN WR(AB,LO,A=ADDR) AND WR(AB,LO,B=ADDR)	7420	K4/99	../..	FI; K4=E10
					7477 K4/99	../..	../..	../..	K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS
166)	5504	TEST374F2	10324	7434	A/B SP WRITE FCN WR(AB,HI,A=ADDR) AND WR(AB,HI,B=ADDR)	7477	K4/99	../..	K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS
167)	5505	TEST712B	16512	7402	DATI, INTERNAL ADDR; JAN=(001000)	7400	K6/70	K7/15	K3/18
									FI; K6=E46,E49-E51,E81,E99; K7=E19,E58,E61,E81; K3=E48,E69; K5=E13,E72
					7401 K6/99	../..	../..	../..	FI; K6=E76
					7777 K6/99	../..	../..	../..	K608=INTERNAL-ADDR-DETECT
168)	5507	TEST410	10666	NONE	(NUA SEQUENCING LOGIC/DCS ERROR)	7777	K1/99	../..	../..
									INTERNAL DCS ERROR
169)	5511	TEST620A	15262	7406	BUTH(INIT-JAM) NEGATED="0" AFTER CLR-JAM-ERRORS UCON	7407	K7/90	K2/10	../..
									FI; K7=E18,E41,E81; K2=E112
170)	5512	TEST701D	16011	7402	BUTR(DIC18BA00), BA<00>="0"	7403	K3/99	../..	../..
									FI; K3=E64
171)	5513	TEST376A	10595	5440	BUTA(R-IGN-1) DOES NOT CAUSE A BRANCH	5441	K3/99	../..	../..
									FI; K3=E64
172)	5517	TEST379A	10512	7432	DAD/3 CAUSES BYTE-WRITE(LO) TO ASPLO; DAD/1 OR DAD/2 NOT	7400	K3/80	K4/20	../..
									FI; K3=E17,E20,E36,E38,E46; K4=E5,E99
					7402 K3/95	K4/15	../..	../..	FI; K3=E27,E36,E38,E46,E49,E59-E60; K4=E5,E33
					7433 K4/99	../..	../..	../..	FI; K4=E28
					7434 K3/99	../..	../..	../..	FI; K3=E59
173)	5521	TEST102A	5225	7432	CSP ADDRESS, LOC(02); EMIT/CSP/ALU-B/D/DBUF/IR WITH DATA=(000125)	7400	K4/60	K3/40	../..
									FI; K4=E47,E55,E68-E69,E95-E96,E98,E105,E108; K3=E8,E9-E10,E14,E32,E35,E42,E92
					7402 K4/70	K3/30	../..	../..	FI; K4=E85,E98,E98; K3=E12-E13,E32,E35,E42
					7403 K4/99	../..	../..	../..	FI; K4=E96
					7410 K3/95	K4/25	../..	../..	FI; K3=E1-E3,E8,E13-E14,E42; K4=E1,E30
					7417 K4/99	../..	../..	../..	FI; K4=E105,E108
					7420 K4/70	K6/10	K2/10	K3/5	K7/5
									FI; K4=E1,E4,E7-E8,E11-E12,E31-E32,E38,E40-E41,E52,E56,E65-E66,E72,E74,E79,E88-E92,E96,E100-E102,E105-E106,E109-E112; K6=E65,E67,E73,E75,E108; K2=E83,E94,E98,E100,E102-E104,E107-E108,E111,E113-E114; K3=E72; K7=E85-E86; K5=E63,E66
					7421 K4/99	../..	../..	../..	FI; K4=E87
					7427 K6/55	K4/30	K7/15	../..	FI; K6=E73; K4=E89; K7=E37
					7433 K4/90	K3/15	../..	../..	FI; K4=E89,E105; K3=E8
					7434 K4/75	K3/10	K6/10	K7/5	K2/5
									FI; K4=E1-E5,E7,E9-E11,E16,E20-E26,

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	FPROR code	Symbolic label	Line number	ENUA	TNUA	-->Module sequence-->			Test summary - Print reference - Chip information
---	---	---	---	---	---	01/00	02/00	03/00	-----
	5521	TEST102A	[Continued]		CSP ADDRESS, LOC(02); EMIT/CSP/ALU-B/D/DBUF/IR WITH DATA=(000125)	7434	K4/75	K3/10	K6/10
									E28-E30,E38,E47,E51,E55-E56,E59,E64,E66,E70-E71,E74,E87-E89,E96,E98-E99,E108; K3=E2-E3,E12,E35,E42-E43,E53,E59,E82,E84; K6=E31,E34,E42,E82,E103; K7=E76; K2=E102
174)	5523	TEST701B	15962	7403	BUTR(DIC18BA00), BA<00>="1"	7402	K3/99	../..	../..
									K308=MICROBRANCH
175)	5524	TEST374C2	10105	7434	A/B SP WRITE FCN WR(B,LO,A=ADDR) AND WR(B,LO,B=ADDR)	7400	K3/99	../..	../..
									FI; K3=E45
					7402 K4/99	../..	../..	../..	FI; K4=E10,E17,E21-E22
					7420 K3/99	../..	../..	../..	FI; K3=E93
					7421 K4/99	../..	../..	../..	FI; K4=E21-E22
					7477 K4/99	../..	../..	../..	K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS
176)	5525	TEST004	2692	5146	NUA SEQUENCING, NO "BUT"	5147	K3/99	../..	../..
									FI; K3=E38
					5156 K3/99	../..	../..	../..	FI; K3=E37
					5346 K3/99	../..	../..	../..	FI; K3=E23
177)	5531	TEST103A	5370	7412	CSP ADDRESS, LOC(13); EMIT/CSP/ALU-B/D/DBUF/IR WITH DATA=(125200)	7400	K4/99	../..	../..
									FI; K4=E59,E66,E74,E92,E102,E109-E112
178)	5533	TEST610B1	14869	7413	PS[CC]=(13); IR=(105300), CC=NON-ADDR(132)	7401	K3/99	../..	../..
									FI; K3=E55
					7403 K3/99	../..	../..	../..	FI; K3=E63,E72
					7411 K3/90	K4/10	../..	../..	FI; K3=E55,E62,E65,E85; K4=E72
					7417 K3/60	K2/40	../..	../..	FI; K3=E55,E72; K2=E105
179)	5535	TEST610C1	14987	7407	PS[CC]=(07); IR=(072000), CC=NON-ADDR(437)	7405	K3/99	../..	../..
									FI; K3=E33
					7406 K3/99	../..	../..	../..	FI; K3=E62
					7415 K2/99	../..	../..	../..	FI; K2=E8
180)	5541	TEST105A	5592	7432	SR LOAD/READ, SR=(000125), RES=SR/LOAD; ALU-A/D/DBUF/IR PATH	7400	K4/85	K3/10	K2/5
									FI; K4=E1,E3,E11,E16,E18,E20,E27,E30,E35-E36,E45-E47,E52,E80; K3=E5,E10,E14,E51,E66; K2=E89
					7402 K4/85	K3/10	K2/5	../..	FI; K4=E30,E35,E48-E47; K3=E8,E14; K2=E101
					7403 K4/50	K3/50	../..	../..	FI; K4=E80; K3=E9
					7406 K3/99	../..	../..	../..	FI; K3=E9
					7410 K4/99	../..	../..	../..	FI; K4=E36,E52
					7412 K3/99	../..	../..	../..	FI; K3=E8
					7417 K4/99	../..	../..	../..	FI; K4=E27,E36,E47,E52
					7420 K4/90	K3/10	../..	../..	FI; K4=E1,E3,E7,E11,E16,E20,E26,E31-E32,E34-E35,E37,E40-E42,E44,E47-E48,E52-E54,E57,E60-E62,E67-E68,E78-E76,E80; K3=E41,E43,E51,E67,E71,E75,E82

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	Module sequence-> #1/% #2/% #3/%	Test summary - Print reference - Chip information
---	---	---	---	---	---	---	-----
	5541	TEST105A	[Continued]			SR LOAD/READ, SR=(000125), RES=SR/LOAD; ALU=A/D/DBUF/IR PATH 7427 K4/99/.. FI; K4=E1,E7,E16,E33,E35 7430 K4/99/.. FI; K4=E1 7431 K4/99/.. FI; K4=E36,E45 7433 K4/99/.. FI; K4=E35,E37,E47,E66 7434 K4/99 K3/5/.. FI; K4=E1,E4,E7,E11,E16,E30-E32,E35-E36,E39,E43, E45,E47-E48,E82-E83,E55,E57,E65,E80; K3=E102	
181)	5545	TEST610P1	15101	7410		PS[CC]=10; IR=(072000), CC=ROM=ADDR(216) 7412 K3/99/.. FI; K3=E33,E62	
182)	5547	TEST104A	5516	7425		CSP BASCON ADDR, LOC(16); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152) 7400 K4/99/.. FI; K4=E38,E96 7417 K4/99/.. FI; K4=E38 7420 K4/99/.. FI; K4=E74,E111-E112 7434 K3/60 K4/35 K2/5 FI; K3=E1-E3,E12-E13; K4=E1,E30,E38; K2=E70	
183)	5551	TEST050A	5077	7405		BUT(INSTR5), IR=(172500); E88(725)=(08) 7477 K3/99/.. K305=INSTR5-DECODE	
184)	5553	TEST101A	5161	7434		ALU=ZERO, D/DBUF/IR PATH; BUT(INSTR5) FOR IR=ZERO 7400 K4/65 K6/20 K3/10 K7/5 K2/5 FI; K4=E33-E34,E42-E43,E51,E59,E82, E95,E100-E101,E109-E110; K6=E73,E82,E91,E96; K3=E1,E43,E51,E71,E82; K7=E52,E54; K2=E70 7402 K4/80 K6/15 K3/5 FI; K4=E33-E34,E51,E82,E95,E100-E101; K6=E91; K3=E13 7410 K3/99/.. FI; K3=E43,E82,E102 7411 K6/99/.. FI; K6=E65,E82 7412 K7/65 K2/30 K3/5 FI; K7=E28,E41,E52,E54,E70; K2=E61,E83; K3=E43 7417 K6/99/.. FI; K6=E82 7420 K4/75 K6/15 K7/5 K6/5 K3/5 K2/5 FI; K4=E31-E32,E38,E40-E41, E47,E55,E66,E74,E79,E92,E102,E111-E112; K6=E65, E73; K7=E41,E48,E65,E70; K5=E14,E54; K3=E82,E84; K2=E34,E85 7421 K4/90 K6/15/.. FI; K4=E31,E40,E47,E72,E74,E111-E112; K6=E65 7422 K7/75 K4/20 K6/5 FI; K7=E6-E7,E15,E23,E31,E45,E48,E65-E67,E69,E74, E78,E91-E92; K4=E31,E40,E72,E74,E111-E112; K6=E43,E65 7424 K4/80 K6/20/.. FI; K4=E31,E40,E74,E79,E111-E112; K6=E65 7426 K6/80 K3/50/.. FI; K6=E73; K3=E51,E82 7427 K4/99/.. FI; K4=E74 7431 K3/99/.. FI; K3=E43 7477 K4/90 K3/10/.. K404=ALU/CARRY-LOOKAHEAD; K313=ALU-FCN-DECODE	
185)	5554	TEST102D	5330	7425		CSP ADDRESS, LOC(01); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152) 7400 K4/99/.. FI; K4=E4 7402 K5/99/.. FI; K5=E44 7420 K4/60 K6/10 K3/5 K2/5 K5/5 K7/5 FI; K4=E35,E52,E65-E66,E72,	

Module codes: K1/DCS K2/UNORD K3/INDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	FPOR code	Symbolic label	Line number	ENUA	TNUA	Module sequence-> #1/% #2/% #3/%	Test summary - Print reference - Chip information
---	---	---	---	---	---	---	-----
	5554	TEST102D	[Continued]			CSP ADDRESS, LOC(01); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152) 7420 K4/60 K6/10 K3/5 E74,E79,E88-E92,E102,E106,E111-E112; K6=E65,E67, E73,E75; K3=E72; K2=E94,E104; K5=E66,E106; K7=E85-E86 7434 K4/99 ../.. ../.. FI; K4=E38,E70,E88-E89,E98,E108	
186)	5561	TEST047A	4988	7412		BUT(INSTR5), IR=(122566); E88(325)=(12) 7400 K3/99 ../.. ../.. FI; K3=E88 7405 K3/99 ../.. ../.. FI; K3=E88 7413 K3/99 ../.. ../.. FI; K3=E88 7416 K3/99 ../.. ../.. FI; K3=E88	
187)	5563	TEST701A	15925	7402		LOAD BA<15:00>=(052525), READ THRU BA/KT-ALU/PBA/STATUS-MUX(PBA) 7400 K5/55 K6/15 K7/15 K4/15 K3/5 FI; K5=E3,E7,E9-E11,E13,E25,E27,E30, E32,E37-E39,E43,E45-E47,E50,E58-E60,E65-E67, E69-E70,E72,E79-E82,E93-E94; K6=E19,E25,E29,E33, E36,E38,E49,E54,E59,E70,E76,E90,E96,E104,E108; K7=E48,E56,E64-E65,E73,E80-E82,E88; K4=E29,E39, E48,E57,E65,E87; K3=E9,E14,E33,E64 7401 K4/45 K3/35 K6/20 FI; K4=E57,E65,E87; K3=E5,E8,E10,E33; K6=E31 7403 K7/50 K4/50 ../.. FI; K7=E80; K4=E39	
188)	5564	TEST102C	5303	7412		CSP ADDRESS, LOC(04); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(125200) 7400 K4/65 K6/15 K2/10 K5/5 K3/5 FI; K4=E35-E36,E45,E51,E56,E59,E66, E72,E79-E80,E82,E89,E91-E92,E95,E98,E100-E102, E105,E108-E110; K6=E73,E75,E82,E84,E91,E93; K2=E95-E96,E104; K5=E53,E63; K3=E56,E95 7403 K4/75 K6/25 ../.. FI; K4=E48,E51,E64; K6=E91 7405 K4/99 ../.. ../.. FI; K4=E98 7420 K3/99 ../.. ../.. FI; K3=E17,E20,E27,E36 7432 K3/99 ../.. ../.. FI; K3=E32 7434 K4/75 K2/25 ../.. FI; K4=E29,E38,E70,E80-E89,E98,E108; K2=E5,E11,E17, E23,E29,E94	
189)	5565	TEST624R	15817	7434		CLOCKING D REGISTER PROPAGATED THRU MICROBREAK JAM (P2-T) 7477 F4/99 ../.. ../.. K403/4=U-CLKD-CNTRL	
190)	5571	TEST624A	15724	4777		MICROBREAK JAM AT MICROADDRESS=(6255); FLAG<8> CLEARED IN JAM WORD 7777 K3/99 ../.. ../.. K311=MICROBREAK-LOGIC	
191)	5572	TEST103R	5423	7432		CSP ADDRESS, LOC(15); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000125) 7411 K2/80 K3/20 ../.. FI; K2=E2,E8-E9,E21,E27,E60,E66,E69,E75,E81; K3=E21-E22,E76 7415 K3/99 ../.. ../.. FI; K3=E21 7420 K4/99 ../.. ../.. FI; K4=E74,E111-E112	
192)	5573	TEST623	15680	7402		BUT(CUA-TRACK) RESETS CUA TO TRACKING NUA VALUE 4747 K3/99 ../.. ../.. FI; K3=E91	

[Continued]

(Continued)

Module codes: K1/DCS K2/UNORD K3/INDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	----	-----	-----	-----	-----	-----	-----
	5573	TEST623	[Continued]			BUT(CUA-TRACK) RESETS CUA TO TRACKING NUA VALUE 7400 K2/99/..	FI; K2=E3,E6,E9,E18,E24-E25,E30,E33,E39,E45,E51,E54,E60,E66,E75,E81,E86-E87,E113
193)	5574	TFST102B	5277	7405		CSP ADDRESS, LOC(10); EMIT/CSP/ALU=B/D/DRUF/IR WITH DATA=(152500) 7400 K4/65 K6/30 K2/5 K5/5 FI; K4=E36,E38,E42-E43,E51,E59,E82,E89,E98, E105,E108-E110; K6=E82,E84,E91; K2=E96; K5=E63 7402 K4/65 K6/15 K5/10 K2/10 K3/5 FI; K4=E45,E51,E64,E78,E95,E98, E100-E101; K6=E91,E93; K5=E53; K2=E95; K3=E95 7403 K2/99/.. FI; K2=E34 7412 K4/99/.. FI; K4=E89,E92,E102,E106 7434 K4/90 K7/5 K3/5 K2/5 FI; K4=E4,E38,E70,E88-E89,E98-E99,E108; K7=E21; K3=E102; K2=E84	
194)	5575	TEST622C	15657	7402		CLR-JAM-ERRORS RESETS STATUS-MUX(JAM)=(001000) AFTER UBREAK JAMUPP 7407 K7/99/.. K705=UBRK-JAM-FLAGS,K707=JAM-CLEAR,K708=STATUS-MUX	
195)	5576	TFST103C	5449	7425		CSP ADDRESS, LOC(16); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(000152) 7402 K4/99/.. FI; K4=E100-E101 7420 K3/99/.. FI; K3=E93	
196)	5577	TEST622B	15634	7401		BUTR(OTHER-JAM) NEGATED=0" AFTER CLR-JAM-ERRORS 7403 K7/99/.. FI; K7=E51-E52,E57	
197)	5600	TEST105B1	5666	7412		BUT(SR3=0), SR=(000152); SR<310>="1010" 7410 K4/99/.. FI; K4=E44 7413 K3/99/.. FI; K3=E67 7416 K3/97/.. FI; K3=E75	
198)	5601	TEST622A	15605	7403		UCONS 'START-DELAY' & 'CLR-NPR-TIMEOUT' DONT EFFECT 'CLR-JAM-ERRORS' 7401 K7/99/.. FI; K7=E36	
199)	5602	TEST610C2	15038	7401		PS(CC)=(01); IR=(072000); CC-ROM-ADDR(037) 7403 K3/99/.. FI; K3=E62 7405 K3/99/.. FI; K3=E55,E66	
200)	5603	TEST621F	15521	7402		PROC-MUX(CUA-PORT)=(055226); CUA-LOCKED, EXFLAGS SET, PREFETCH=JAM CLEAR 7400 K2/99/.. FI; K2=E46,E58,E90,E100,E105	
201)	5604	TEST374A2	9959	7434		A/B SP WRITE FCN WR(A,LO,A=ADDR) AND WR(A,LO,B=ADDR) 7400 K4/80 K2/20/.. FI; K4=E11,E17,E23,E47; K2=E34 7402 K4/75 K3/25/.. FI; K4=E17,E26; K3=E93 7420 K4/70 K2/30/.. FI; K4=E10,E17,E23-E24,E26; K2=E32,E38,E44,E50,E53 7477 K4/99/.. K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS	
202)	5605	TEST621E	15495	7402		STATUS-MUX(JAM-PORT)=(001001) AFTER MICROBREAK JAM 7400 K2/75 K7/25/.. FI; K2=E56,E76,E83,E100; K7=E56,E73	

Module codes: K1/PCS K2/WORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	----	-----	-----	-----	-----	-----	-----
203)	5606	TEST374A2	10032	7434		A/B SP WRITE FCN WR(A,HI,A=ADDR) AND WR(A,HI,B=ADDR) 7477 K4/99/.. K405=SP-REWRITE-CNTL,K406/7=A/B-SPADS	
204)	5607	TEST621D	15476	7403		BUTR(OTHER-JAM) ASSERTED="1" AFTER MICROBREAK JAM 7401 K7/70 K3/30/.. FI; K7=E32,E38,E57,E73; K3=E44	
205)	5610	TEST105A1	5619	7405		BUT(SR3=0), SR=(000125); SR<310>="0101" 7407 K3/99/.. FI; K3=E77 7415 K3/99/.. FI; K3=E76 7425 K3/99/.. FI; K3=E68	
206)	5611	TEST621C	15458	7406		BUTR(INIT-JAM) STAYS NEGATED AFTER MICROBREAK JAM 7407 K2/99/.. FI; K2=E46	
207)	5612	TEST610P2	14919	7406		PS(CC)=(06); IR=(005300); CC-ROM-ADDR(253) 7402 K3/99/.. FI; K3=E55,E63,E66 7404 K3/85 K4/45/.. FI; K3=E55,E65; K4=E78 7414 K3/99/.. FI; K3=E55	
208)	5613	TFST621A	15440	7402		CLOCKING D-REGISTER PROPAGATED THRU MICROBREAK JAM (P3=T) 7477 K4/99/.. K403/4=U-CLKD-CNTL	
209)	5614	TFST376A1	10619	7434		SP/SP-ADDRESS: FLTPT/BIT02="0"; R-IDR-1/BIT00="1" FORCED 7400 K3/40 K2/40 K4/25 FI; K3=E50,E57,E94,E97; K2=E55,E108,E112; K4=E4,E9, E34,E101 7410 K2/65 K3/35/.. FI; K2=E95,E118; K3=E47,E96	
210)	5615	ERROR621A	15426	0000		MICROBREAK JAMUPP AT (5522) ATTEMPTED; DID NOT OCCUR 5615 K3/75 K2/10 K5/10 K6/5 K7/5 FI; K3=E32,E41,E52,E61,E91,E91,E101, E111-E112; K2=E35,E41,E70,E77; K5=E60-E61,E68, E77; K6=E39-E40; K7=E48,E57	
211)	5616	TEST713B	16640	7402		DATI=NOINT, ILLEGAL INTERNAL ADDR; JAM=(001040) 7400 K6/70 K7/15 K5/15 FI; K6=E46,E51; K7=E74,E88; K5=E38,E72	
212)	5617	TEST621A	15344	4777		MICROBREAK JAM AT MICROADDRESS=(5522); ACTIVE-BUT, WR-CSP LATCHES CLEARED 4747 K3/99/.. FI; K3=E91,E111	
213)	5620	TEST105E	5743	7415		SR LOAD/READ, SR=(152500); SR<1512>="1101" 7405 K5/99/.. FI; K5=E60 7411 K4/75 K6/10 K5/5 K2/5 FI; K4=E33-E34,E39,E45-E46,E51,E54,E78,E95, E98,E100-E101; K6=E91,E93; K5=E53; K2=E95 7414 K4/85 K6/10 K5/5 K2/5 FI; K4=E33-E34,E36,E39,E45-E46,E51,E62,E80, E82,E95,E98,E100-E101; K6=E91,E93; K5=E53; K2=E95	

Module codes: K1/PCS K2/WORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRORP	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	---	-----	-----	----	----	-----	-----
214)	5621	TF8T620C	15309	7402		STATUS-MUX(JAM-PORT)=001000 AFTER CLR-JAM-ERRORS UCON 7400/../../.. I1 MOS BATTERY BACKUP MUST BE GOOD I1 7400 K7/95 K2/5 K6/5 FI; K7=E48-E49,E52,E56-E58,E64-E66,E73,E80-E82, E88; K2=E37,E64,E66,E70; K6=E51 7401 K7/95 K2/15/.. FI; K7=E28,E37; K2=E100 7403 K7/99/../.. FI; K7=E80	
215)	5622	TF8T610A2	14806	7406		PS(CC)=(06); IR=(105200); CC=ROM-ADDR(145) 7404 K3/90 K4/10/.. FI; K3=E55,E62,E65,E72,E85; K4=E72 7407 K3/99/../.. FI; K3=E62,E72	
216)	5623	TF8T620R	15286	7401		RUTR(OTHER-JAM) NEGATED=0 AFTER CLR-JAM-ERRORS UCON 7403 K7/80 K6/20 K3/5 FS; E7=E18,E26,E32,E38,E49,E57-E58,E74; K6=E2-E3, E62,E64,E105; K3=E64	
217)	5624	TF8T610D2	15145	7410		PS(CC)=(10); IR=(072000); CC=ROM-ADDR(116) 7414 K3/99/../.. FI; K3=E63,E66	
218)	5625	TF8T3758	10563	7432		DAD/3 CAUSES BYTE-WRITE(LO) TO BSPLO; DAD/1 OR DAD/2 NOT 7400 K3/99/../.. FI; K3=E46 7402 K4/99/../.. FI; K4=E99-E100	
219)	5627	TEST610A1	14744	7405		PS(CC)=(05); IR=(108200); CC=ROM-ADDR(665) 7400 K3/99/../.. FI; K3=E57 7401 K3/70 K2/30/.. FI; K3=E55,E72; K2=E105 7404 K3/99/../.. FI; K3=E62,E72 7406 K3/99/../.. FI; K3=E62,E68,E85 7407 K3/80 K4/20/.. FI; K3=E34,E55,E62,E65,E72; K4=E72 7411 K3/99/../.. FI; K3=E34,E55,E65 7412 K3/99/../.. FI; K3=E5,E8-E10,E14,E36,E72 7415 K3/99/../.. FI; K3=E55,E62-E63,E72	
220)	5631	TEST046A	4920	7426		BUT(INSTR5), IR=(000200); E78(200)=(26) 6000 K2/99/../.. FI; K2=E47 7426 K2/99/../.. FI; K2=E25 7427 K3/99/../.. K305=INSTR5-DECODE	
221)	5633	TF8T045A	4878	7405		BUT(DM0#SM0#BYTE), IR=(004300); DM0, -SM0, BYTE (SWAB) 7404 K3/99/../.. FI; K3=E85	
222)	5635	TEST044A	4833	7447		BUT(INSTR1), IR=(076250); CLASS=A(XPC) 7417 K3/99/../.. FI; K3=E90,E99 7464 K3/99/../.. FI; K3=E90,E110	

Module codes: K1/DCS K2/UNWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRORP	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	---	-----	-----	----	----	-----	-----
223)	5636	TEST105D	5716	7405		SR LOAD/READ, SR=(152500); ALU-A/D/DBUF/IR PATH 7400 K4/99/../.. FI; K4=E27,E38-E36,E39,E42-E43,E48,E67 7402 K2/95 K4/5/.. FI; K2=E6,E9,E12,E18,E26,E30,E33,E39,E45,E51,E54, E60,E66,E69,E75,E86-E88,E94,E99,E103; K4=E77	
224)	5637	TEST043B	4796	7711		BUT(INSTR1), IR=(161707); CLASS=G(DOP#-SM0) 5126 K6/99/../.. FI; K6=E42 7710 K3/99/../.. FI; K3=E108 7717 K3/99/../.. FI; K3=E107 7750 K3/99/../.. FI; K3=E90	
225)	5641	TF8T043A	4767	7404		BUT(DM0#SM0#BYTE), IR=(161707); DM0, -SM0, -BYTE (SUB) 7400 K3/99/../.. FI; K3=E74 7406 K3/99/../.. FI; K3=E109	
226)	5642	TEST105C	5690	7412		SR LOAD/READ, SR=(125200); ALU-A/D/DBUF/IR PATH 7400 K4/99/../.. FI; K4=E27,E32-E37,E39,E41-E43,E45-E46,E48,E60,E62, E67-E68,E72 7403 K4/99/../.. FI; K4=E33-E34,E39,E45-E46,E54,E64,E77-E78 7405 K4/99/../.. FI; K4=E45-E46 7420 K4/99/../.. FI; K4=E28 7434 K4/99/../.. FI; K4=E29	
227)	5643	TEST042B	4731	7714		BUT(INSTR1), IR=(144020); CLASS=G(DOP#-SM0) 7417 K3/99/../.. FI; K3=E90,E110 7514 K3/99/../.. FI; K3=E99 7614 K3/99/../.. FI; K3=E100 7710 K3/99/../.. FI; K3=E106,E120 7715 K2/99/../.. FI; K2=E77 7717 K3/99/../.. FI; K3=E97,E105-E107	
228)	5645	TF8T042A	4703	7401		BUT(DM0#SM0#BYTE), IR=(144020); -DM0, -SM0, BYTE 7400 K3/70 K2/30/.. FI; K3=E54,E60,E65,E85; K2=E19,E41,E77-E78 7403 K5/50 K3/80/.. FI; K5=E98; K3=E109 7406 K2/99/../.. FI; K2=E78	
229)	5646	TEST104R	5543	7432		CSP BASCON ADDR, LOC(18); EMIT/CSP/ALU-B/D/DBUF/IR WITH DATA=(000125) 7400 K4/99/../.. FI; K4=E38 7402 K4/99/../.. FI; K4=E51,E100-E101	
230)	5647	TF8T041R	4667	7517		BUT(INSTR1), IR=(120777); CLASS=B(DOP#-MOV#SM0#-DM0) 7617 K3/99/../.. FI; K3=E118	
231)	5651	TEST041A	4638	7412		BUT(IR15-12), IR=(120777); IR<15:12>="1010" 7042 K2/99/../.. FI; K2=E1	

Module codes: K1/DCS K2/UNWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
232)	5652	TEST105R	5642	7425	SR LOAD/READ, SR=(000152);	7420	K4/70	K3/30	FI; K4=E31-E32,E35,E37,E40-E41,E44,E52-E53,E57,E61,E75-E76; K3=E43,E51,E74,E76-E77,E82,E102
						7432	K4/99	././.	FI; K4=E55
233)	5653	TEST040R	4602	7517	BUT(INSTR1), IR=(050777), CLASS=B(SOP=MOV=SMO=DM0)	7417	K3/99	././.	FI; K3=E80
234)	5655	TEST040A	4574	7405	BUT(IR15=12), IR=(050777); IR<15:12>="0101"	7401	K3/99	././.	FI; K3=E121
235)	5657	TEST037A	4531	7446	BUT(INSTR1), IR=(060205), CLASS=A(DOP=SMO=DM0)	7417	K7/35	K3/35	FI; K7=E17; K3=E115; K2=E73
						7447	K3/99	././.	FI; K3=E94
236)	5661	TEST036A	4486	7455	BUT(INSTR1), IR=(150506), CLASS=A(DOP=SMO=DM0)	7457	K3/99	././.	FI; K3=E94
237)	5663	TEST035A	4442	7443	BUT(INSTR1), IR=(030701), CLASS=A(DOP=SMO=DM0)	7417	K3/99	././.	FI; K3=E60,E80,E98,E90,E95,E97,E105,E107,E110,E115-E117,E120
						7441	K3/45	K6/30	FI; K3=E43,E94,E107; K6=E91-E92; K5=E67,E98
						7442	K3/99	././.	FI; K3=E94,E108
						7460	K3/99	././.	FI; K3=E90
						7540	K3/99	././.	FI; K3=E120
238)	5665	TEST034A	4398	7511	BUT(INSTR1), IR=(005112), CLASS=B(SOP=DM0)	7517	K3/99	././.	FI; K3=E98
						7751	K3/99	././.	FI; K3=E98
239)	5667	TEST033A	4354	7517	BUT(INSTR1), IR=(106274), CLASS=B(SOP=DM0)	7417	K3/99	././.	FI; K3=E80,E100,E108,E120
						7515	K2/99	././.	FI; K2=E114
						7562	K3/99	././.	FI; K3=E108
						7716	K3/99	././.	FI; K3=E100
240)	5670	TEST047R	5019	7413	BUT(IR11=FLOAT), IR=(122566); FP-ROM(534)=(13)	7477	K3/99	././.	K105=FP-INSTR-DECODE
241)	5671	TEST032R	4317	7612	BUT(INSTR1), IR=(110125), CLASS=C(MOV=SMO=DM0)	7451	K3/99	././.	FI; K3=E109
						7602	K3/99	././.	FI; K3=E4,E6-E7,E11,E15,E43,E118
						7752	K3/99	././.	FI; K3=E85,E94,E104

Module codes: K1/DCS K2/UMOPD K3/IRDECODE K4/DATAPATH K5/RTCACHE K6/TIMING K7/STATUS

##)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
242)	5673	TEST032A	4288	7412	BUT(MOV/DR7;IR5=3), IR=(110125); IR<5:3>="010", MOV	7402	K3/80	K5/20	FI; K3=E57,E97; K5=E68
243)	5674	TEST103D	5476	7405	CSP ADDRESS, LOC(07); EMIT/CSP/ALU=B/D/DBUF/IR WITH DATA=(157500)	7322	K3/99	././.	FI; K3=E22
						7400	K4/99	././.	FI; K4=E59,E109-E110
244)	5675	TEST031R	4253	7604	BUT(INSTR1), IR=(010242), CLASS=C(MOV=SMO=DM0)	7404	K3/99	././.	FI; K3=E99
						7417	K3/90	K5/15	FI; K3=E87,E90,E113,E119; K5=E98
						7514	K3/55	K6/30	FI; K3=E115-E117; K6=E91-E92; K5=E67
						7607	K3/99	././.	FI; K3=E98
						7614	K3/99	././.	FI; K3=E97,E118
						7640	K3/99	././.	FI; K3=E90
						7700	K3/99	././.	FI; K3=E99,E110
						7704	K3/99	././.	FI; K3=E98,E120
245)	5677	TEST031A	4224	7402	BUT(DM0=SMO=BYTE), IR=(010242); -DM0, SMO, -BYTE	7005	K3/99	././.	FI; K3=E56
						7400	K3/99	././.	FI; K3=E47,E56,E60,E80,E90,E109,E119-E120
						7406	K3/99	././.	FI; K3=E109
246)	5701	TEST030A	4117	7552	BUT(INSTR1), IR=(005204), CLASS=D(SOP=DM0)	7313	K3/99	././.	FI; K3=E2
						7553	K3/99	././.	FI; K3=E87
247)	5703	TEST027A	4074	7561	BUT(INSTR1), IR=(106102), CLASS=D(SOP=DM0)	7777	K3/99	././.	K304/8=INSTR1-DECODE
248)	5705	TEST026A	4029	7574	BUT(INSTR1), IR=(105403), CLASS=D(SOP=DM0)	7570	K3/99	././.	FI; K3=E108
						7574	K2/99	././.	FI; K2=E67
						7757	K6/99	././.	FI; K6=E82
249)	5707	TEST025A	3986	7543	BUT(INSTR1), IR=(006303), CLASS=D(SOP=DM0)	7540	K3/99	././.	FI; K3=E99
						7557	K3/99	././.	FI; K3=E105
250)	5710	TEST014D	3436	7401	BUT(IR11=FLOAT), IR=(000125); FP-ROM(020)=(01)	7400	K3/99	././.	FI; K3=E86
251)	5711	TEST024B	3949	7557	BUT(INSTR1), IR=(005706), CLASS=D(SOP=DM0)	5125	K3/99	././.	FI; K3=E12
						7306	K3/99	././.	FI; K3=E5
						7407	K3/99	././.	FI; K3=E44

(Continued)

Module codes: K1/DCS K2/UMOPD K3/IRDECODE K4/DATAPATH K5/RTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	11/66	12/66	13/66	Test summary - Print reference - Chip information
5711	TEST0248	[Continued]			BUT(INSTR1),	IR=(008706),	CLASS-D(SOP+DM0)		
					7417 K3/99	/..	/..	FI; K3=E80,E88,E108,E117	
					7547 K3/99	/..	/..	FI; K3=E118	
					7550 K3/99	/..	/..	FI; K3=E120	
					7553 K3/99	/..	/..	FI; K3=E106	
					7555 K3/99	/..	/..	FI; K3=E103,E107	
					7556 K3/99	/..	/..	FI; K3=E103,E105	
					7577 K3/99	/..	/..	FI; K3=E108	
252)	5712	TFST0308	4147	7424	BUT(IR11=FLOAT),	IR=(008904);	FP=ROM(240)=(04)		
					7420 K3/99	/..	/..	FI; K3=E86	
					7425 K4/60	K2/20	K3/20	FI; K4=E32,E41-E43,E59,E66,E71,E79,E82,E92,E102,	
								E109-E110; K2=E96,E101,E104,E110,E119; K3=E12,	
								E43,E52,E71,E82,E86,E102	
					7426 K3/99	/..	/..	FI; K3=E86	
253)	5713	TFST030C	4166	7403	BUTR(IR11),	IR=(005204);	IR11="1"		
					7401 K3/99	/..	/..	K307=MICROBRANCH	
254)	5715	TEST024A	3920	7402	BUTR(DR6/7L),	IR=(005706);	DR6		
					7403 K3/99	/..	/..	FI; K3=E93	
255)	5717	TEST023A	3877	7560	BUT(INSTR1),	IR=(106004),	CLASS-D(SOP+DM0)		
					8122 K3/99	/..	/..	FI; K3=E14,E23	
					8401 K3/99	/..	/..	FI; K3=E1	
					7401 K3/99	/..	/..	FI; K3=E19	
					7417 K3/99	/..	/..	FI; K3=E47,E60,E74,E80,E88,E90,E99,E104,E108-E110,	
								E116,E119-E120	
					7460 K3/99	/..	/..	FI; K3=E100	
					7510 K3/99	/..	/..	FI; K3=E109	
					7520 K3/99	/..	/..	FI; K3=E100	
					7540 K3/99	/..	/..	FI; K3=E85,E108,E110	
					7557 K3/99	/..	/..	FI; K3=E97,E108-E108,E118	
					7561 K3/88	K4/15	/..	FI; K3=E79,E87,E103,E105; K4=E9	
					7570 K3/99	/..	/..	FI; K3=E76,E110	
					7760 K3/99	/..	/..	FI; K3=E96,E104,E114,E119	
256)	5721	TEST022A	3833	7757	BUT(INSTR1),	IR=(003063),	CLASS-F(BRANCH)		
					0000 K3/99	/..	/..	FI; K3=E6	
					5800 K3/99	/..	/..	FI; K3=E11	
					7417 K3/60	K7/40	/..	FI; K3=E96,E104; K7=E13,E101	
					7517 K3/99	/..	/..	FI; K3=E25	
257)	5722	TFST015C	3519	7655	BUT(INSTR1),	IR=(000152),	CLASS-E(JMP)		
					7651 K3/99	/..	/..	FI; K3=E103,E106	
					7654 K3/99	/..	/..	FI; K3=E93,E108	
					7657 K3/99	/..	/..	FI; K3=E69,E89,E103	

Module codes: K1/DCB K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	11/66	12/66	13/66	Test summary - Print reference - Chip information
258)	5723	TFST021R	3775	7757	BUT(INSTR1),	IR=(002315),	CLASS-F(BRANCH)		
					7417 K3/99	/..	/..	FI; K3=E89,E96,E104,E109	
					7751 K3/99	/..	/..	FI; K3=E116	
259)	5725	TEST021A	3746	7400	BUT(DM0=8MO=8BYTE),	IR=(002315);	-DM0, -8MO, -8BYTE		
					7401 K3/99	/..	/..	FI; K3=E54,E85,E87-E88,E116	
					7402 K3/99	/..	/..	FI; K3=E56,E109-E110,E114,E119	
					7403 K3/50	K6/25	K5/25	FI; K3=E112,E114; K6=E83; K5=E59	
					7404 K3/99	/..	/..	FI; K3=E74,E90,E109,E119-E120	
					7406 K7/99	/..	/..	FI; K7=E20	
					7600 K2/99	/..	/..	FI; K2=E36	
					7606 K2/99	/..	/..	FI; K2=E15	
260)	5726	TEST050R	5108	7412	BUT(INSTR5),	IR=(175200);	E08(752)=(12)		
					7434 K7/99	/..	/..	FI; K7=E13,E29,E94,E101	
					7477 K3/99	/..	/..	K305=INSTR5-DECODE	
261)	5727	TFST020C	3711	7757	BUT(INSTR1),	IR=(001257),	CLASS-F(BRANCH)		
					7417 K3/99	/..	/..	FI; K3=E95-E96,E104,E114	
					7740 K3/99	/..	/..	FI; K3=E99	
					7755 K3/99	/..	/..	FI; K3=E117	
262)	5730	TFST047C	5039	7406	BUT(MOV/DR7=IRS-3),	IR=(122566);	IR<5:3>="110", -MOV		
					7407 K3/99	/..	/..	K308=MICROBRANCH	
263)	5731	TEST020R	3689	7405	BUT(MOV/DR7=IRS-3),	IR=(001257);	IR<5:3>="101", -MOV		
					7407 K3/99	/..	/..	FI; K3=E54	
					7415 K3/99	/..	/..	FI; K3=E94,E120	
264)	5733	TFST020A	3662	7412	BUT(IR9=6),	IR=(001257);	IR<9:6>="1010"		
					7232 K3/99	/..	/..	FI; K3=E58	
					7402 K3/70	K6/30	/..	FI; K3=E76,E83,E85-E86,E88; K6=E82-E83	
					7413 K6/99	/..	/..	FI; K6=E73	
265)	5734	TEST030D	4189	7405	BUT(INSTR5),	IR=(005204);	E08(452)=(05)		
					7400 K3/99	/..	/..	FI; K3=E88	
					7403 K3/99	/..	/..	FI; K3=E88	
					7407 K3/99	/..	/..	FI; K3=E88	
					7415 K3/99	/..	/..	FI; K3=E88	
					7417 K3/99	/..	/..	FI; K3=E88	
					7425 K3/99	/..	/..	FI; K3=E68	
					7426 K3/99	/..	/..	FI; K3=E38,E78,E109	
					7437 K3/99	/..	/..	FI; K3=E38	
					7477 K3/99	/..	/..	K305=INSTR5-DECODE	

Module codes: K1/DCB K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRPDR ##) code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> s1/s% s2/s% s3/s%			Test summary - Print reference - Chip information
266)	5735 TEST017P	3626	7757	BUT(INSTR1),	IR=(000522);	CLASS=F(BRANCH)		
				7417 K3/99	../..	FI; K3=E109-E110		
				7652 K3/99	../..	FI; K3=E114		
				7717 K3/99	../..	FI; K3=E89		
				7752 K3/99	../..	FI; K3=E96		
267)	5737 TEST017A	3598	7405	BUT(IR9=6);	IR=(000522);	IR<9:6>="0101"		
				7401 K3/90	K5/10	FI; K3=E80,E79,E93-E86,E88,E114; K5=E98		
				7405 K3/99	../..	FI; K3=E23		
				7415 K6/65	K5/35	FI; K6=E82-E83; K5=E59		
268)	5741 TEST016A	3555	7757	BUT(INSTR1),	IR=(100000);	CLASS=F(BRANCH)		
				7400 K3/99	../..	FI; K3=E51		
				7417 K3/75	K6/15	FI; K3=E43,E95,E98,E110,E113; K6=E91-E92; K5=E98		
				7512 K6/99	../..	FI; K6=E74		
				7557 K3/99	../..	FI; K3=E99		
				7657 K3/99	../..	FI; K3=E58,E70,E100		
				7717 K3/99	../..	FI; K3=E100		
				7760 K3/99	../..	FI; K3=E100		
				7770 K3/99	../..	FI; K3=E95,E110,E115,E119		
				7777 K3/99	../..	FI; K3=E108		
269)	5743 TEST015A	3473	7425	BUT(INSTR5),	IR=(000152);	E78(152)=(25)		
				5076 K3/99	../..	FI; K3=E10		
				5673 K3/99	../..	FI; K3=E3		
				7305 K3/99	../..	FI; K3=E20		
				7400 K3/99	../..	FI; K3=E33,E106		
				7420 K4/50	K3/35	FI; K4=E12,E14; K3=E75,E103; K6=E65		
				7424 K3/99	../..	FI; K3=E67,E78,E88		
				7434 K2/99	../..	FI; K2=E103,E111,E114		
				7435 K3/99	../..	FI; K3=E76,E78		
				74?? K3/99	../..	K308=INSTR5-DECODE		
				7655 K3/99	../..	FI; K3=E36		
270)	5745 TEST014A	3368	7432	BUT(INSTR5),	IR=(000125);	E78(125)=(32)		
				7400 K3/99	../..	FI; K3=E88		
				7420 K3/50	K6/25	K5/20	FI; K3=E66-E67,E77-E78,E84-E88,E93,E101, E103,E112; K6=E65-E66,E73-E74,E93,E92; K5=E67, E70,E106; K2=E112	
				7422 K3/99	../..	FI; K3=E78,E88		
				7430 K3/99	../..	FI; K3=E78,E88		
				7434 K2/55	K7/15	K6/15	FI; K2=E33,E71,E102,E108; K7=E17; K6=E91; K3=E43	
				7436 K3/99	../..	FI; K3=E75,E78		
				74?? K3/99	../..	K308=INSTR5-DECODE		

Cont

[Cont1

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRPDR ##) code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> s1/s% s2/s% s3/s%			Test summary - Print reference - Chip information
5745	TEST014A	[Continued]		BUT(INSTR5),	IR=(000125);	E78(125)=(32)		
				7672 K3/99	../..	FI; K3=E98		
271)	5746 RFDFT08R	10415	7777	{NVA SEQUENCING LOGIC ERROR}				
				???? K1/99	../..	INTERNAL DCS ERROR		
272)	5747 TEST013G	3331	7417	BUT(INSTR1),	IR=(000000);	CLASS=OTHER(NOT-A-THRU-G)		
				7400 K3/99	../..	FI; K3=E97,E115		
				7407 K3/99	../..	FI; K3=E76,E97,E118		
				7410 K3/99	../..	FI; K3=E97-E98		
				7413 K3/99	../..	FI; K3=E75,E106		
				7415 K3/99	../..	FI; K3=E77,E107		
				7416 K3/99	../..	FI; K3=E67,E105		
				7437 K3/99	../..	FI; K3=E68,E108		
				7440 K3/99	../..	FI; K3=E90,E95,E99,E113,E115-E117,E121		
				7457 K3/99	../..	FI; K3=E70,E97,E100		
				7460 K3/99	../..	FI; K3=E47,E57,E90,E94,E97,E108,E110		
				7810 K3/99	../..	FI; K3=E120		
				7540 K3/99	../..	FI; K3=E108		
				7650 K3/99	../..	FI; K3=E96,E98		
				7710 K3/99	../..	FI; K3=E110		
				7757 K3/99	../..	FI; K3=E89,E110		
				7776 K2/99	../..	FI; K2=E32		
273)	5750 TEST013F	3312	7434	BUT(INSTR5),	IR=(000000);	E78(000)=(34)		
				7400 K3/95	K4/5	K2/5	FI; K3=E38,E59,E68,E88-E89,E96,E98-E99,E104-E107, E109-E110,E113-E114,E116-E119; K4=E4; K2=E25	
				7402 K3/99	../..	FI; K3=E104,E106,E113,E115-E117		
				7414 K3/99	../..	FI; K3=E89		
				7420 K3/99	../..	FI; K3=E78		
				7421 K2/85	K3/25	K5/20	FI; K2=E91,E111; K3=E78,E93; K5=E94	
				7422 K2/80	K3/30	K5/20	FI; K2=E91,E111; K3=E78,E93; K5=E94	
				7424 K3/55	K2/35	K5/15	FI; K3=E76,E78,E88,E93; K2=E91,E103; K5=E94	
				7426 K3/99	../..	FI; K3=E78		
				7427 K2/99	../..	FI; K2=E71		
				7430 K3/99	../..	FI; K3=E75,E78,E88		
				7435 K3/99	../..	FI; K3=E67,E78		
				7436 K3/99	../..	FI; K3=E77-E78		
				7437 K3/99	../..	FI; K3=E78		
274)	5751 TEST013F	3291	7400	BUT(MOV/DR7;IR5=3),	IR=(0000000);	IR<5:3>="000", -MOV		
				7401 K2/50	K3/30	K5/15	K6/5	FI; K2=E84,E91,E103; K3=E67,E93; K5=E94; K6=E65
				7402 K3/45	K2/35	K5/15	K6/5	FI; K3=E77,E103,E107,E109; K2=E91,E114; K5=E105; K6=E73
				7404 K2/45	K3/35	K5/20	FI; K2=E91,E114; K3=E75,E103; K5=E105	
				7410 K3/99	../..	FI; K3=E47,E57,E76,E113,E115-E117,E119-E120		

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%	Test summary - Print reference - Chip information
275)	5752	TEST014R	3394	7403	BUTR(DR6/7L), IR=(000125);	DRS=(DR6+7) 7402 K3/99/.. FI; K3=E37-E39,E64,E66,E79	
276)	5753	TEST013D	3272	7400	BUT(IR9=8), IR=(000000); IR<916>="0000"	7401 K3/99/.. FI; K3=E67 7402 K3/99/.. FI; K3=E77,E103 7404 K3/99/.. FI; K3=E78 7410 K3/99/.. FI; K3=E76	
277)	5754	TEST621H	15572	7402	CHECK "WR-CBP" LATCH CLEARED IN JAMMED U-WORD, WRITE NOT DONE	7401 K7/68 K3/35/.. FI; K7=E25,E41; K3=E32	
278)	5755	TEST013C	3253	7401	BUTR(IR11), IR=(000000); IR11="0"	7403 K3/99/.. FI; K3=E39,E56	
279)	5756	TEST624D	15864	7403	BUTR(PREFETCH=JAM) ASSERTED="1", GETS PREFETCH-H AT JAMUPP="1"	7401 K3/99/.. K306=PREFETCH=JAM-LOGIC 7747 K3/99/.. FI; K3=E17	
280)	5757	TEST013R	3233	7400	BUT(IR11=FLOAT), IP=(000000); IR<11>="0", FP-ROM(000)=(00)	7401 K2/45 K3/30 K5/15 K6/5 FI; K2=E97,E113; K3=E67,E86,E103; K5=E105; K6=E66,E73 7402 K2/45 K3/35 K5/15 K6/5 FI; K2=E97-E98; K3=E77,E86,E114; K5=E97; K6=E82 7404 K2/50 K3/35 K5/15 FI; K3=E97-E98; K3=E78,E86,E114; K5=E97 7410 K2/45 K3/40 K5/15 FI; K2=E97,E107; K3=E76,E86,E114; K5=E97 7417 K3/99/.. FI; K3=E86 7420 K2/50 K3/25 K5/15 K6/10 FI; K2=E97,E107; K3=E68,E114; K5=E97; K6=E82	
281)	5761	TEST013A	3205	7400	BUT(IR15=12), IR=(000000); IR<15>="0000"	7401 K2/55 K3/25 K5/20 FI; K2=E85,E102; K3=E67,E113; K5=E95 7402 K2/50 K3/30 K5/20 FI; K2=E85,E102; K3=E77,E113; K5=E95 7404 K2/45 K3/35 K5/20 FI; K2=E85,E108; K3=E75,E113; K5=E95 7405 K2/99/.. FI; K2=E102-E103,E108,E111,E114 7410 K2/45 K3/30 K5/15 K6/10 FI; K2=E85,E108; K3=E76,E113; K5=E95; K6=E91 7417 K2/60 K5/20 K7/10 K6/5 FI; K2=E1,E17,E23,E34,E55-E56,E62, E70-E71,E76,E82-E83,E85,E92,E100-E101,E105; K5=E54,E84,E94-E95,E97,E105; K7=E28,E37,E54; K6=E101; K4=E26,E29,E38,E87	
282)	5762	TEST014C	3414	7652	BUT(INSTR1), IP=(000125), CLASS=E(JMP)	7417 K3/99/.. FI; K3=E96,E98,E103,E107 7452 K3/99/.. FI; K3=E58,E70,E99 7612 K3/99/.. FI; K3=E100 7640 K3/99/.. FI; K3=E90	[Continued]

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%	Test summary - Print reference - Chip information
	5762	TEST014C	[Continued]		BUT(INSTR1), IP=(000125), CLASS=E(JMP)	7642 K3/99/.. FI; K3=E90,E98,E119 7650 K3/99/.. FI; K3=E103,E107 7653 K3/99/.. FI; K3=E58,E79,E93,E105,E109 7656 K3/99/.. FI; K3=E69,E75,E89,E103,E106,E109 7657 K3/99/.. FI; K3=E98,E105-E107 7740 K3/99/.. FI; K3=E100 7752 K3/99/.. FI; K3=E99,E120	
283)	5763	TEST012G	3169	7474	BUT(INSTR1), IR={177777}, CLASS=A(FLOATING-POINT), FLAG<4,5>="00"	8888/../.. III WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER III 7434 K3/99/.. FI; K3=E37,E58,E70,E97-E98,E100 7437 K3/99/.. FI; K3=E90,E97,E105-E107,E118 7454 K3/99/.. FI; K3=E68,E108 7464 K3/99/.. FI; K3=E76,E113,E118 7470 K3/99/.. FI; K3=E78,E106 7475 K4/45 K3/40 K2/15 K7/5 FI; K4=E3,F33-E34,E51,E55,E71,E82, E100-E101; K3=E3,E43,E51,E67,E82,E94,E102,E105; K2=E77,E95,E106; K7=E43 7476 K4/50 K3/35 K2/15 K5/5 FI; K4=E3,E42-E43,E55,E59,E82,E109-E110; K3=E41,E43,E57,E77,E82,E94,E107; K2=E96,E100, E106; K5=E59 7477 K3/65 K4/40/.. FI; K3=E1,E41,E43,E51-E52,E61,E82,E102; K4=E3,E26, E29,E55,E70-E71 7574 K3/99/.. FI; K3=E70,E97-E98,E100,E108,E116-E117,E120-E121 7674 K3/99/.. FI; K3=E70,E90,E97-E100 7774 K3/99/.. FI; K3=E95-E97,E99-E100,E104,E110,E119	
284)	5764	TEST012F	3147	7402	BUTR(DR6/7L), IR=(177777); DR7	8888/../.. III WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER III 7403 K3/99/.. FI; K3=E64,E66	
285)	5765	TEST012E	3127	7417	BUT(MOV/DR7=IRS-3), IR=(177777); IR<5,3>="111", -MOV	8888/../.. III WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER III 7400 K2/99/.. FI; K2=E26,E91 7406 K3/75 K5/25/.. FI; K3=E93; K5=E94 7407 K3/60 K5/20 K2/10 K4/10 K6/5 K7/5 FI; K3=E57,E66,E71,E76, E78-E79,E81,E84,E92-E93,E95,E99,E101,E105-E107, E110,E113,E115-E119; K5=E48-E49,E67-E68,E94,E106, E113-E115; K2=E91,E103,E111-E112; K4=E12-E15, E106; K6=E65-E66; K7=E73,E82 7413 K3/35 K5/30 K2/15 K6/5 K4/5 K7/4 FI; K3=E75,E78,E86-E87,E91, E103; K5=E56,E61,E70,E92,E105-E106; K2=E91,E114; K6=E73-E74; K4=E106; K7=E88 7415 K5/40 K3/30 K2/10 K7/5 K6/5 K4/5 FI; K5=E56,E61,E67,E105-E106, E112; K3=E77-E78,E86-E87,E101,E103; K2=E91,E114; K7=E88; K4=E73-E74; K4=E106 7416 K5/35 K3/35 K2/15 K6/10 K4/5 K7/5 FI; K5=E49,E67-E68,E94,E106,	

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERROR ###)	code	Symbolic label	Line number	ENUA	TNUA	11/94	12/94	13/94	Test summary - Print reference - Chip information
	5765	TEST012E		[Continued]	BUT(MOV/DR7;IR5=3); IR=(177777); IR<513>="111", -MOV 7416 K6/35 K3/35 K2/15 E120; K3=E67,E78,E86,E93,E101; K2=E91,E103; K6=E65-E66; K4=E106; K7=E82 7437 K3/99 ../.. ../.. FI; K3=E68				
286)	5766	TEST621G	15550	7401	RUTR(PREFETCH=JAM) NEGATED; GETS PREFETCH-H AT JANUPP="0" 7403 K3/99 ../.. ../.. FI; K3=E44 7407 K3/99 ../.. ../.. K306=PREFETCH=JAM-LOGIC 7763 K2/75 K3/25 ../.. FI; K2=E10-E12,E37-E39,E64-E66; K3=E2,E4-E5				
287)	5767	TEST012D	3108	7417	BUT(IR9=6); IR=(177777); IR<916>="1111" 8888 ../.. ../.. 111 WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER 111 7407 K3/99 ../.. ../.. FI; K3=E76 7412 K3/99 ../.. ../.. FI; K3=E34,E54,E64,E67-E68,E74-E75,E98 7414 K3/75 K5/25 ../.. ../.. FI; K3=E103; K5=E105 7415 K3/45 K5/25 K2/10 K4/10 K6/5 K7/5 FI; K3=E34,E44,E50,E56, E76-E79,E84-E86,E88,E103,E112; K5=E51,E61,E70, E105-E106,E121; K2=E97,E113; K4=E12,E14,E105; K6=E73-E74; K7=E64 7416 K3/40 K5/30 K2/15 K7/5 K6/5 K4/5 FI; K3=E67,E78,E84-E86,E88, E96,E103,E112; K5=E51,E61,E70,E105-E106,E122; K2=E97,E113; K7=E44; K6=E73-E74; K4=E105 7437 K3/99 ../.. ../.. FI; K3=E68 7457 K3/99 ../.. ../.. FI; K3=E98				
288)	5770	TEST046B	4950	7401	BUT(IR118FLOAT); IR=(000200); FP-ROM(040)=(01) 7477 K3/99 ../.. ../.. K305=FP-INSTR-DECODE				
289)	5771	TEST012C	3088	7403	RUTR(IR11); IR=(177777); IR11="1" 8888 ../.. ../.. 111 WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER 111 5403 K2/99 ../.. ../.. FI; K2=E34 7401 K3/99 ../.. ../.. FI; K3=E23,E33-E34,E37-E39,E44,E56,E70,E76-E77 7407 K3/99 ../.. ../.. FI; K3=E74 7443 K3/99 ../.. ../.. FI; K3=E37,E98 7475 K2/90 K3/15 ../.. ../.. FI; K2=E34,E40,E59,E65,E68,E74,E80; K3=E39 7477 K3/99 ../.. ../.. FI; K3=E39 7767 K2/99 ../.. ../.. FI; K2=E40				
290)	5772	TEST015F	3500	7403	BUTR(DR6/7L); IR=(000182); DR2=(DR6+7) 7402 K3/99 ../.. ../.. FI; K3=E66,E79				
291)	5773	TEST012B	3069	7437	BUT(IR118FLOAT); IR=(177777); IR<11>="1", FP-ROM(776)=(17) 8888 ../.. ../.. 111 WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER 111 7400 K7/80 K6/15 K3/15 K2/15 FI; K7=E80; K6=E91; K3=E34; K2=E97 7401 K3/75 K5/25 ../.. ../.. FI; K3=E114; K5=E97 7403 K3/99 ../.. ../.. FI; K3=E34 7411 K7/99 ../.. ../.. FI; K7=E80 7413 K3/99 ../.. ../.. FI; K3=E34				
									(Continued)

(Continued)

Module codes: K1/DCS K2/UMWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

			->Module sequence->			Test summary - Print reference - Chip information		
ERROR ##)	Code	Symbolic label	Line number	ENUA	TNUA	11/94	12/94	13/94
5773	TEST012B	[Continued]			BUT(IR118FLOAT), IR=(177777);	IR<11>="1",	FP-ROM(776)=(17)	
			7416	K6/99	/..	FI;	K6=E91	
			7417	K3/65	K5/15	K2/10	K6/5	K7/5 K4/5 FI; K3=E34,E52,E54,E56,E64, E67-E68,E74-E75,E87-E88,E98-E99,E112,E114, E116-E117; K5=E20,E59-E60,E97-E98,E107; K2=E97, E107; K6=E82-E83; K7=E81; K4=E38,E105
			7427	K3/55	K5/20	K2/10	K7/5	K6/5 K4/5 FI; K3=E34,E44,E56,E76-E77, E83,E85-E88,E99,E112,E114,E116; K5=E20,E59-E60, E97-E98,E109; K2=E97,E107; K7=E81; K6=E82-E83; K4=E105
			7433	K3/99	/..	FI;	K3=E75,E86	
			7434	K3/40	K5/30	K2/15	K7/5	K6/5 K4/5 FI; K3=E76,E83,E85-E86,E88, E112,E114,E117; K5=E21,E59-E60,E97-E98,E108; K2=E97-E98; K7=E56; K6=E82-E83; K4=E105
			7435	K3/99	/..	FI;	K3=E77,E86	
			7436	K3/80	K5/25	K2/10	K7/5	K6/5 K4/5 FI; K3=E50,E67,E75,E79, E83-E86,E88-E89,E112,E114; K5=E21,E59-E60, E97-E99; K2=E97-E98; K7=E56; K6=E82-E83; K4=E105
292)	5774	TEST021C	3797	7411	BUT(IR118FLOAT), IR=(002315);	FP-ROM(462)=(11)		
			7401	K3/99	/..	FI;	K3=E86	
			7416	K5/99	/..	FI;	K5=E59	
			7477	K3/99	/..	K305=FP-INSTR-DECODE		
293)	5775	TEST012A	3031	7417	BUT(IR15=12), IR=(177777);	IR<15112>="1111"		
			8888	/..	111	WATCH FOR BUSDIN PULLED LOW BY OTHER DRIVER 111		
			3000	K7/99	/..	FI;	K7=E10,E94	
			3020	K7/99	/..	FI;	K7=E27	
			3220	K7/99	/..	FI;	K7=E10,E21,E27,E29,E42,E52	
			4102	K6/99	/..	FI;	K6=E70,E74,E92,E105	
			5760	K2/99	/..	FI;	K2=E61	
			7005	K2/85	K5/25	K3/25	FI;	K3=E5,E29,E34,E38-E39,E50,E61,E68,E71,E76, E83-E85; K5=E20-E21,E48-E49,E51,E60; K3=E34,E37, E39,E81,E103,E112
			7376	K2/99	/..	FI;	K2=E46	
			7400	K2/40	K3/25	K7/15	K5/15	K6/10 K4/5 FI; K2=E1,E5,E11,E17,E23, E25-E26,E29,E32-E34,E38-E39,E44-E46,E50-E51, E53-E54,E56,E59,E61-E62,E65,E70-E71,E74,E76-E77, E80,E82-E86,E91,E97-E98,E102-E103,E105,E107-E108, E111,E113-E114; K3=E23,E31-E34,E36-E39,E43,E45, E48,E51-E52,E57,E65-E66,E73,E92-E93,E101,E103, E112-E114; K7=E10,E13,E17,E20-E21,E27,E29,E33, E41,E52,E94; K5=E6,E20-E21,E31,E48-E49,E51,E54, E56,E61,E77,E95; K6=E65-E66,E70,E73-E74,E79-E80, E82-E83,E86,E95-E96,E98-E99,E101,E105,E108; K4=E29-E30,E38
			7401	K3/99	/..	FI;	K3=E113	
			7402	K2/99	/..	FI;	K2=E29,E34,E61	

(Conti

Module codes: K1/DCS K2/UMWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERROR ##)	code	Symbolic label	Line number	ENUA	TNUA	Module sequence-> #1/#2/#3/#4	Test summary - Print reference - Chip information
5775	TEST012A	[Continued]				BUT(IR15=12), IR=(177777); 7403 K7/96 K2/5/.. FI; K7=E28,E56,E64-E65,E73,E81-E82,E88; K2=E61 7404 K6/50 K2/50/.. FI; K6=E105; K2=E76 7408 K2/55 K6/25 K3/15 K5/5 FI; K2=E34,E71,E83-E85,E98,E102-E103, E107-E108,E111,E113-E114; K6=E3,E5; K3=E34,E45; K5=E65 7407 K3/50 K5/25 K2/10 K6/5 K7/5 K4/5 FI; K3=E21,E76,E83,E85, E87-E88,E108,E110,E113; K5=E31,E77,E91,E95,E98; K2=E85,E108; K6=E91-E92; K7=E80; K4=E95 7412 K3/85 K5/10 K7/5 K2/5 FI; K3=E34,E54,E64,E67-E68,E74-E75; K5=E68, E92; K7=E33; K2=E61 7413 K5/45 K3/30 K6/10 K2/10 K7/5 K4/5 FI; K5=E31,E59,E67,E70,E77, E84,E88,E98,E101; K3=E21,E75,E85,E113,E121; K6=E91-E92,E95,E105; K2=E85,E108; K7=E80; K4=E95 7414 K5/99/../.. FI; K5=E60-E61,E77 7415 K3/40 K5/25 K2/15 K6/5 K4/5 K7/5 FI; K3=E21,E77,E85,E94,E104, E113; K5=E6,E67,E77,E90,E95,E98; K2=E85,E102; K6=E91-E92; K4=E95; K7=E65 7416 K3/40 K5/30 K2/10 K6/5 K4/5 K7/5 FI; K3=E21,E67,E85,E94,E104, E113; K5=E6,E67,E77,E95,E98,E100; K2=E85,E102; K6=E91-E92; K4=E95; K7=E65 7437 K3/99/../.. FI; K3=E34,E38,E68 7447 K3/99/../.. FI; K3=E70,E98	
294)	5776	TEST624C	15839	7402		PROC-MUX(CUA-PORT)=(062851); CUA LOCKED, EXFLAG CLEAR, PREFETCH=JAM SET 7216 K3/99/../.. FI; K3=E48 7400 K3/75 K2/25/.. FI; K3=E44,E48; K2=E111	
295)	5777	TEST011	2847	6532		BUTA(SUBR-B) -> BUTA(RETURN) SEQUENCE, RETURN="1101 0101 1010"=(6532) 0777 K2/99/../.. FI; K2=E55 4644 K2/99/../.. FI; K2=E46 4747 K2/70 K4/20 K3/5 FI; K2=E5,E11,E23,E34,E46,E55; K4=E3; K3=E23 5245 K6/99/../.. FI; K6=E80 6132 K2/99/../.. FI; K2=E19,E41,E77-E78 6432 K2/99/../.. FI; K2=E7,E13,E25 6512 K2/99/../.. FI; K2=E7,E13,E28 6532 K2/65 K4/35/.. FI; K2=E26,E97; K4=E15 6533 K2/99/../.. FI; K2=E8,E14,E26 6536 K2/99/../.. FI; K2=E8,E14,E26 7005 K3/85 K2/45/.. FI; K3=E4,E6-E7,E11,E15,E43; K2=E70,E85 7271 K3/60 K2/40/.. FI; K3=E4,E6-E7,E11,E15,E43; K2=E61,E77,E85 7274 K2/70 K3/30/.. FI; K2=E1,E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E23, E58,E70 7275 K2/99/../.. FI; K2=E8,E25 7301 K6/99/../.. FI; K6=E104 7417 K3/99/../.. FI; K3=E23 7532 K2/99/../.. FI; K3=E35,E41,E77 7577 K2/99/../.. FI; K3=E42	

Module codes: K1/DCS K2/WORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

ERROR ##)	code	Symbolic label	Line number	ENUA	TNUA	Module sequence-> #1/#2/#3/#4	Test summary - Print reference - Chip information
296)	6522	TEST002	2654	6631		NVA SEQUENCING, NO 'BUT' 2631 K2/99/../.. FI; K2=E55 6231 K3/55 K2/45/.. FI; K3=E63,E111; K2=E1,E19 6777 K3/50 K2/50/.. FI; K3=E11; K2=E32	
297)	6520	TEST720R	16788	7402		DATIS=BYTE, ODD ADDR JAM; JAM=(101004) 7400 K6/99/../.. FI; K6=E82	
298)	6521	TEST713D	16689	7402		DATI=NOINT, I/O PAGE, 16, BIT I/O PAGE MODE; SERVICE=(101740) 7400 K6/99/../.. FI; K6=E81 7401 K2/99/../.. FI; K2=E8,E14,E26	
299)	6522	TEST713C	16664	7403		BUTR(OTHER-JAM=FP), ILLEGAL INTERNAL ADR; OTHER-JAM="1" 7401 K7/99/../.. FI; K7=E74	
300)	6523	TEST712D	16555	7402		DATI, I/O PAGE, 16, BIT I/O PAGE MODE; SERVICE=(101740) 7400 K7/99/../.. FI; K7=E82 7403 K7/65 K6/35/.. FI; K7=E80; K6=E89	
301)	6524	TEST712C	16530	7401		BUTR(OTHER-JAM=FP), VALID INTERNAL ADDR; OTHER-JAM="0" 7403 K2/65 K5/15/.. FI; K2=E26,E88,E104,E110,E116; K5=E105	
302)	6526	TEST711D	16427	7402		CLEAR JAM ERRORS/SSYN TIMEOUT; JAM=(001000) 7400 K6/90 K7/10/.. FI; K6=E3,E51,E80-E81,E99; K7=E58	
303)	6530	TEST711C	16404	7402		DATOB=BYTE, I/O PAGE, 16, BIT I/O PAGE MODE; SERVICE=(005740) 7400 K2/70 K4/20 K7/5 K6/5 FI; K2=E1,E8,E14,E26,E31,E37,E43,E49,E52, E71,E82,E91,E105; K4=E13,E15; K7=E81,E97; K6=E34 7401 K2/99/../.. FI; K2=E26	
304)	6534	TEST711R	16374	7402		DATOB=BYTE, SSYN TIMEOUT; JAM=(021200) 7400 K6/65 K7/30 K5/5 FI; K6=E80-E82,E81,E99; K7=E58,E64-E65,E84; K5=E84 7401 K6/50 K7/25 K8/20 K3/10 FI; K6=E47,E55,E76,E85; K7=E81,E84; K5=E64, E67,E93; K3=E34	
305)	6535	TEST512F2	13261	7412		KT-BSPHI(SF1=(06), NEGATED 7400 K3/99/../.. FI; K3=E47 7477 K3/99/../.. K312=KT-SP-ADDR-DECODE	
306)	6537	TEST512D1	13185	7400		KT-ASPHI(SF1=(16), ASSERTED, -P2=8R6+P815 7412 K3/99/../.. FI; K3=E47 7477 K3/99/../.. K312=KT-SP-ADDR-DECODE	
307)	6541	TEST373A	9738	4777		BUTA(DIAGNOSE) CAUSES XFER TO B.M. ROM/JAMUPP TO DCS RETURN 0000 K2/99/../.. FI; K2=E24,E30,E39,E45,E48,E51,E54,E60,E66,E68-E69,	

Module codes: K1/DCS K2/WORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
6541	TEST373A	[Continued]			BUTA(DIAGNOSE) CAUSES XFER TO B,M, ROM/JANUPP TO DCS RETURN				
					0000 K2/99	./..	./..		E78,E81
					0003 K3/99	./..	./..	FI;	K3=E23
					0020 K2/99	./..	./..	FI;	K2=E21
					0055 K2/99	./..	./..	FI;	K2=E18
					0212 K2/99	./..	./..	FI;	K2=F4,E10,E16,E22,E27=E28,E31,E36-E37,E42-E43, E49,E52,E58,E64,E67,E73,E79
					0777 K2/99	./..	./..	FI;	K2=E24
					1020 K2/99	./..	./..	FI;	K2=E50
					1025 K6/99	./..	./..	FI;	K6=E42
					2000 K2/99	./..	./..	FI;	K2=E48,E51,E57,E63
					2042 K2/99	./..	./..	FI;	K2=E21,E74
					2153 K2/99	./..	./..	FI;	K2=E75
					2211 K2/99	./..	./..	FI;	K2=E75
					2247 K3/99	./..	./..	FI;	K3=E22
					2310 K2/99	./..	./..	FI;	K2=E75
					2314 K2/99	./..	./..	FI;	K2=E75
					2408 K2/99	./..	./..	FI;	K2=E75
					2442 K2/99	./..	./..	FI;	K2=E21
					2526 K2/99	./..	./..	FI;	K2=E24
					2530 K2/99	./..	./..	FI;	K2=E24
					2577 K2/99	./..	./..	FI;	K2=E48
					2671 K2/99	./..	./..	FI;	K2=E57
					3007 K2/99	./..	./..	FI;	K2=E57,E78
					3055 K2/99	./..	./..	FI;	K2=E5,E11,E15,E17,E23,E29,E32,E38,E44,E50,E53, E59,E65,E68,E74,E80
					3393 K2/99	./..	./..	FI;	K2=E75
					3462 K2/99	./..	./..	FI;	K2=E42,E74
					3505 K2/99	./..	./..	FI;	K2=E15,E21,E74
					3507 K2/99	./..	./..	FI;	K2=E21
					3922 K2/99	./..	./..	FI;	K2=E5,E11,E15,E17,E21,E27,E29,E32,E44,E50,E53, E68,E73-E74,E80
					3637 K2/99	./..	./..	FI;	K2=E48,E68
					3777 K2/99	./..	./..	FI;	K2=E42
					4621 K2/99	./..	./..	FI;	K2=E68
					4747 K3/85 K2/15	./..	./..	FI;	K3=E9-E9; K2=E18
					5501 K2/99	./..	./..	FI;	E2=E48,E57,E63,E78
					5555 K3/99	./..	./..	FI;	K2=E48,E57,E63,E78
					5597 K2/99	./..	./..	FI;	K2=E78
					6073 K2/99	./..	./..	FI;	K2=E73
					7005 K8/99	./..	./..	FI;	K2=E6,E12,E18,E24,E33,E45,E75
					7037 K4/80 K2/50	./..	./..	FI;	K4=E66; K2=E68
					7274 K3/99	./..	./..	FI;	K2=E35,E47,E55
					7373 K2/99	./..	./..	FI;	K2=E36,E73
					7637 K6/60 K7/40	./..	./..	FI;	K6=E39-E40,E42-E43,E56; K7=E36,E52
					7777 K2/99	./..	./..	FI;	K2=E47

Module codes: K1/DCS K2/UMORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
3081	6547	TEST512A1	13026	7405	KT-ASPHI[SF]=(02), NEGATED, -SR6				
					7412 K3/99	./..	./..	FI;	K3=E45,E47
					7477 K3/99	./..	./..	K312=KT-SP-ADDR-DECODE	
3091	6544	TEST512A2	13053	7400	KT-BSPHI[DF]=(16), ASSERTED, DR6=PS15=-FLOAT				
					7412 K3/85 K4/15	./..	./..	FI;	K3=E45,E47,E58,E69,E79; K4=E13,E15
					7477 K3/99	./..	./..	K312=KT-SP-ADDR-DFCODE	
3101	6546	TEST512F2	13108	7400	KT-BSPHI[SF]=(16), ASSERTED, F2=SR6+SM0=PS13=-FLOAT				
					7412 K3/65 K4/35	./..	./..	FI;	K3=E45,E47,E89; K4=E13,E15
					7477 K3/99	./..	./..	K312=KT-SP-ADDR-DECODE	
3111	6552	MFSS05	12715	6545	BUTR(MF-SAME-STACK)=H, EXPECTED L				
					6547 K3/80 K2/20	./..	./..	FI;	K3=E56,E79,E89,E96; K2=E106,E118
3121	6553	MFSS06	12725	6547	BUTR(MF-SAME-STACK)=L, EXPECTED H				
					6545 K3/75 K2/20 K5/5	./..	./..	FI;	K3=E13,E47,E56,E63,E69,E79,E89,E96; K2=E62,E84, E106,E118; K5=E63
3131	6556	TEST512C1	13130	7400	KT-ASPHI[DF]=(16), ASSERTED, F1=DR6+DM0=PS13=-FLOAT				
					7412 K2/99	./..	./..	FI;	K2=E119
					7477 K3/99	./..	./..	K312=KT-SP-ADDR-DECODE	
3141	6557	TEST507D	12243	7407	BUTR(MASK-PS[T]), MASK-PS[T]=1=1				
					7406 K2/99	./..	./..	FI;	K2=E78,E101,E105,E112
3151	6561	TEST500	10968	6131	(NUA SEQUENCING LOGIC ERROR)				
					7777 K1/99	./..	./..	INTERNAL DCS ERROR	
3161	6563	TEST774	9898	NONF	(NUA SEQUENCING LOGIC/DCS ERROR)				
					7777 K1/99	./..	./..	INTERNAL DCS ERROR	
3171	6565	TEST512R1	13081	7412	KT-ASPHI[DF]=(06), NEGATED				
					7400 K3/99	./..	./..	FI;	K3=E45,E47
					7477 K3/99	./..	./..	K312=KT-SP-ADDR-DECODE	
3181	6566	TEST512C2	13157	7412	KT-BSPHI[SF]=(06), NEGATED				
					7400 K3/99	./..	./..	FI;	K3=E47
					7406 K4/99	./..	./..	FI;	K4=E59
					7477 K3/99	./..	./..	K312=KT-SP-ADDR-DFCODE	
3191	6567	ERROR624A	15803	0005	MICROBREAK JANUPP AT (6255) ATTEMPTED; DID NOT OCCUR				
					6567 K3/99	./..	./..	FI;	K3=E81,E91,E101,E111-E112

Module codes: K1/DCS K2/UMORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##	PPROR Code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
320)	6571	TEST372A	9606	6222	RUTA(SUBR-A)/BUTA(RETURN)	SEQUENCE; D<14103> -> RETURN; "1100 1001 0010"=(6222			
					4022 K2/99	../. ../. FI;	K2=E92		
					4222 K2/99	../. ../. FI;	K2=E48,E77,E89,E92-E93,E95,E102,E106,E118		
					6000 K2/99	../. ../. FI;	K3=E90		
					6022 K2/99	../. ../. FI;	K3=E1,E6,E12,E18,E24-E25,E30,E33,E39,E45,E51,		
							E54,E60,E66,E69,E75,E81,E89,E92-E93,E96,E101,E107		
					6076 K2/99	../. ../. FI;	K2=E84,E90		
					6200 K6/50	K2/50 ../. ../. FI;	K6=E34; K2=E86		
					6202 K2/99	../. ../. FI;	K2=E1,E6,E12,E18,E24,E30,E33,E39,E45,E51,E54,		
							E60,E66,E69,E75,E81,E86-E88,E104,E110,E113,E116		
					6220 K2/99	../. ../. FI;	K2=E6,E9,E12,E18,E24,E26,E30,E33,E39,E45,E51,		
							E54,E60,E66,E69,E75,E81,E86-E88,E104,E110,E114,		
							E116		
					6223 K2/99	../. ../. FI;	K2=E9,E26,E86,E103		
					6226 K2/99	../. ../. FI;	K2=E9,E26,E86,E114		
					6332 K2/99	../. ../. FI;	K2=E9,E26,E86,E113		
					6262 K2/99	../. ../. FI;	K2=E3,E25,E86,E98		
					6322 K2/99	../. ../. FI;	K2=E3,E25,E92,E98		
					6362 K2/99	../. ../. FI;	K2=E98		
					6376 K2/70	K7/15 K5/15 FI;	K2=E61,E71,E93; K7=E36,E51; K5=E15		
					6622 K2/99	../. ../. FI;	K2=E42,E77,E92,E107		
					7222 K2/99	../. ../. FI;	K2=E72,E77,E92,E102		
					7700 K2/99	../. ../. FI;	K2=E86		
321)	6572	TEST373B	9814	7425	EXEC B,N, ROM CODE FOR "PLPADR" ASSERT TO ASPHI/READ				
					7400 K4/85	K2/35 K3/10 FI;	K4=E1,E16,E20,E51,E54,E59-E60,E67,E69,E84,E93,		
							E104,E114; K2=E4,E7,E10,E16,E18,E22,E25,E27-E28,		
							E37,E43,E49,E52,E58,E64,E73; K3=E3,E7		
					7402 K3/55	K2/25 K4/15 FI;	K3=E1-E20,E22-E26,E45,E51,E66; K2=E4,E10,E16,		
							E22,E27-E28,E31,E36-E37,E43,E49,E52,E58,E64,E67,		
							E69,E73,E79; K4=E13,E15,E51,E54,E62,E69,E113-E114		
					7403 K3/95	K4/5 ../. ../. FI;	K3=E1-E19,E22,E24-E26; K4=E29		
					7417 K4/80	K2/20 ../. ../. FI;	K4=E9-E60,E84,E93; K2=E31,E69		
					7420 K4/99	../. ../. FI;	K4=E1,E61,E66,E68,E70,E74-E77,E83,E85,E93-E94,		
							E103-E104,E113-E114		
					7426 K2/99	../. ../. FI;	K2=E43		
					7432 K2/99	../. ../. FI;	K2=E4,E16,E22,E27-E28,E37,E43,E52,E58,E67,E79		
					7433 K4/99	../. ../. FI;	K4=E84,E60,E66-E68,E70,E75-E76,E83,E103		
					7434 K7/30	K6/30 K2/30 K3/15 FI;	K7=E41,E66; K6=E78,E90; K2=E18,E75;		
							K3=E7		
322)	6573	TEST361A	8827	7434	SR=(052535), D[C]="1"; SR/XMU=(100125) (FLTPT)				
					7400 K4/96	K3/5 ../. ../. FI;	K4=E27,E29,E37,E44,E46,E54,E69,E78; K3=E92		
					7402 K4/99	../. ../. FI;	K4=E46		
					7417 K4/99	../. ../. FI;	K4=E27		{Continued}

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/RTORCHE K6/TIMING K7/STATUS

##	PPROR Code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
	6573	TEST361A	{Continued}		SR=(052535), D[C]="1"; SR/XMU=(100125) (FLTPT)				
					7420 K4/99	../. ../. FI;	K4=E37,K44		
					7425 K4/99	../. ../. FI;	K4=E18		
					7426 K4/99	../. ../. FI;	K4=E37		
323)	6574	TEST710D	16265	7402	DATO, I/O PAGE, 16, BIT I/O PAGE MODE; SERVICE=(003740)				
					7400 K7/50	K6/30 K5/20 FI;	K7=E44,E68,E71; K6=E51,E53-E54,E59,E105; K5=E7,		
							E27,E30,E95		
324)	6575	TEST710E	16294	7402	CLEAR JAM ERRORS/ODD ADDR; JAM=(001000)				
					7401 K7/99	../. ../. FI;	K7=E25,E59		
325)	6576	TEST512D2	13212	7405	KT-BSPHI[DF]=(02), NEGATED, -DR6				
					7412 K3/99	../. ../. FI;	K3=E47,E79,E93		
					7417 K3/99	../. ../. FI;	K312=KT-BP-ADDR-DECODE		
326)	6577	TEST512F1	13234	7400	KT-ASPHI[DF]=(16), ASSERTED, F1=DR6=DN0=PS15				
					7405 K3/99	../. ../. FI;	K3=E45		
					7412 K3/99	../. ../. FI;	K3=E47		
					7417 K3/99	../. ../. FI;	K312=KT-BP-ADDR-DECODE		
327)	6600	TEST510D	12438	7406	BUTM(MASK-PS[T]), MASK-PS[T]=1=0=0				
					7407 K2/99	../. ../. FI;	K2=E72,E101,E105		
328)	6601	TEST503DA	11359	7406	BUTM(FLTPT-INSTR), IR=(125122), -FLTPT				
					7407 K2/99	../. ../. FI;	K2=E53,E112		
329)	6603	TEST510A	12363	7407	BUTR(BG-SERVICE-L), NEGATED WHEN PS PRIO<715>=(7)				
					7403 K7/80	K3/15 K2/5 FI;	K7=E3,E5,E28,E76; K3=E50,E74; K2=E100		
330)	6604	TEST510E	12497	7403	BUTR(SERVICE-H), ASSERTED=1 WHEN FLAG<7>H=1				
					7402 K2/99	../. ../. FI;	K2=E78,E100		
331)	6605	TEST506F	12099	7403	BUTR(INTR-HIGH), INTR-HIGH=H=1; INTERNAL SERVICE CLEAR				
					4747 K6/99	../. ../. FI;	K6=E78		
					7401 K7/85	K3/10 K2/5 FI;	K7=E37,E43,E48,E50,E57-E58,E66-E67,E76-E77,		
							E84; K3=E44,E50; K2=E72,E100		
332)	6607	TEST511A	12755	NONE	(MHA SEQUENCING LOGIC ERROR)				
					7777 K1/99	../. ../. FI;	INTERNAL DCS ERROR		
333)	6610	TEST510F	12524	7402	BUTR(SERVICE-H), NEGATED=0, NO INPUTS ACTIVE				
					7403 K7/80	K2/15 K3/5 FI;	K7=E3,E7-E8,E11,E16,E27; K2=E38,E78,E100;		
							K3=E64		
334)	6611	TEST507C	12221	7406	BUTM(PS03), PS<03>H=0				
					7407 K2/99	../. ../. FI;	K2=E112		
					7407 K2/80	K3/20 ../. ../. FI;	K210=MULTIPLE-BUT; K306=PS[CC]		

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/RTORCHE K6/TIMING K7/STATUS

FROR	Code	Symbolic label	Line number	ENUA	TNUA	Module sequence->	Test summary - Print reference - Chip information
###	---	-----	-----	----	-----	1/4 2/4 3/4	-----
335)	6613	TEST507A	12144	7402	LOAD/READ PS=(140125) PS=PORT PROC=MUX; UCON FUNCTION		
					7400 K2/50 K3/45 K4/5 K7/5 FI; K2=E102-E103,E105,E111,E113-E114,E116,E118; K3=E35,E71-E73,E83; K4=E63,E72; K7=E3		
					7401 K3/45 K2/45 K4/10 FI; K3=E28,E47,E54,E63; K2=E108,E118; K4=E72		
336)	6615	TEST504A	11578	4777	NUA SEQUENCING ERROR, JAMUPP ERROR; RUTA(DIAGNOSE) TO B.M. 'LOADNZW4/LOADNZW5'		
					4744 K2/99/../.. FI; K2=E54		
					4757 K2/99/../.. FI; K2=E54		
337)	6616	TEST510C	12418	7403	BUTR(SERVICE-H), ASSERTED=1 WHEN INTR-HIGH-H=0		
					7345 K3/99/../.. FI; K3=E76		
					7402 K2/75 K3/25/.. FI; K2=E100; K3=E64		
338)	6617	TEST510DA	12475	7407	BUTM(D00), D=(040401)		
					7406 K2/99/../.. K210=MULTIPLE-BUT		
339)	6621	TEST506A	11952	7402	LOAD/READ PS=(030252) PS/PROC=MUX; BUTA(CLR=...) SETS BUSDIN=ENIT; UCON FCN		
					7400 K2/50 K3/40 K7/10 K5/5 FI; K2=E61-E62,E70-E71,E82-E83,E98,E102-E103,E107-E108,E111-E114,E116-E118; K3=E47,E52,E57,E62-E63,E72-E73,E83,E89; K7=E3,E27,E42,E44,E48; K8=E13,E54,E84		
					7401 K2/90 K3/10/.. FI; K3=E62,E82,E117; K3=E57		
					7403 K2/99/../.. FI; K2=E108		
340)	6623	TEST505A	11847	7402	SET ALL FLAGS, CHECK BUTA(CLR=FLAG-RES-UCON) CLEARS SHORT TERM FLAGS		
					7400 K2/99/../.. FI; K2=E101,E105,E119		
341)	6624	TEST533A	13528	7402	SHIFTER, AMUX/D[HI]#D[LO]=(052652), D=(0)(052652)		
					7400 K4/99/../.. FI; K4=E29,E51,E54,E60-E62,E66-E69,E77,E81-E85,E93-E94,E103-E104,E113-E114		
342)	6627	TEST520A	13322	7403	INSTR-BRANCH-L=H, IR=(101004)BHI, PS(INVVC)=(04)		
					7402 K3/99/../.. FI; K3=E64,E83		
					7407 K3/99/../.. K2=PS(CC)-BRANCH(F64,72-73,83)		
343)	6630	TEST503B	11318	7406	RUTH(FLAG7), FLAG<7>H=0		
					7407 K2/85 K3/15/.. FI; K2=E112; K3=E54		
344)	6631	TEST003	2673	5525	NUA SEQUENCING, PAGE (6) -> (5), UBF=(35)		
					4525 K2/80 K4/10 K3/10 FI; K2=E35,E47,E58,E62,E64,E67,E72-E73,E78,E91; K4=E3; K3=E111		
					5425 K3/80 K2/50/.. FI; K3=E58,E91; K2=E1,E7		
					5520 K2/99/../.. FI; K2=E76		
					5544 K2/99/../.. FI; K2=E90		
					5546 K2/99/../.. FI; K2=E71		

(Continued)

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FROR	Code	Symbolic label	Line number	ENUA	TNUA	Module sequence->	Test summary - Print reference - Chip information
###	---	-----	-----	----	-----	1/4 2/4 3/4	-----
6631	TEST003	[Continued]			NUA SEQUENCING, PAGE (6) -> (5), UBF=(35)		
					6525 K2/99/../.. FI; K2=E46,E59,E65,E68,E74,E80		
					7525 K2/99/../.. FI; K2=E35,E47		
345)	6632	TEST503C	11338	7402	BUTR(FLTP-PROC-H), FLTP-PROC-H=0		
					4402 K2/99/../.. FI; K2=E34		
					7403 K3/50 K2/50/.. FI; K3=E64; K2=E119		
					7406 K2/60 K3/40/.. FI; K2=E2,E8,E66,E69,E75,E81; K3=E21,E81		
346)	6633	TEST533B	13569	7402	SHIFTER, AMUX/D[HI]#D[LO]=(125125), D=(0)(125125)		
					6367 K3/99/../.. FI; K3=E38		
					7400 K4/99/../.. FI; K4=E51,E54,E59-E62,E66-E69,E74-E76,E83-E85,E93-E94,E103-E104,E113		
					7401 K4/99/../.. FI; K4=E75-E76		
					7403 K4/99/../.. FI; K4=E51,E54,E62,E69,E84,E103,E114		
347)	6634	TEST534B	13650	7402	SHIFTER, AMUX/D[LO]#D[HI]=(052652), D=(0)(125125)		
					7400 K4/99/../.. FI; K4=E93-E94,E103-E104,E113		
					7403 K4/99/../.. FI; K4=E114		
348)	6635	TEST534D	13725	7402	SHIFTER, AMUX/D[C]#D[HI]=(000377), D=(0)(177777)		
					7400 K4/75 K2/20 K3/10 FI; K4=E59,E93,E104,E113-E114; K2=E65,E68; K3=E75		
					7403 K4/99/../.. FI; K4=E114		
349)	6636	TEST534F	13805	7402	SHIFTER-COUNTER, AMUX/COUNT#D[HI]=(052652), CTR=(125), D=(0)(125000)		
					7400 K4/99/../.. FI; K4=E87,E90-E91,E93,E104,E113		
					7401 K4/99/../.. FI; K4=E114		
					7403 K4/99/../.. FI; K4=E8,E91,E114		
350)	6637	TEST535B	13926	7402	SHIFTER, BMUX/4#D[C]#AMUX=(002645), D=(0)(055132)		
					7400 K4/99/../.. FI; K4=E69,E83-E85		
					7403 K4/99/../.. FI; K4=E69		
351)	6640	TEST536B	14019	7402	SHIFTER, CMUX/2#D[C]#BMUX=(034343), D=(0)(161616)		
					7400 K4/99/../.. FI; K4=E54,E60-E61,E67,E75		
					7401 K5/99/../.. FI; K5=E105		
					7403 K4/99/../.. FI; K4=E54		
352)	6641	TEST536D	14103	7402	SHIFTER, CMUX/D[C]#BMUX=(063146), D=(0)(146314)		
					7400 K4/90 K3/15/.. FI; K4=E60-E62,E67-E68,E75-E76; K3=E42		
					7403 K4/99/../.. FI; K4=E54		
353)	6642	TEST536F	14196	7402	SHIFTER, CMUX/BMUX#ENDMUX=(063146), D/SR=(0)(031463)(077777)		
					7400 K4/99/../.. FI; K4=E60-E62,E67-E68,E75-E77		
					7403 K4/99/../.. FI; K4=E54		
354)	6643	NEWCTP537A	14291	7370	(NUA SEQUENCING LOGIC ERROR)		
					7777 K1/99/../.. INTERNAL DCS ERROR		

Module codes: K1/DCS K2/UWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##)	FRROP code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/# #2/# #3/#			Test summary - Print reference - Chip information
355)	6644	TEST710B	16212	7402	DATO, ODD ADDR ERR; JAM=(101004) 7400 K7/45 K5/45 K6/10 FI; K7=E61,E68-E66,E80,E82,E84; K5=E13,E43-E45, E64; K6=E76 7401 K6/80 K7/20 ../.. FI; K6=E71,E80-E81,E99; K7=E80,E82				
356)	6645	TEST710C	16236	7402	DATO, I/O PAGE, 18. BIT I/O PAGE MODE; SERVICE=(002340) 7400 K7/65 K5/15 K6/18 K4/5 K3/5 FI; K7=E14,E27,E32,E44,E64-E65,E68, E70,E73,E76-E77,E79-E82,E84,E88; K5=E13-E14,E39, E67,E93; K6=E43,E46-E47,E53-E54,E59,E81; K4=E20, E65; K3=E34 7401 K2/65 K7/15 K6/15 FI; K2=E1,E82; K7=E80; K6=E43 7403 K7/99 ../.. FI; K7=E80				
357)	6646	TEST503K	11531	7407	BUTM(FLPT=INSTR), IR=(175282), FLPT 7406 K2/50 K3/50 ../.. K210=MULTIPLE-BUT; K304=FP-DECODE				
358)	6647	TEST507F	12286	7407	CHECK PS(C)=PS(000)=1", D(C) INPUTS PS(C), CIN/PS(C), CIN/D(C)=PS(C) ALL SET 7400 K3/50 K2/50 ../.. FI; K3=E93; K2=E112 7401 K4/99 ../.. FI; K4=E86 7406 K3/99 ../.. FI; K3=E35,E46				
359)	6653	TEST536E	14154	7402	SHIFTER, CMUX/BMUX:8ENDMUX=(114631), D/SR=(0)(146314)(100000) 7400 K4/99 ../.. FI; K4=E54,E60-E62,E67-E68,E75-E77,E80				
360)	6655	TEST536C	14067	7402	SHIFTER, CMUX/D(C):BMUX=(114631), D=(1)(031463) 7400 K4/95 K6/5 ../.. FI; K4=E54,E60-E62,E67-E68,E70,E75-E76; K6=E108				
361)	6657	TEST536A	13973	7402	SHIFTER, CMUX/2D(C):BMUX=(143434), D=(1)(016161) 7400 K4/99 ../.. FI; K4=E60-E62,E68,E76,E84 7401 K4/99 ../.. FI; K4=E54,E60-E62,E67-E68,E70,E75-E76				
362)	6661	TEST535A	13879	7402	SHIFTER, BMUX/4D(C):AMUX=(175132), D=(1)(122645) 7400 K4/99 ../.. FI; K4=E3,E60,E69,E75,E77,E83-E85				
363)	6663	TEST534E	13765	7402	SHIFTER/COUNTER, AMUX/COUNT(DLO)=(125125), CTR=(252), D=(0)(000125) 7400 K4/96 K2/5 ../.. FI; K4=E87,E90-E91,E93,E104,E113-E114; K2=E55 7401 K4/80 K6/15 K2/5 FI; K4=E87,E90-E91; K6=E103; K2=E55 7403 K4/99 ../.. FI; K4=E91,E114				
364)	6665	TEST534C	13691	7402	SHIFTER, AMUX/8D(C):D(HI)=(177400), D=(1)(000000) 7400 K4/99 ../.. FI; K4=E67-E88 7401 K4/99 ../.. FI; K4=E28-E29,E93,E104,E113-E114				
365)	6667	TEST534A	13615	7402	SHIFTER, AMUX/D(LO):D(HI)=(125125), D=(0)(052652) 6356 K2/65 K3/35 ../.. FI; K2=E2,E8-E9,E21,E27,E60,E66,E69,E75,E81;				

Module codes: K1/DCS K2/WORD K3/TPDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	PROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/# #2/# #3/#			Test summary - Print reference - Chip information
366)	6667	TEST534A	[Continued]			SHIFTER, AMUX/D(LO):D(HI)=(125125), D=(0)(052652) 6356 K2/65 K3/35/.. K3=E21-E22,E44,E56,E77 7341 K2/99/.. FI; K2=E34 7343 K3/99/.. FI; K3=E37-E39,E44,E64 7400 K4/90 K3/5 K2/5 FI; K4=E28-E29,E77,E93-E94,E103-E104,E113-E114; K3=E3; K2=E74 7401 K4/99/.. FI; K4=E28-E29,E93,E104,E113-E114			
366)	6670	TEST503D	11379	7476		BUT(INSTR1), IR=(175282), CLASS=A(FLTPT), FLAG<4:5>H="10" 7474 K2/85 K7/15/.. FI; K2=E62,E100; K7=E44 7477 K3/99/.. K308=FLAG8/FP-INSTR1			
367)	6671	TEST500	11008	6377		(NUA SEQUENCING LOGIC ERROR) 6173 K2/70 K3/30/.. FI; K2=E2-E3,E6-E7,E12,E15,E18,E24,E30,E36; K3=E23, E50,E70 7376 K3/99/.. FI; K3=E58 7777 K1/99/.. INTERNAL DCS ERROR			
368)	6672	TEST504F	11742	7407		BUTM(EXFLAG2), EXFLAG<2>H="1" 7406 K2/99/.. K210=FLAG8			
369)	6673	TEST500A	11070	7434		SP REWRITE FUNCTION WR(A,HI,A) DOESN'T WRITE B-SIDE 7477 K4/99/.. K408=SP-REWRITE-CNTL,K406/7=A/B-SPADS			
370)	6674	TEST507F	12265	7401		BUTR(INTR-HIGH), INTR-HIGH-H="0"; INTERNAL SERVICE SET 7403 K2/65 K3/35/.. FI; K2=E72; K3=E44			
371)	6675	TEST510H	12396	7407		BUTR(BG/FP/SERVICE), NEGATED="0" WHEN PS PRIO<7:5>=(7) 7277 K2/99/.. FI; K2=E43 7377 K2/65 K3/35/.. FI; K2=E16-E18,E44-E45,E67-E69; K3=E3,E7-E8,E25 7417 K3/99/.. FI; K3=E50,E76			
372)	6677	TEST503A	11238	4332		NUA SEQUENCING ERROR; BUTA(DIAGNOS) TO B.M. "LOADNEW4", "LOADNEW5" 0000 K2/99/.. FI; K2=E11,E17,E23,E29,E32,E38,E44,E50,E53-E54,E57, E59,E63,E68,E68,E72,E74,E78,E80,E92-E93 0127 K2/75 K3/25/.. FI; K2=E35,E47,E72,E78; K3=E111 0332 K2/99/.. FI; K2=E63,E80 0732 K2/99/.. FI; K2=E8,E11,E15,E17,E23,E29,E32,E38,E44,E50,E53, E59,E65,E68,E74,E80 4032 K2/99/.. FI; K2=E30 4126 K2/99/.. FI; K2=E6,E12,E18,E24,E30,E33,E39,E45,E51,E54,E60, E66,E69,E75,E81 4322 K2/99/.. FI; K2=E81 4330 K2/99/.. FI; K2=E81 4331 K2/99/.. FI; K2=E81 4335 K2/99/.. FI; K2=E81 4336 K2/99/.. FI; K2=E80 4337 K2/99/.. FI; K2=E57,E78			

[Continued]

Module codes: K1/DCS K2/WORD K3/TPDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/# #2/# #3/#			Test summary - Print reference - Chip information
6677	TEST503A	(Continued)				NUA SEQUENCING ERROR; BUTA(DIAGNOSE) TO B,M, 'LOADNZW4', 'LOADNZW5'			
					4352	K2/99	../. ..	../. ..	FI; K2=E30
					4372	K2/99	../. ..	../. ..	FI; K2=E30
					4375	K2/99	../. ..	../. ..	FI; K2=E92
					4732	K2/99	../. ..	../. ..	FI; K2=E5, E11, E17, E21, E23, E29, E32, E38, E44, E50, E53, E59, E65, E68, E74, E80
					4747	K3/99	../. ..	../. ..	FI; K3=E1-E20, E24-E26
					7005	K2/99	../. ..	../. ..	FI; K2=E5, E11, E15, E17, E29, E32, E50, E53-E54, E63, E68, E69, E74, E80
373	6700	TEST503E	11407	7402		EXFLAG<2>H="01", READ THRU CUA-PORT, PROC-MUX			
					7400	K2/85	K3/15	../. ..	FI; K2=E92, E101, E103, E106, E108, E111-E112, E119, K3=E48
					7403	K2/99	../. ..	../. ..	FI; K2=E108
374	6701	TEST372A	9658	5555		BUTA(SUBR-A)/BUTA(RETURN) SEQUENCE; D<14:03> -> RETURN; "1011 0110 1101"=(5555			
					4555	K2/99	../. ..	../. ..	FI; K2=E77, E89, E92, E95, E102, E106
					5155	K2/99	../. ..	../. ..	FI; K2=E6, E12, E18, E24, E30, E33, E39, E42, E45, E51, E54, E60, E66, E69, E75, E77, E81, E89, E92-E93, E96, E106-E107
					5355	K2/99	../. ..	../. ..	FI; K2=E92, E107
					5455	K2/99	../. ..	../. ..	FI; K2=E3, E6, E12, E18, E24-E25, E30, E33, E39, E44, E51, E54, E60, E66, E69, E75, E81, E89, E92-E93, E96, E99, E101
					5515	K2/99	../. ..	../. ..	FI; K2=E3, E6, E12, E18, E24-E25, E30, E33, E39, E45, E51, E54, E60, E66, E69, E75, E81, E86-E87, E89, E96, E98, E101
					5557	K2/99	../. ..	../. ..	FI; K2=E9, E28, E66, E114
					7555	K2/99	../. ..	../. ..	FI; K2=E72, E77, E92, E102
375	6702	TEST504G	11762	7406		BUTM(EXFLAG1), EXFLAG<1>H="0"			
					7407	K2/99	../. ..	../. ..	FI; K2=E112
376	6704	TEST410B	10814	7434		IR PATTERN LOOP; D<14:00>=ZERO NEGATED CORRECT # TIMES			
					7400	K4/65	K3/35	../. ..	FI; K4=E24, E33; K3=E84
377	6705	TEST410F	10911	7402		IR PATTERN LOOP; "BYTE-CONSTANT/SECOND-1-OR-2" ASSERTED CORRECT # TIMES			
					7400	K4/99	../. ..	../. ..	FI; K4=E33
					7401	K3/99	../. ..	../. ..	FI; K3=E49, E60
378	6706	TEST504H	11783	7406		BUTR(FPS06), FPS<05>H="0"			
					7407	K3/99	../. ..	../. ..	FI; K3=E54
379	6707	TEST410C	10844	7402		IR PATTERN LOOP; "BYTE-H" ASSERTED/NEGATED CORRECT # TIMES			
					7400	K3/65	K5/35	../. ..	FI; K3=E85; K5=E63
					7401	K6/85	K7/15	../. ..	FI; K6=E60, E78-E80; K7=E10
380	6710	TEST503F	11433	7406		BUTM(EXFLAG2), EXFLAG<2>H="0"			
					7407	K2/99	../. ..	../. ..	FI; K2=E112

Module codes: K1/DCS K2/UMWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

###	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/# #2/# #3/#			Test summary - Print reference - Chip information
381	6711	TEST410A	10790	7422		IR PATTERN LOOP; D<14:00>=ZERO ONLY ASSERTED 2. TIMES			
					7420	K2/99	../. ..	../. ..	FI; K2=E40, E88-E89
					7421	K4/65	K3/35	../. ..	FI; K4=E99; K3=E66
					7424	K2/99	../. ..	../. ..	FI; K2=E77, E88-E89, E95, E99, E101, E106
					7436	K2/99	../. ..	../. ..	FI; K2=E88-E89, E106
382	6712	TEST505B	11891	7407		EXFLAG<2> NOT CLEARED BY BUTA(CLR-FLAG-RES-UCON)			
					7406	K2/99	../. ..	../. ..	K207=ACTIVE-BUT, K210=FLAGS
383	6713	TEST371B	9538	7401		SR/NOP HAS NO EFFECT ON GUARD="0100"			
					7477	K4/99	../. ..	../. ..	K408=SR/RES/GUARD-LOGIC
384	6714	TESTA503A	11292	7402		LOAD/READ FLAGS:FPS=(125252) FLAG-PORT PROC-MUX; UCON FUNCTION			
					4777	K3/99	../. ..	../. ..	FI; K3=E41, E52, E111
					7400	K3/60	K2/40	K7/5	K5/5 FI; K3=E1-E20, E22-E27, E29, E33, E35, E41, E47, E51, E54, E94; K2=E4, E10, E16, E22, E28, E31, E36-E37, E43, E49, E52, E58, E62, E64, E67, E73, E79, E82, E98-E99, E101-E103, E106-E108, E110-E111, E113-E114, E117, E119; K7=E42, E44; K5=E84
					7401	K2/85	K7/20	K5/15	K3/10 FI; K2=E62, E71, E82-E83, E100-E101, E106, E108, E117; K7=E42, E44, E51, E79; K5=E15, E59, E67, E70; K3=E57, E61
					7403	K2/65	K3/25	K4/10	FI; K2=E95, E106, E108, E118; K3=E41; K4=E33
385	6715	TEST371A	9490	7434		SR/NOP HAS NO EFFECT ON SR=(052525)			
					7400	K4/99	../. ..	../. ..	FI; K4=E36, E80
					7433	K4/99	../. ..	../. ..	FI; K4=E35
386	6716	TEST504I	11804	7403		BUTM(FLTPT-FD-H), FLTPT-FD-H="1"			
					7402	K3/50	K2/50	../. ..	FI; K3=E64, E86; K2=E119
387	6717	TEST370D	9467	7401		SR/LEFT-GUARD/ENB SHIFTS GUARD LEFT, GD<3:0>="0100" AFTER			
					7400	K4/99	../. ..	../. ..	FI; K4=E53
388	6720	TEST503G	11453	7407		BUTM(EXFLAG1), EXFLAG<1>H="1"			
					4756	K3/99	../. ..	../. ..	FI; K3=E61
					7406	K2/65	K4/15	K3/18	FI; K2=E112; K4=E86; K3=E54
389	6721	TEST370C	9429	7434		SR/LEFT-GUARD/ENB, GD<3>="1"; SR=(052525) AFTER			
					7477	K4/99	../. ..	../. ..	K408=SR/RES/GUARD-LOGIC
390	6722	TEST505C	11912	7406		EXFLAG<1> CLEARED BY BUTA(CLR-FLAG-RES-UCON)			
					7406	K2/99	../. ..	../. ..	K207=ACTIVE-BUT, K210=FLAGS

Module codes: K1/DCS K2/UMWORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
391)	6723	TEST370B	9405	7402	SR/LEFT-GUARD/ENB SHIFTS GUARD LEFT, GD<3;0>="1010" AFTER	7400	K4/99	../..	FI; K4=E30
					7401	K2/99	../..	../..	FI; K2=E83
392)	6724	TEST507B	12201	7403	BUTR(P815), P8<15>H="1"	7402	K8/99	../..	FI; K5=E84
393)	6725	TEST370A	9360	7434	SR LEFT 1, SR=(125252) AFTER; GUARD/ENB="1010" AFTER	7400	K2/85	K4/10	K7/8
					7401	K4/99	../..	../..	FI; K2=E46,E55,E61,E83,E105; K4=E80,E87; K7=E28
394)	6726	TEST500C	11100	7402	IR PATTERN LOOP; 'OVERLAP=L' ASSERTED TO ASPLO CORRECT # TIMES	7400	K3/80	K4/10	K7/5
					7401	K3/99	../..	../..	FI; K3=E46,E59,E80,E84,E87,E89,E116-E117; K4=E25,E96,E99; K7=E69; K2=E55,E105
395)	6727	TEST410D	10879	7402	IR PATTERN LOOP; "BYTE-CONSTANT/FIRST-1-OR-2" ASSERTED CORRECT # TIMES	7400	K3/80	K5/5	K4/8
					7401	K3/96	K4/5	../..	FI; K3=E50,E59-E60,E79,E89,E116-E117; K5=E63,E66; K4=E15
396)	6730	TEST504B	11647	7407	BUTM(FLAG7), FLAG<7>H="1"	7406	K2/99	../..	FI; K2=E106
397)	6731	TEST367A	9317	7401	SR/LOAD-GUARD/DISABLED SETUP BY BUTA(CLR=RES), GUARD NOT ALTERED	7477	K4/99	../..	../..
398)	6732	TEST503H	11474	7407	BUTR(FP808), FP8<05>H="1"	7406	K2/99	../..	FI; K2=E7,E13,E25,E110,E117
399)	6733	TEST366C	9278	7434	SR LEFT 1, DIC="1"; SR=(025251) AFTER; (GUARD/DIS)	7400	K4/99	../..	../..
					7402	K4/99	../..	../..	FI; K4=E36
					7420	K4/99	../..	../..	FI; K4=E45
					7421	K4/99	../..	../..	FI; K4=E35
					7421	K4/99	../..	../..	FI; K4=E52,E71,E86
400)	6734	TEST506D	12076	7406	BUTM(MASK-PS(T)), MASK-PS(T)=0; H="0"	7407	K2/99	../..	FI; K2=E72,E78,E105,E112
401)	6735	TEST366B	9256	7401	CHECK GUARD NOT ALTERED ON SR/LEFT-GUARD/DISABLED SHIFT	7402	K4/99	../..	FI; K4=E30,E80,E86
402)	6736	TEST500E	11164	7402	IR PATTERN LOOP; BUTR(PREFETCH(01H)) ASSERTED CORRECT # OF TIMES	7400	K3/99	../..	../..
					7401	K3/90	K2/10	../..	FI; K3=E32,E38,E44,E48,E80; K2=E105

Module codes: K1/DCS K2/UMOPD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

##	ERROR code	Symbolic label	Line number	ENUA	TNUA	->Module sequence-> #1/% #2/% #3/%			Test summary - Print reference - Chip information
403)	6737	TEST366A	9209	7434	SR LEFT 1, DIC="0"; SR=(112524) AFTER; (GUARD/DIS)	7005	K4/99	../..	FI; K4=E80
					7400	K4/99	../..	../..	FI; K4=E45
					7421	K4/99	../..	../..	FI; K4=E52,E71,E86
404)	6740	TEST504C	11667	7403	BUTR(FLTP-T-PROC-H), FLTP-T-PROC-H="1"	7402	K2/65	K3/35	../..
					7402	K2/65	K3/35	../..	FI; K2=E101,E106,E119; K3=E64
405)	6741	TEST365B	9187	7401	CHECK GUARD NOT ALTERED ON SR/RIGHT-GUARD/DISABLED SHIFT	7402	K4/99	../..	FI; K4=E30,E53,E80
406)	6742	TEST503I	11495	7402	BUTR(FLTP-T-FD-H), FLTP-T-FD-H="0"	7403	K3/99	../..	FI; K3=E64
					7756	K2/99	../..	../..	FI; K2=E40
407)	6743	TEST365A	9146	7434	SR RIGHT 1, DIC="1"; SR=(145252) AFTER	7400	K4/99	../..	FI; K4=E10,E17,E36,E45
					7420	K4/99	../..	../..	FI; K4=E52
408)	6744	TEST506C	12054	7407	BUTM(P803), P8<03>H="1"	7406	K2/80	K3/20	../..
					7406	K2/80	K3/20	../..	K210=MULTIPLE-BUT; K306=PS(ICC)
409)	6745	TEST364R	9119	7401	BUT(GD3=2), GD<3;0>="0101" AFTER SR/RIGHT, SR<00>="0"	7477	K4/99	../..	../..
					7477	K4/99	../..	../..	K408=SR/RES/GUARD-LOGIC
410)	6746	TEST500D	11136	7402	IR PATTERN LOOP; 'OVERLAP=L' ASSERTED SAME # TIMES TO ASPLO AND BSPLO	7401	K4/99	../..	FI; K4=E99
411)	6747	TEST364A	9091	7402	BUT(GD3=2), GD<3;0>="1010" AFTER SR/RIGHT, SR<00>="1"	7477	K4/99	../..	../..
					7477	K4/99	../..	../..	K408=SR/RES/GUARD-LOGIC
412)	6750	TEST504D	11688	7475	BUT(INSTR1), IN=(172525), CLASS=A(FLTP-T), FLAG<4;5>H="01"	7477	K3/99	../..	../..
					7477	K3/99	../..	../..	K305=FLAG8/FP-INSTR1
413)	6751	TEST363B	9069	7401	BUT(GD3=2), GD<3;0>="0100" AFTER SR/RIGHT, SR<00>="0"	7400	K3/60	K4/40	../..
					7403	K4/99	../..	../..	FI; K3=E54; K4=E53
414)	6752	TEST534G	13844	7402	R.M. COUNTER CLEARED TO ZERO BY BUTA(LAST)	7400	K4/65	K2/35	../..
					7637	K2/70	K3/30	../..	FI; K4=E87,E90-E91; K2=E55
					7637	K2/70	K3/30	../..	FI; K2=E22-E24,F49-E51,E73-E75; K3=E6,E9,E13,E19
415)	6753	TEST363A	9033	7434	SR RIGHT 1, DIC="0"; SR=(052525) AFTER	7400	K4/99	../..	FI; K4=E10,E17,E24,E45,E74,E85,E94
					7410	K4/99	../..	../..	FI; K4=E47

[Continued]

Module codes: K1/DCS K2/UMOPD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRPR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	---	-----	-----	----	----	-----	-----
	6753	TEST363A	[Continued]			SR RIGHT 1, D<00>="0"; SR=(052825) AFTER 7417 K4/99/.. FI; K4=E36 7420 K4/99/.. FI; K4=E52 7426 K4/99/.. FI; K4=E35	
416)	6754	TEST506B	12034	7402		BUTR(P815), PS<15>H="0" 7403 K3/99/.. FI; K3=E54	
417)	6754	TEST362F	9010	7402		BUT(GD3-2), GD<310>="1000" AFTER SR/RIGHT, SR<00>="1" 7400 K4/80 K3/15 K2/5 FI; K4=E26,E28-E30,E47,E53,E55,E71,E90,E96; K3=E1-E3,E12-E13,E56; K2=E39,E54	
418)	6756	TEST503J	11513	7406		BUTM(D00), D=(000000) 7407 K2/99/.. FI; K2=E112	
419)	6757	TEST362D	8989	7405		BUTR(SR1-08COUNT), SP<110>="10" 4747 K2/99/.. FI; K2=E46,E55 7477 K4/99/.. K408=SR/RES/GUARD-LOGIC	
420)	6760	TEST504E	11716	7402		EXFLAG8<211>H="10", READ THRU CUA-PORT, PROC-MUX 7400 K2/99/.. FI; K2=E103,E106,E111-E112	
421)	6761	TEST702R	14089	7402		LOAD BA<17118>="10", 18, BIT MODE, READ THRU STATUS-MUX(SERVICE)<918> 7400 K6/55 K4/25 K7/10 K5/10 FI; K6=E11,E51-E54; K4=E65; K7=E56; K5=E26, E60	
422)	6762	TEST520D	13417	7402		INSTR-BRANCH-L="L", IR=(101401)=BLOS, PS(NZVC)=(01) 7407 K3/99/.. K2=PS(CC)-BRANCH(E64,72-73,83)	
423)	6763	TEST362R	8962	7434		SR=(125252), D(C)="0"; SR/XMUX=(000052) (FLTPT) 7400 K4/99/.. FI; K4=E46,E55,E78 7420 K4/99/.. FI; K4=E37 7421 K4/99/.. FI; K4=E44 7424 K4/99/.. FI; K4=E44	
424)	6764	TEST520B	13360	7402		INSTR-BRANCH-L="L", IR=(003000)=BGT, PS(NZVC)=(00) 7403 K3/99/.. FI; K3=E64,E83 7407 K3/99/.. K2=PS(CC)-BRANCH(E64,72-73,83)	
425)	6764	TEST362A	8917	7434		SR RIGHT 1, D<00>="1"; SR=(128252) AFTER 7400 K4/99/.. FI; K4=E3,E13,E22,E24,E29,E35-E36,E45,E52,E55,E74, E76,E85,E94,E104	
426)	6766	TESTA504A	11621	7402		LOAD/READ FLAG=FPS=(052525) FLAG-PORT PROC-MUX; UCON FUNCTION 7400 K2/90 K3/10/.. FI; K2=E78,E98-E99,E101-E103,E105-E108,E110-E114, E119; K3=E27,E47,E94,E96 7403 K2/99/.. FI; K2=E106,E108	

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUS

FRPR	code	Symbolic label	Line number	ENUA	TNUA	Module sequence	Test summary - Print reference - Chip information
---	---	-----	-----	----	----	-----	-----
427)	6767	TEST361E	8894	7400		BUT(GD3-2), GD<310>="0000" AFTER ENABLED, SR/LOAD 7401 K4/80 K2/15 K3/5 FI; K4=E17,E28,E30,E53,E80,E86-E87; K2=E33,E45, E51; K3=E54 7402 K4/90 K3/10/.. FI; K4=E26,E30,E53,E87; K3=E56 7404 K3/99/.. FI; K3=E74	
428)	6770	TEST701C	15981	7402		LOAD BA<15100>=(125252), READ THRU BA/KT-ALU/PBA/STATUS-MUX(PBA) 7400 K5/50 K4/15 K6/15 K7/15 FI; K5=E7,E9-E11,E27,E37-E38,E43,E45-E47, E80,E59,E69-E70,E72,E79-E82,E94; K4=E39,E48,E57; K6=E19,E28,E29,E33,E49,E59; K7=E56,E64-E66,E73, E80-E82,E88 7401 K7/50 K4/50/.. FI; K7=E88; K4=E29 7403 K5/65 K6/20 K7/10 FI; K5=E7,E26-E27,E46-E47; K6=E25,E59; K7=E80	
429)	6771	TEST361D	8873	7403		BUTR(SR1-08COUNT), SR<110>="01" 6377 K2/65 K3/35/.. FI; K2=E4-E5,E58-E59; K3=E1,E16 7407 K3/99/.. FI; K3=E74 7477 K4/99/.. K408=SR/RES/GUARD-LOGIC	
430)	6772	TEST520C	13388	7403		INSTR-BRANCH-L="H", IR=(002416)=BLT, PS(NZVC)=(16) 7402 K3/99/.. FI; K3=E72,E83 7407 K3/99/.. K2=PS(CC)-BRANCH(E64,72-73,83)	
431)	6773	TESTD410	10707	7361		(NVA SEQUENCING LOGIC ERROR) 6777 K2/99/.. FI; K2=E48 7777 K1/99/.. INTERNAL DCS ERROR	
432)	6775	TEST520F	13445	7403		INSTR-BRANCH-L="H", IR=(103406)=BLO, PS(NZVC)=(06) 7402 K3/99/.. FI; K3=E83 7407 K3/99/.. K2=PS(CC)-BRANCH(E64,72-73,83)	

Module codes: K1/DCS K2/UNORD K3/IRDECODE K4/DATAPATH K5/KTCACHE K6/TIMING K7/STATUSKD11-K MICRO V00A-1

I D E N T I F I C A T I O N

PRODUCT CODE: MAINDEC-11-DQKUB-B0
PRODUCT NAME: KD11-K Microdiagnostic
MAINTAINER: Diagnostic Engineering
AUTHOR: Don North
DATE CREATED: 19-January-1977
LAST REVISION: 15-June-1977, Version /101/

COPYRIGHT (C) 1976, 1977; DIGITAL EQUIPMENT CORPORATION
146 MAIN STREET
MAYNARD, MASSACHUSETTS, USA
01754 617-897-5111

THIS SOFTWARE IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR
USE ON A SINGLE COMPUTER SYSTEM, AND CAN BE COPIED (WITH INCLU-
SION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM,
EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT
NOTICE, AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO
RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF
ITS SOFTWARE ON EQUIPMENT NOT SUPPLIED BY DIGITAL.

TABLE OF CONTENTS

59	--	IDENTIFICATION
94	--	REVISION HISTORY
106	--	MICROWORD FIELD DEFINITIONS
129	--	MICROWORD BIT LAYOUT
187	--	A & B & C SCRATCHPAD LAYOUT, DCS SPECIFIC
243	--	MICROWORD FIELD SPECIFICATION
248	--	MICROWORD FIELD FORMAT
260	--	NULL FIELD/MACRO SPECIFICATION
266	--	ALU AND INTERNAL DATA BUS CONTROL
270	--	<ALU>-ALU FUNCTION CONTROL BITS
295	--	<BEN>-B-BUS DATA SOURCE
305	--	<BSEL>-B-BUS SOURCE SELECTION CONTROL
345	--	<AEN>-A-BUS DATA SOURCE
355	--	<ASEL>-A-BUS SOURCE SELECTION CONTROL
398	--	<RIF>-ASP, BSP REGISTER IMMEDIATE FIELD
435	--	<COUT>-CARRY OUT BIT MUX SELECTION
451	--	CLOCKS
455	--	<WHEN>-D/SP WHEN TO CLOCK
463	--	<CLKD>-ENABLE D-REGISTER CLOCKING
471	--	<CLKSR>-ENABLE SR-REGISTER CLOCKING
479	--	<CLKBA>-ENABLE CLOCKING OF BA-REGISTER
487	--	<SCC>-ENABLE SETTING OF PS CONDITION CODES
498	--	BUS/UCON & CSP-ADDRESS & SHIFT-TREE CONTROL
502	--	BUS/UCON CONTROL
505	--	<BEGIN>-BEGIN BUS/UCON OPERATION
513	--	<SELECT>-SELECT BUS OR UCON
521	--	BUS CONTROL
524	--	<BUSCODE>-BUS CODE ACTION FIELD
540	--	UCON CONTROL
544	--	<FLPGO>-START HOT FLOATING POINT
552	--	<UCON-XFER>-UCON OPERATION
560	--	<UCON-LOAD>-LOAD UCON REGISTER
568	--	CSP ADDRESS SPECIFICATION
571	--	<CSPADDR>-CSP IMMEDIATE ADDRESS
597	--	SHIFT CONTROL
600	--	<BMUX>-SECOND LEVEL OF SHIFT TREE
608	--	<AMUX>-FIRST LEVEL OF SHIFT TREE
625	--	SP REWRITE & REGISTER CLOCKS
629	--	<WRCSP>-WRITE TO CSP
637	--	<MOD>-MODE CONTROL OF FOLLOWING BITS
645	--	SP REWRITE (A,B) CONTROL
649	--	<HILO>-SP HI/LO SELECT
659	--	<WRSEL>-REWRITE ADDRESS SELECT
669	--	<WRSP>-REWRITE A/B SELECT
688	--	REGISTER LOADING
692	--	<LOADRES>-LOAD RESIDUAL CONTROL REGISTER
701	--	<LOADCOUNT>-LOAD COUNTER
711	--	SFUENCING FIELD
715	--	<UBF>-BUT MICROBRANCH FIELD
719	--	NO BUT

TABLE OF CONTENTS

722	--	ACTIVE ONLY
739	--	INACTIVE ONLY
810	--	BOTH ACTIVE AND INACTIVE
831	--	<UPF>-MICRO POINTER FIELD
874	--	MISCELLANEOUS FIELDS
878	--	<NEXT-PAGE>-NEW PAGE ADDRESS LOADED DURING BUT(SUBROUTINE)
884	--	<MULTIPLE>-SELECT CODE FOR BUT(MULTIPLE)
898	--	EMIT FIELD - IMMEDIATE DATA FROM MICROWORD
925	--	RETURN ADDRESS - FOR MICROSUBROUTINE CALLS
931	--	UCON SELECTION AND CONTROL FIELDS
934	--	SELECTION
957	--	CONTROL
979	--	BASE MACHINE EXTENSION BITS
1052	--	SPECIAL DCS FIELDS
1056	--	FIELDS USED IN PAGES 4, 5, OR 6 OF DCS
1059	--	<LOAD-DCS-CTR>-LOAD DIAGNOSTIC COUNTER FROM EMIT
1072	--	<CTR>-4 BIT DCS COUNTER VALUE FROM EMIT
1096	--	<LOAD-ENUA-ERRCOD>-LOAD THE ENUA AND ERRCOD REGISTERS
1106	--	<ENUA>-ENUA VALUE FROM EMIT
1113	--	<VERIFY>-VERIFY BIT FOR SELF CHECK TEST
1125	--	FIELDS USED IN PAGE 7 OF DCS EXTENSION
1128	--	<EOP>-SIGNAL SUCCESSFUL END OF PASS
1137	--	<DAD>-DCS CONTROL OF BASE MACHINE EXTENSION DAD BITS
1149	--	FIELDS USED IN ALL PAGES OF DCS EXTENSION
1152	--	<SCOPE>-SCOPE ON ERROR, DIAGNOSTIC BUT
1170	--	MACRO DEFINITIONS
1173	--	PRIMITIVE OPERATIONS
1176	--	TIMING
1210	--	WRITING THE A AND B SCRATCH PADS
1234	--	ASP AND BSP PHYSICAL REGISTER ADDRESSES
1266	--	ASP AND BSP BASE MACHINE FUNCTIONAL REGISTER ADDRESSES
1339	--	ASP AND BSP INDIRECT REGISTER ADDRESSES
1364	--	ASP, BSP INDIRECT ADDRESSING
1377	--	ASP AND BSP DCS SPECIFIC FUNCTIONAL REGISTER ADDRESSES
1401	--	WRITING THE C SCRATCH PAD
1410	--	CSP IMPLIED ADDRESSING
1422	--	CSP DIRECT ADDRESSING
1436	--	SHIFT TREE SPECIFICATION
1440	--	ENABLED ONTO BUS A
1471	--	FIRST TWO LEVELS ONLY (AMUX,BMUX)
1479	--	ALU FUNCTIONS
1500	--	COUT GENERATION
1515	--	CLOCKS
1519	--	BASIC REGISTER CLOCKS (D, SR, BA, CC)
1528	--	REDEFINED FROM SP REWRITE FIELD (RES, COUNTER)
1534	--	RES REGISTER CONTROL VALUES (FROM EMIT)
1548	--	CC CONTROL (FROM EMIT)
1557	--	RUS CONTROL MACROS
1571	--	KT/KJ CONTROL FUNCTIONS
1599	--	UCON CONTROL MACROS

TABLE OF CONTENTS

1595	--	PROCESSOR UCON CONTROL SETUP
1613	--	DCS/WCS/ECS CONTROL
1621	--	CACHE/KT UCON CONTROL
1641	--	I/O UCON CONTROL
1646	--	BUS CONTROL
1666	--	CONSOLE I-O
1682	--	REMOTE CONSOLE INTERFACE
1692	--	DCS ROM EXTENSION MACROS
1694	--	GENERAL FUNCTIONS
1705	--	DAD<1:0> BIT FUNCTIONS
1714	--	DIAGNOSTIC MODE BUT ENABLES
1732	--	MICROBRANCH FIELD MACROS
1748	--	MISCELLANEOUS
1750	--	OTHER SOURCES ENABLED FOR A-BUS
1756	--	PAGING, RETURN REGISTER
1771	--	ADVANCED OPERATIONS
1775	--	DATA INTO CSP, AT P3 ONLY
1821	--	MISC CONSTANTS INTO ASP, BSP, AT P2-T * P3
1849	--	DATA INTO ASP, BSP, AT P2-T * P3
2055	--	D AND SR <- (BUS-A FCN BUS-R), AT P2-T OR P3-T
2098	--	DIC GETS SET
2116	--	D-REGISTER <- (BBUS = ABUS), BITWISE, AT P2-T OR P3-T
2153	--	D-REGISTER <- D-REGISTER THRU SHIFT-TREE
2187	--	D <- WHATEVER'S LEFT, AT P2-T OR P3-T
2234	--	SR <- DATA, AT P2 T OR P3 T
2269	--	RES-REG OPERATION MACROS
2278	--	BASE MACHINE COUNTER
2286	--	ENABLE ON BUS-A/R ONLY
2312	--	LOADING BA REGISTER
2325	--	D AND SR TOGETHER
2333	--	UCON FUNCTIONS
2337	--	PROCESSOR UCON FUNCTIONS
2364	--	CACHE/KT UCON FUNCTIONS
2390	--	I-O UCON FUNCTIONS
2406	--	DCS UCON FUNCTIONS
2413	--	CONSOLE UCON FUNCTIONS
2428	--	DBUF UCON FUNCTIONS
2437	--	MULTIPLE UCON FUNCTIONS
2452	--	SPECIFIC MACROS FOR PREFETCH/OVERLAP/SP-INHIBIT TESTS
2479	--	SPECIFIC MACROS FOR BYTE/BYTE CONSTANT/D=ZERO TESTS
2498	--	SUBROUTINE CALL MACROS
2568	--	JAM UPP LOG MACROS
2585	--	- - - MICRODIAGNOSTIC CODE - - - - -
2602	--	TEST001-007: NUA SEQUENCING
2775	--	TEST010-011: MICROSUBROUTINE OPERATION
2906	--	INIT REGISTERS, CONSOLE DEFAULT ERROR DISPLAY, REVISION CODE
2996	--	TEST012-050: IR DECODE (INSTR1, INSTR5, FLIPT, RELATED "BUTS")
5139	--	TEST101: D -> DBUF -> IR PATH
5188	--	TEST102-104: TESTING CSP ADDRESS/READ/WRITE FUNCTIONS
5573	--	TEST105: SR CAN LOAD/STORE AS A REGISTER

TABLE OF CONTENTS

5772	--	TEST114-121: ALU LOGIC TESTS / D(C) TESTS
7236	--	TEST130-136: ALU ARITHMETIC FUNCTION/CARRY LOOKAHEAD TESTS
8078	--	TEST320: D(C) SELECTION / COUNT7-DOUT07 / D<14:00>=ZERO@BIT<00>
8286	--	TEST350-352: ASP/BSP HI/LO ADDRESSING MODES, DATA VALIDITY
8800	--	TEST361-372: TESTING SR, GUARD, RES, AND XMUX
9585	--	TEST372A-372B: TESTING CUA, PROCESSOR MUX, AND BUTA(SUBR A)
9712	--	TEST373: CHECK JAMUPP W/ BUTA(DIAGNOSE), B ⁴ EXT BIT FLPADR
9840	--	TEST374: A/B SP PEWRITE MODES VERIFICATION
10488	--	TEST375-376: BYTE WRITE TO ASP/BSP LO, SP ADDRS R-OR-1/FLTPT-INHIBIT
10645	--	TEST410: BYTE/BYTE CONSTANT/D=ZERO
10954	--	TEST500: PREFETCH/OVERLAP/SP DEFEAT
11206	--	TEST503-510: PROCESSOR UCON TESTS (FLAGS, FPS, PS, BUTM) & ASSOC LOGIC
12636	--	TEST511: MFSS LOGIC TESTS
12868	--	TEST512: KT SRC/DST ADDRESSING LOGIC TESTS
13288	--	TEST520A-520E: TESTING THE "INSTR BRANCH" ROM
13506	--	TEST533-537: SHIFT TREF
14448	--	TEST551: BASE MACHINE DATAPATH COUNTER CAN COUNT
14702	--	TEST610: CONDITION CODE LOGIC
15234	--	TEST620-624: TESTING UBREAK AND JAMUPP
15901	--	TEST701-702: LOAD/READ THE RA, FULL 18. BITS
16132	--	TEST710-722: BUS FUNCTION DECODE, BUS ERROR CONDITIONS
17056	--	TEST730-731: BUS CYCLES TO/FROM MEMORY
17523	--	TEST740: BUS CYCLE MODIFICATION - PREFETCH ALTERATION, OVERLAP YANK
17687	--	TEST761-763: TESTING UNIBUS INTERRUPT SERVICE WITH DL11-W LINE CLOCK
18191	--	END OF PASS CODE
18294	--	VERIFY MODE CODE
18352	--	DCS MICROCODE REVISION NUMBER
18394	--	COMMON SUBROUTINES
18396	--	CONSOLE DISPLAY SUBROUTINE
18445	--	CLEAR I-O / BUS CONTROL / SERVICE AREA STATUS LATCHES SUBR
18491	--	SUBR FOR PUTTING SELECTED PORTIONS OF D[15-00] INTO IR
18570	--	UCON SUBROUTINES (FLAGS, PS, FPS, CUA, SERVICE, JAM, PBA)
18691	--	SUBR FOR LOADING FPS<3:0> (VIA BUTA(DIAGNOSE))
18738	--	SUBR TO COPY D-REGISTER TO DBUF TO IR
18810	--	JAM UPP SERVICE SUBROUTINE
18894	--	MICROBRANCH (BUT) TAKEOFF WORDS
19233	--	MICROBRANCH (BUT) TARGET WORDS
19766	--	END OF WD11-K MICRODIAGNOSTIC CODE

55
56
57 !.PAGE=====

58
59 .TOC * IDENTIFICATION

60 !
61 .TITLE KD11-K MICRODIAGNOSTIC

```

62
63
64
65
66
67 |*****|
68 |
69 |           K   K   DDDDDD           1   1           /   K   K
70 |           K   K   D   D           11  11           K   K
71 |           K   K   D   D           1 1  1 1           K   K
72 |          KKKK   D   D   XXX       1   1           KKKK
73 |           K   K   D   D           1   1           K   K
74 |           K   K   D   D           1   1           K   K
75 |           K   K   DDDDDD          11111 11111 /   K   K
76 |
77 |
78 |          DDDDDD  IIIII  AAA  GGGGG  N   N  00000  SSSSS  TTTTTT  IIIII  CCCCC
79 |          D   D   I   A   A   G   G   NN   N  O   O   S   S   T   I   C
80 |          U   U   D   D   I   A   A   G   G   NN   N  O   O   S   S   T   I   C
81 |          U   U   D   D   I   A   A   G   GGG  N   N  N  O   O   SSSSS  T   I   C
82 |          U   U   D   D   I   AAAAAA  G   G   N   N  N  O   O   S   S   T   I   C
83 |          U   U   D   D   I   A   A   G   G   N   NN  O   O   S   S   T   I   C
84 |          U UUUUU U  DDDDDD  IIIII  A   A  GGGGG  N   N  00000  SSSSS  T   IIIII  CCCCC
85 |          U
86 |          U
87 |
88 |*****|

```

91
92 !.PAGE=====

93
94 .TOC * REVISION HISTORY

95
96 .IDENT /V101A0/

97
98 REV-NUMBER : 000101 !NO BIT15 DURING EXECUTION
99 REV-NUMBER-AND-B15 : 100101 !BUT SET W/REV. NUMBER AT EOP ONLY

100
101
102
103
104 !.PAGE=====

105
106 .TOC * MICROWORD FIELD DEFINITIONS

107
108 ! NOTE: THE FOLLOWING ARE THE ASSIGNED RANGES OF THE

```

109 1 MICROWORD FIELD BIT DEFINITIONS USED IN THIS
110 1 SOURCE LISTING:
111 1
112 1 BYTS(NUMBER) WHERE HELD
113 1 -----
114 1
115 1 [47:00] 6P/48 MAIN MACHINE ROM,
116 1 DCS MAIN ROM
117 1 [59:48] 12/12 * MAIN MACHINE ROM EXTENSION,
118 1 USES 12/12 BITS
119 1 [54:48] 7/12 * DCS 4-BIT ROM EXTENSION,
120 1 USES 7/12 BITS, [54:48]
121 1
122 1 * = NOTE OVERLAP OF BM EXTENSION/DCS EXTENSION BITS.
123 1 THERE BITS ARE MUTUALLY EXCLUSIVE.
124 1
125 1
126 1

```

```

127 1.PAGE=====
128 1
129 1.TOC * MICROWORD BIT LAYOUT
130 1 BASE 1-EMIT 6-RETURN 9-RES-BITS DCS UCON
131 1 MACHINE 2-SHIFT-TREE 7-PAGING 10-MULTIPLE EXTENSION PROCESSOR
132 1 CONTROL 3-RESIDUAL-CTL 5-CSP-ADDRESS 8-UCON-CONTROL 10-MULTIPLE CONTROL
133 1 -----
134 1 [147] ALU3 1-EMITH15 4-UCONL10 6-RETR11 9-HISMUXSEL1 DCS-CTR3 PS<3:0>-CLK
135 1 [146] ALU2 1-EMITH14 4-UCON-I/O-SEL 6-RETR11 9-HISMUXSEL1 DCS-CTR2
136 1 [145] ALU1 1-EMITH13 4-UCON-WCS-SEL 6-RETR10 9-SRS1-L DCS-CTR1
137 1 [144] ALU0 1-EMITH12 4-UCON-KT-SEL 6-RETR09 9-SRS0-L DCS-CTR0
138 1 [143] BEN1
139 1 [142] BEN0 4-UCONL09 6-RETR08 9-GUARD-EN-H ENUA11 UBREAK-CLK
140 1 [141] RSEL1 1-EMITH11 4-UCONL08 6-RETR07 ENUA10 <NU>
141 1 [140] RSEL0 1-EMITH10 4-UCONL07 6-RETR06 ENUA09 <NU>
142 1 [139] AEN1 1-EMITH09 4-UCONL06 6-RETR05 ENUA08 SEL-HBMUX1L
143 1 [138] AEN0 1-EMITH08 4-UCONL05 6-RETR04 ENUA07 SEL-HBMUX0L
144 1 [137] ASEL1 1-EMITH07 4-UCON-PROC-SEL 6-RETR03 ENUA06
145 1 [136] ASEL0 1-EMITH06 4-UCONM12 6-RETR02 ENUA05 FPS<7:4>-CLK
146 1 [135] RIF2 1-EMITH05 4-UCONM11 6-RETR01 ENUA04 PS<7:4>-CLK
147 1 [134] RIF1 1-EMITH04 4-UCON-FP-SFL 6-RETR00 ENUA03
148 1 [133] RIF0 1-EMITH03 4-UCONH15 7-NEXT-PAGE2 10-MULT-SEL2 ENUA02 1R-CLOCK
149 1 [132] COUT2 1-EMITH02 4-UCONH14 7-NEXT-PAGE1 10-MULT-SEL1 ENUA01 PS<15:12>-CLK
150 1 [131] COUT1 1-EMITH01 4-UCONH13 7-NEXT-PAGE0 10-MULT-SEL0 ENUA00 FLAG<8:0>-CLK
151 1 [130] COUT0 1-EMITH00
152 1 [129] * WHEN
153 1 [128] * CLK-D
154 1 [127] * CLK-SR
155 1 [126] * CLK-BA
156 1 [125] * SET-CC
157 1 [124] * BEGIN
158 1 [123] SSELECT(=0) 2-BMUX 5-CSPADR3 8-SELECT(=1)
159 1 [122] BUSCOD2 2-AMUX2 5-CSPADR2 8-FLPGO
160 1 [121] BUSCOD1 2-AMUX1 5-CSPADR1 8-UCON-KFER
161 1 [120] BUSCOD0 2-AMUX0 5-CSPADR0 8-UCON-LOAD
162 1 [119] * WRCSP

```

```

163 1 [118] HT/LO 3-LOAD-RES
164 1 [117] WRSEL
165 1 [116] WRB 3-LOAD-COUNT
166 1 [115] WRA
167 1 [114] * MOD(=0) 3-MOD(=1)
168 1 [113] * URF4
169 1 [112] * URF3
170 1 [111] * URF2
171 1 [110] * URF1
172 1 [109] * UPF0
173 1 [108] * UPF8
174 1 [107] * UPF7
175 1 [106] * UPF6
176 1 [105] * UPF5
177 1 [104] * UPF4
178 1 [103] * UPF3
179 1 [102] * UPF2
180 1 [101] * UPF1
181 1 [100] * UPF0
182 1
183 1
184 1
185 1
186 1
187 1
188 1
189 1
190 1
191 1
192 1
193 1
194 1
195 1
196 1
197 1
198 1
199 1
200 1
201 1
202 1
203 1
204 1
205 1
206 1
207 1
208 1
209 1
210 1
211 1
212 1
213 1
214 1
215 1
216 1

```

(* = DEDICATED TO THE CORRESPONDING SINGLE FUNCTION)

```

199 1.PAGE=====
200 1
201 1.TOC * A & B & C SCRATCHPAD LAYOUT, DCS SPECIFIC
202 1
203 1 THE USE OF THE A, B, & C SCRATCHPADS AS TEMPORARY STORAGE AREAS
204 1 IS OUTLINED BELOW. NOTE THAT IN MOST CASES, THE REGISTERS
205 1 EMPLOYED HAVE NO RESEMBLANCE TO ANY SIMILAR NAMED REGISTER
206 1 IN THE KD11-F BASE MACHINE MICROCODE, EITHER LIVING OR DEAD.
207 1
208 1 THE BELOW DEFINITIONS ARE -DCS SPECIFIC- ONLY.
209 1
210 1
211 1
212 1
213 1
214 1
215 1
216 1

```

OCTAL ADDRESS	ASPHI	ASPL0	BSPHI	BSPLO	CSP
00	...	...	...	...	<RETURN>
01	(000000)	...	(000000)	...	TEMP/(000152)
02	...	...	...	...	TEMP/(000125)
03	(177777)	...	(177777)	...	TEMP
04	...	...	...	...	TEMP/(125200)
05	(125252)	...	(125252)	...	TEMP/(052522)/(170360)
06	...	...	...	...	TEMP/(053433)/(007417)
07	(052525)	...	(052525)	...	TEMP
10	...	...	...	...	(052525)

```

217 1
218 1 11 (000001) . . . (000001) . . . (125252)
219 1
220 1 12 . . . . . . . . (177777)
221 1
222 1 13 (100000) . . . (100000) . . . (000000)
223 1
224 1 14 . . . . . . . . TEMP/(000100)/(087077)
225 1
226 1 15 (000200) . . . (000200) . . . TEMP/(000077)/(177067)
227 1
228 1 16 . . . . . . . . TEMP/(170000)
229 1
230 1 17 TEMPAMI TEMPALO TEMPBHI TEMPALO TEMP/(007700)
231 1
232 1
233 1

```

234 1 NOTES: (XXXXXX) ::= A REGISTER W/ A CONSTANT VALUE, NAMED:
235 1 CXXXXXX-A, IF ON THE A-SIDE, OR
236 1 CXXXXXX-B, IF ON THE B-SIDE
237 1

240 1
241 1 .PAGE=====

242 1 .TOC * MICROWORD FIELD SPECIFICATION

243 1
244 1
245 1
246 1
247 1
248 1 .TOC * MICROWORD FIELD FORMAT
249 1
250 1 .RADIX 8 I ALL NUMBERS ARE OCTAL, UNLESS OTHERWISE NOTED
251 1
252 1 .WIDTH 64R I MICROWORD IS 64\10 BITS WIDE, BIT <00> IS RIGHTMOST BIT
253 1
254 1 .ROUNDS (400017777) I ADDRESSES ARE 12 BITS, ON PAGES 4:7
255 1
256 1 .PROJECT <15100>*<31116>*<47132>*<63149> I OUTPUT FORMAT (DEFAULT SPEC)
257 1
258 1
259 1

260 1 .TOC * NULL FIELD/MACRO SPECIFICATION

261 1
262 1 .FIELD N1:=<63>
263 1 .MACRO NULL1:=N/0
264 1

265 1 .TOC * ALU AND INTERNAL DATA BUS CONTROL

266 1
267 1
268 1
269 1 .TOC * <ALU>-ALU FUNCTION CONTROL BITS
270 1

271 1 SPECIFIES ALU FUNCTION CODE AND CINMUX SELECT, ALWAYS IN EFFECT.
272 1 .FIELD ALU1:=<47:44>
273 1
274 1 NOT-A1:=00 ICOMPLEMENT A, L 0000 -1
275 1 A-PLUS-B-PLUS-PS[C]1:=01 IADD, A 1001 -PS[C]
276 1 NOT-A-AND-B1:=02 IAND, L 0010 -PS[C]
277 1 ZERO1:=03 IZERO, L 0011 -PS[C]
278 1 A-PLUS-B-PLUS-DIC1:=04 IPLUS, A 1001 -D[C]
279 1 A-PLUS-NOT-B-PLUS-DIC1:=05 IPLUS, A 0110 -D[C]
280 1 A-XOR-B1:=06 IXOR, L 0110 -D[C]
281 1 A-AND-NOT-B1:=07 IAND, L 0111 -D[C]
282 1 DIVIDE1:=10 IDIVIDE STEP, I
283 1 I SUB, IF D[C]W=1 A 0110 -D[C]=-1
284 1 I ADD, IF D[C]W=0 A 1001 -D[C]=-0
285 1 A-PLUS-R1:=11 IPLUS, A 1001 -0
286 1 B1:=12 ISELECT B, L 1010 -0
287 1 A-AND-B1:=13 IAND, L 1011 -0
288 1 A-PLUS-B-PLUS-11:=14 IPLUS, A 1001 -1
289 1 A-MINUS-B1:=15 IMINUS, A 0110 -1
290 1 A-IOR-B1:=16 IOR, L 1110 -1
291 1 A1:=17 ISELECT A, L 1111 -1
292 1
293 1
294 1

295 1 .TOC * <BEN>-B-BUS DATA SOURCE

296 1 SPECIFIES GATING OF DATA ONTO B-BUS, ALWAYS IN EFFECT.
297 1 .FIELD BEN1:=<43:42>
298 1 BSPLO1:=0 IDIRECT BSP LOCATIONS 00-17
299 1 BSBPH1:=1 IDIRECT BSP LOCATIONS 20-37
300 1 CSP1:=2 IUSE <CSPADDR> [SIC] AS ADDRESS (4 BIT)
301 1 BASCON1:=3 I1 OF 4 BASE CONSTANTS IN CSP17 TO CSP14 (2 BIT)
302 1
303 1
304 1

305 1 .TOC * <BSEL>-B-BUS SOURCE SELECTION CONTROL

306 1 SPECIFIES CONTROL OF INDIVIDUAL B-BUS SOURCES, ALWAYS IN EFFECT.
307 1 .FIELD BSEL1:=<41:40>
308 1 NOT USED WHEN BEN/CSP
309 1 ICSP17 TO CSP14 IMMEDIATE ADDRESS WHEN BEN/BASCON
310 1 B17:=0 I
311 1 R16:=1 I
312 1 B15:=2 I
313 1 R14:=3 I
314 1 IUSED IN CONJUNCTION WITH <RIF> FOR SP ADDRESS WHEN BEN/BSPL0 OR BEN/BSBPHI
315 1 DF1:=0 IDESTINATION FIELD
316 1 SF1:=1 ISOURCE FIELD

317 1 IMMED01:=7 IDIRECT ADDRESS, LOW BIT=0
318 1 R001:=2 IFOR JOINT USE W/ RIF FIELD
319 1 R021:=2 I
320 1 R041:=2 I
321 1 R061:=2 I
322 1 R101:=2 I
323 1 R121:=2 I
324 1 R141:=2 I

```

325      R16:=2      |
326      IMMEDI:=3      |DIRECT ADDRESS, LOW BIT=1
327      R01:=3      |FOR JOINT USE W/ RIF FIELD
328      R03:=3      |
329      R05:=3      |
330      R07:=3      |
331      R11:=3      |
332      R13:=3      |
333      R15:=3      |
334      R17:=3      |
335      C000000:=3      |ASPHI/BSPHI CONSTANTS
336      C177777:=3      |
337      C125252:=3      |
338      C052525:=3      |
339      C000001:=3      |
340      C100000:=3      |
341      C000200:=3      |
342
343
344
345      .TOC *      <AEN>-A-BUS DATA SOURCE
346      |SPECIFIES GATING OF DATA ONTO A-BUS, ALWAYS IN EFFECT.
347      .FIELD AEN:=<39:38>
348      |XMUX:=0      |XMUX=SR OR FLTPY ASSEMBLE
349      |CMUX:=1      |SHIFT TREE
350      |ASPL0:=2      |DIRECT ASP LOCATIONS 00-17
351      |ASPHI:=3      |DIRECT ASP LOCATIONS 20-37
352
353
354
355      .TOC *      <ASEL>-A-BUS SOURCE SELECTION CONTROL
356      |SPECIFIES CONTROL OF INDIVIDUAL A-BUS SOURCES, ALWAYS IN EFFECT.
357      .FIELD ASELO:=<36>
358      |XMUX CONTROL WHEN AEN/XMUX [USES ASELO ONLY]
359      |SR:=0      |SR OUTPUT ONTO BUS-A
360      |FLTPY:=1      |FLTPY-ASSEMBLE ONTO BUS-A
361      .FIELD ASELI:=<37:36>
362      |CMUX CONTROL WHEN AEN/CMUX. SHIFTS CMUX INPUT APPROPRIATE AMOUNT
363      |LEFT-1:=0      |LOW BIT GETS SENDMUX OUTPUT
364      |DIRECT:=1      |OUTPUT=INPUT
365      |RIGHT-1:=2      |HIGH BIT GETS DIC
366      |RIGHT-2:=3      |HIGH BITS BOTH GET DIC
367      |USED IN CONJUNCTION WITH <RIF> FOR SP ADDRESS WHEN AEN/ASPL0 OR AEN/ASPHI
368      |IMMED0:=0      |DIRECT ADDRESS, LOW BIT=0
369      |R00:=0      |FOR JOINT USE W/ RIF FIELD
370      |R02:=0      |
371      |R04:=0      |
372      |R06:=0      |
373      |R10:=0      |
374      |R12:=0      |
375      |R14:=0      |
376      |R16:=0      |
377      |IMMED1:=1      |DIRECT ADDRESS, LOW BIT=1
378      |R01:=1      |FOR JOINT USE W/ RIF FIELD

```

```

379      R03:=1      |
380      R05:=1      |
381      R07:=1      |
382      R11:=1      |
383      R13:=1      |
384      R15:=1      |
385      R17:=1      |
386      C000000:=1      |ASPHI/BSPHI CONSTANTS
387      C177777:=1      |
388      C125252:=1      |
389      C052525:=1      |
390      C000001:=1      |
391      C100000:=1      |
392      C000200:=1      |
393      DF:=2      |DESTINATION FIELD
394      SF:=3      |SOURCE FIELD
395
396
397
398      .TOC *      <RIF>-ASP, BSP REGISTER IMMEDIATE FIELD
399      |SPECIFIES ADDRESSES WITH ASP, BSP ALONG WITH AEN, ASELO & BEN, BSEL
400      .FIELD RIF:=<35:33>
401      |R00-OR-01:=4      |LOW BIT IS 0/1, SPECIFIED BY
402      |R00:=4      |
403      |R01:=4      |
404      |R02-OR-03:=5      |USING EITHER IMMED0/IMMED1 MODES
405      |R02:=5      |
406      |R03:=5      |
407      |R04-OR-05:=6      |
408      |R04:=6      |
409      |R05:=6      |
410      |R06-OR-07:=7      |
411      |R06:=7      |
412      |R07:=7      |
413      |R10-OR-11:=0      |
414      |R10:=0      |
415      |R11:=0      |
416      |R12-OR-13:=1      |ADDR<3:0>H = -RIF<2>H & RIF<1:0>H & A/BSEL<0>H
417      |R12:=1      |
418      |R13:=1      |
419      |R14-OR-15:=2      |
420      |R14:=2      |
421      |R15:=2      |
422      |R16-OR-17:=3      |
423      |R16:=3      |
424      |R17:=3      |
425
426      C000000:=4      |ASPHI/BSPHI CONSTANTS
427      C177777:=5      |
428      C125252:=6      |
429
430      C052525:=7      |
431      C000001:=0      |
432      C100000:=1      |
433      C000200:=2      |

```

```

433
434
435 .TOC * <COUT>-CARRY OUT BIT MUX SELECTION
436 |SPECIFY INPUT TO D[C] REGISTER, LOADED WHEN D REGISTER LOADED, ALWAYS IN EFFECT.
437 .FIELD COUT1:1=<32:30>
438     CIN1:1=0          |OUTPUT OF CINMUX [SIC]
439     PS[C]1:1=1        |PS C-BIT
440     ALU001:1=2        |ALU OUTPUT BIT 00
441     ALU071:1=3        |ALU OUTPUT BIT 07
442     ALU151:1=4        |ALU OUTPUT BIT 15
443     COUT071:1=5       |BYTE CARRY BIT
444     COUT151:1=6       |WORD CARRY BIT
445     D[C]1:1=7         |PROPAGATE [SAVE] LAST D[C]
446
447 |-----
448
449
450
451 .TOC * CLOCKS
452
453
454
455 .TOC * <WHEN>-D/SR WHEN TO CLOCK
456 |SPECIFY CLOCK D/SR REGISTERS AT P2 T OR P3 T, ALWAYS IN EFFECT.
457 .FIELD WHEN1:1=<29>,0
458     AT-P2-T1:1=0      |CLOCK D AND/OR SR AT P2 T[100 NS].
459     AT-P3-T1:1=1      |CLOCK D AND/OR SR AT P3 T[150 NS].
460
461
462
463 .TOC * <CLKD>-ENABLE D-REGISTER CLOCKING
464 |ENABLES CLOCKING OF D-REGISTER, ALWAYS IN EFFECT.
465 .FIELD CLKD1:1=<28>,0
466     NO1:1=0           |NOP
467     YFS1:1=1          |CLOCK D[C], D-REGISTER AT <WHEN>
468
469
470
471 .TOC * <CLKSR>-ENABLE SR-REGISTER CLOCKING
472 |ENABLES CLOCKING OF SR-REGISTER, ALWAYS IN EFFECT.
473 .FIELD CLKSR1:1=<27>,0
474     NO1:1=0           |NOP
475     YFS1:1=1          |CLOCK SR-REGISTER AT <WHEN>
476
477
478
479 .TOC * <CLKBA>-ENABLE CLOCKING OF BA-REGISTER
480 |ENABLES CLOCKING OF BA-REGISTER AT P1 T[60 NS], ALWAYS IN EFFECT.
481 .FIELD CLKBA1:1=<26>,0
482     NO1:1=0           |NOP
483     YFS1:1=1          |CLOCK BA-REGISTER AT P1 T[60 NS].
484
485
486

```

```

487 .TOC * <SCC>-ENABLE SETTING OF PS CONDITION CODES
488 |ENABLE CLOCKING OF PS CONDITION CODES AT P2 T[100 NS] OF NEXT UWORD, D MUST
489 |BE CLOCKED AT P2 T OR EARLIER OF PREVIOUS MICROWORD, ALWAYS IN EFFECT.
490 .FIELD SCC1:1=<25>,0
491     NO1:1=0           |NOP
492     YFS1:1=1          |ENABLE CLOCKING IN NEXT UWORD
493
494 |-----
495
496
497
498 .TOC * BUS/UCON & CSP-ADDRESS & SHIFT-TREE CONTROL
499
500
501
502 .TOC * BUS/UCON CONTROL
503
504
505
506 .TOC * <BEGIN>-BEGIN BUS/UCON OPERATION
507 |INITIATE BUS XOR UCON OPERATION, ALWAYS IN EFFECT.
508 .FIELD BEGIN1:1=<24>,0
509     NO1:1=0           |NOP FOR BUS AND UCON OPERATIONS
510     YFS1:1=1          |BEGIN OPERATION SPECIFIED
511
512
513
514 .TOC * <SFLECT>-SELECT BUS OR UCON
515 |SELECT BUS XOR UCON, ONLY USED IF BEGIN/YES.
516 .FIELD SELECT1:1=<23>
517     BUS1:1=0          |SELECT BUS
518     UCON1:1=1         |SELECT UCON
519
520
521
522 .TOC * BUS CONTROL
523
524
525 .TOC * <BUSCODE>-BUS CODE ACTION FIELD
526 |BUS ACTION CODES, ONLY USED IF BEGIN/YES & SELECT/BUS.
527 .FIELD BUSCODE1:1=<22:20>
528     DAT1-CLKIR1:1=0    |DATA IN, LOAD IN
529     DAT1-NOINT1:1=1    |DATA IN, NO INTERNAL ADDRESS
530     DAT01:1=2          |DATA OUT
531     DATIB1:1=3         |DATA IN, ALLOW; ODD ADDRESS
532     DATIB[P]1:1=1      |DATA IN, ALLOW; ODD ADDRESS, FORCE TO PAUSE
533     DATIP1:1=4         |DATA IN, NO CACHE, LOCK BUS
534
535     DATOR1:1=5         |DATA OUT, ALLOW; ODD ADDRESS
536     DAT1:1=6           |DATA IN
537     DATIP[P]1:1=6      |DATA IN, ALLOW; FORCE TO PAUSE
538     INVALIDDAT1:1=7    |INVALIDATE CACHE LOCATION FUNCTION
539
540
541
542 .TOC * UCON CONTROL

```

```

541
542
543
544 .TOC * <FLPGO>=START HOT FLOATING POINT
545 |INITIATES HOT FLOATING POINT FUNCTION, ONLY USED IF BEGIN/YES & SELECT/UCON.
546 .FIELD FLPGO::=<22>
547 NO::=0 |NOP
548 YES::=1 |YELL GO
549
550
551
552 .TOC * <UCON-XFER>=UCON OPERATION
553 |EXECUTE A UCON FUNCTION, ONLY USED IF BEGIN/YES & SELECT/UCON.
554 .FIELD UCON-XFER::=<21>
555 NO::=0 |NOP
556 YES::=1 |START UCON OPERATION
557
558
559
560 .TOC * <UCON-LOAD>=LOAD UCON REGISTER
561 |LOAD UCON CONTROL REGISTER, ONLY USED IF BEGIN/YES & SELECT/UCON.
562 .FIELD UCON-LOAD::=<20>
563 NO::=0 |NOP
564 YES::=1 |LOAD UCON CONTROL REGISTER
565
566
567
568 .TOC * CSP ADDRESS SPECIFICATION
569
570
571 .TOC * <CSPADDR>=CSP IMMEDIATE ADDRESS
572 |SPECIFY CSP 4 BIT ADDRESS, ONLY USED IF BEN/CSP.
573 .FIELD CSPADDR::=<23:20>
574 D00::=17 |NOTE INVERSION
575 D01::=16 |
576 D02::=15 |
577 D03::=14 |
578 D04::=13 |
579 D05::=12 |
580 D06::=11 |
581 D07::=10 |
582 D10::=07 |
583 D11::=06 |
584 D12::=05 |
585 D13::=04 |
586 D14::=03 |
587 D15::=02 |
588 D16::=01 |
589 D17::=00 |
590 C052525::=07 |CSP/DCS CONSTANTS
591 C125252::=06 |
592 C177777::=05 |
593 C000000::=04 |
594

```

```

595
596
597 .TOC * SHIFT CONTROL
598
599
600 .TOC * <BMUX>=SECOND LEVEL OF SHIFT TREE
601 |BMUX CONTROLS SHIFT RIGHT OF 0 OR 4, ALWAYS IN EFFECT.
602 .FIELD BMUX::=<23>
603 DIRECT::=0 |BMUX<15:00>
604 RTGHT-4::=1 |4*D[C] # AMUX <15:04>
605
606
607
608 .TOC * <AMUX>=FIRST LEVEL OF SHIFT TREE
609 |AMUX CONTROLS INPUT OF D-REG/COUNTER TO TREE, ALWAYS IN EFFECT.
610 .FIELD AMUX::=<22:20>
611 DIRECT::=0 |D<HI> # D<LO>
612 D[LO]#D[LO]::=1 |D<LO> # D<LO>
613 SIGNEXT::=2 |D[D[C]] # D<LO>
614 COUNTER#D[LO]::=3 |COUNTER # D<LO>
615 COUNTER::=3 |SAME
616 D[HI]#D[HI]::=4 |D<HI> # D<HI>
617 SWAB::=5 |D<LO> # D<HI>
618 RIGHT-8::=6 |8*D[C] # D<HI>
619 COUNTER#D[HI]::=7 |COUNTER # D<HI>
620
621 |-----
622
623
624
625 .TOC * SP REWRITE & REGISTER CLOCKS
626
627
628
629 .TOC * <WRCSF>=WRITE TO CSP
630 |WRITE CSP FROM BMUX (BUSIN/CACHE), ALWAYS IN EFFECT.
631 .FIELD WRCSF::=<19>,0
632 NO::=0 |NOP
633 YES::=1 |ON P3, 120-150 NS.
634
635
636
637 .TOC * <MOD>=MODE CONTROL OF FOLLOWING BITS
638 |CONTROLS REDEFINITION OF SP REWRITE/REGISTER CLOCK BITS, ALWAYS IN EFFECT.
639 .FIELD MOD::=<14>,0
640 CLASP::=0 |CONTROL ASP/RSP CLOCKING
641 LOADRF::=1 |CONTROL RES-REG/COUNTER LOADING
642
643
644
645 .TOC * SP REWRITE [A,B] CONTROL
646 |WHEN MOD/CLKSP
647
648

```

```

649 .TDC * <HILO>=SP HI/LO SELECT
650 |WHICH HALF OF SP'S TO REWRITE, ONLY IF MOD/CLKSP.
651 .FIELD HILO:=<18>
652     LO:=0 |REWRITE ENABLE A/B SP LO [00-17]
653     LI:=0 |
654     HI:=1 |REWRITE ENABLE A/B SP HI [20-37]
655     HI:=1 |
656
657
658
659 .TDC * <WRSEL>=REWRITE ADDRESS SELECT
660 |WHICH WRITE ADDRESS TO USE ON REWRITE, ONLY IF MOD/CLKSP.
661 .FIELD WRSEL:=<17>
662     A-ADDR:=0 |USE A ADDRESS ON REWRITE
663     A:=0 |
664     B-ADDR:=1 |USE B ADDRESS ON REWRITE
665     B:=1 |
666
667
668
669 .TDC * <WRSP>=REWRITE A/B SELECT
670 |ENABLE REWRITE OF SPECIFIC SP'S, ONLY IF MOD/CLKSP.
671 .FIELD WRSP:=<16:15>
672     NOP:=0 |NO ASP/BSP REWRITE
673     WR-A:=1 |WRITE ASP ONLY, ON P3 120-150 NS.
674     A:=1 |
675     ASP:=1 |
676     WR-B:=2 |WRITE BSP ONLY, ON P3 120-150 NS.
677     B:=2 |
678     BSP:=2 |
679     WR-A-AND-B:=3 |WRITE BOTH ON P3
680     AB:=3 |
681     PA:=3 |
682     ABSP:=3 |
683     BASP:=3 |
684     BOTH:=3 |
685
686
687
688 .TDC * REGISTER LOADING
689 |WHEN MOD/LOADREG
690
691
692 .TDC * <LOADRES>=LOAD RESIDUAL CONTROL REGISTER
693 |ENABLE LOAD OF RESIDUAL CONTROL REGISTER FROM B-BUS, ONLY IF MOD/LOADREG.
694 .FIELD LOADRES:=<18>
695     NO:=0 |NOP
696     YES:=1 |LOAD RES WITH B-BUS<14:11>
697     |AT P2 T[100 NS], B-BUS<14> COMPLEMENTED
698
699
700
701 .TDC * <LOADCOUNT>=LOAD COUNTER
702 |ENABLE LOAD OF COUNTER FROM B-BUS <7:0>, ONLY IF MOD/LOADREG.

```

```

703 .FIELD LOADCOUNT:=<16>
704     NO:=0 |NOP
705     YES:=1 |LOAD COUNTER AT P2 T[100 NS].
706
707 |-----
708
709
710
711 .TDC * SEQUENCING FIELD
712
713
714
715 .TDC * <URF>=BUT MICROBRANCH FIELD
716 |SPECIFIES CONDITIONS TO MODIFY <UPF>/<J> FIELD DURING BRANCH, ALWAYS IN EFFECT.
717 .FIELD URF:=<13:0>,30
718
719 .TDC * NO BUT
720     NULL:=30 |SPECIFY NO MODIFICATION - DEFAULT
721
722
723 .TDC * ACTIVE ONLY
724 |PURELY ACTIVE BUTS GENERATE SIDE EFFECTS; THEY DO NOT MODIFY THE <UPF> FIELD
725 |BY THE "OR-ING"-IN-OF-CONDITIONS METHOD. THEY MAY MODIFY EXPLICITLY THE ENTIRE <UPF> FIELD,
726 |AS IN BUT(RETURN)
727     P-OP:=22 |FORM R[SF]-IOR="001
728     CUA-TRACK:=31 |RESUME/RESTART CUA TRACKING
729     CLR-FLAG-PFS-UCON:=32 |CLEAR FLAG<2:0>, EX-FLAG<1>, RES-REGISTER, UCON-REGISTER
730     DIAGNOSF:=33 |SPECIAL DIAGNOSTIC BUT
731     SUBRB:=34 |RETURN <- EMIT<14:03>, PAGE <- EMIT<02:00>
732     SUBR-R:=34 |SYNONYMS ARE:
733     GOTO:=34 |
734     GO-TO:=34 |
735     SUBRA:=35 |RETURN <- D<14:03>, PAGE <- EMIT<02:00>
736     SUBR-A:=35 |SYNONYM
737     B36:=36 |TBD
738     RETURN:=37 |PAGE <- RETURN<11:09>, NUA <- RETURN<08:00>
739
740 .TDC * INACTIVE ONLY
741 |INACTIVE BUTS ONLY CAUSE MODIFICATION OF THE <UPF> FIELD BY THE "OR-ING"-
742 |IN-OF-CONDITIONS METHOD.
743
744     SR3-01:=00 |----UPF MASK-----
745     CASF:=00 |876 543 210 OCTAL **NOT AFFECTED
746     SR03:=00 |*** **0 000 (000)
747     SR02:=00 |
748     SR01:=00 |*** **0 111 (007)
749     |*** **1 011 (013)
750     |*** **1 101 (015)
751
752     SP00:=00 |
753     IR15-12:=01 |*** **1 110 (016)
754     DNP:=01 |*** **0 000 (000)
755     INSTR5:=02 |
756     INSTR-5:=02 |*** **0 000 (000)
757     IR11-FLTTP3-01:=03 |
758     IR11-A1:=03 |*** **0 000 (000)
759     IR9-61:=04 |*** **1 111 (017)
760     |*** **0 000 (000)

```

```

757      SOP1:=04      1
758      MOV-DR7:IR5-3:=05      1*** **0 000 (000)
759      MOV-DR7:=05      1*** **0 111 (007)
760      IR5-3:=05      1*** **1 000 (010)
761      BGSRV-FPSERV:D(C):FPRET:=07      1*** **0 000 (000)
762      BGSRV-FPSERV:=07      1*** **0 111 (007)
763      D(C)-C:=07      1*** **1 011 (013)
764      FPRET1-0:=07      1*** **1 100 (014)
765      COUT07:DOUT07:FPS05:=10      1*** **0 000 (000)
766      COUT07:=10      1*** **0 011 (003)
767      DOUT07:=10      1*** **0 101 (005)
768      COUT07:DOUT07:=10      1*** **0 001 (001)
769      FPS05:=10      1*** **0 110 (006)
770      DM0:SM0:RYTE:=11      1*** **0 000 (000)
771      DM0:=11      1*** **0 011 (003)
772      SM0:=11      1*** **0 101 (005)
773      BYTE:=11      1*** **0 110 (006)
774      GD3-2:=12      1*** **0 **0 (000)
775      RG-SERVICE-L:MFSS:MULTIPLE:=14      1*** **0 000 (000)
776      BG-SERVICE-L:=14      1*** **0 011 (003)
777      MFSS:=14      1*** **0 101 (005)
778      MULTIPLE:=14      1*** **0 110 (006)
779      MASKED-PS(T):=14      1
780      D00:=14      1
781      PS(N):=14      1
782      FLAG7:=14      1
783      EXFLAG1:=14      1
784      FLTPTS:=14      1
785      EXFLAG2:=14      1
786      INIT-JAM:=14      1
787      D14-00EQ04D15:=15      1*** **0 **0 (000)
788      D14-00-EQ-08D15:=15      1
789      D14-00-EQ-0:=15      1*** **0 **01 (001)
790      D15:=15      1*** **0 **10 (002)
791      IR11:PS15:=16      1*** **0 **00 (000)
792      IR11-R:=16      1*** **0 **01 (001)
793      PS15:=16      1*** **0 **10 (002)
794      VECTOR-LOAD:DR6-7L:=21      1*** **0 **00 (000)
795      VECTOR-LOAD:=21      1*** **0 **01 (001)
796      DR6-7L:=21      1*** **0 **10 (002)
797      D(C):BA00:=23      1*** **0 **00 (000)
798      D(C)-R:=23      1*** **0 **01 (001)
799      BA00:=23      1*** **0 **10 (002)
800      OTHER-JAM:FP-PROC:=24      1*** **0 **00 (000)
801      OTHER-JAM:=24      1*** **0 **01 (001)
802      FP-PROC:=24      1*** **0 **10 (002)
803      INTP-HIGH:INSTR-BRANCH-L:=26      1*** **0 **00 (000)
804      INTR-HIGH:=26      1*** **0 **01 (001)
805      INSTR-BRANCH-L:=26      1*** **0 **10 (002)
806      PREFETCH-JAM:FP-FD:=27      1*** **0 **00 (000)
807      PREFETCH-JAM:=27      1*** **0 **01 (001)
808      FP-FD:=27      1*** **0 **10 (002)
809
810      .TOC *      BOTH ACTIVE AND INACTIVE

```

```

811      THESE BUTS HAVE BOTH ACTIVE AND INACTIVE EFFECTS
812      1----UPF MASK-----
813      1876 543 210 OCTAL **NOT AFFECTED
814      INSTR1:=06      1*** **0 000 (000) BUS CONTROL, SP REWRITE DEFEAT
815      INSTR-1:=06      1
816      SR1-0:COUNT-IS-377:=13      1*** **0 000 (000) BUMP COUNTER
817      SP1-0:=13      1*** **0 001 (001) STILL BUMP COUNTER
818      COUNT-IS-377-A:=13      1*** **0 110 (006) BUMP COUNTER
819      COUNT-IS-377:D(C):=17      1*** **0 **00 (000) BUMP COUNTER
820      COUNT-IS-377-R:=17      1*** **0 **01 (001) BUMP COUNTER
821      D(C)-A:=17      1*** **0 **10 (002) STILL BUMP COUNTER
822      COUNT-IS-377:=25      1*** **0 **00 (000) BUMP COUNTER
823      PREFETCH-L:SERVICE:=20      1*** **0 **00 (000) TIMING
824      PREFETCH-L:=20      1*** **0 **01 (001) TIMING
825      SERVICE:=20      1*** **0 **10 (002) TIMING
826      LAST:=20      1*** **0 **11 (003) TIMING
827
828
829
830
831      .TOC *      <UPF>-MICRO POINTER FIELD
832      1SPECIFIES EITHER NEXT MICROINSTRUCTION ADDRESS OR BASE TARGET
833      1ADDRESS TO BE USED "UNDER" THE BUT-CODE IN <UBF>.
834      1FIELD      UPF:=<R10>,000      1ACTUAL MICROWORD POINTER FIELD
835      1ADDRESS      J1:=<R10>      1THIS FIELD ALSO HAS MICROADDRESS QUALITIES
836
837      1BASE MACHINE MICROCODE ENTRY POINTS:
838
839      1THESE ENTRY POINTS HAVE BEEN FIXED AS OF 31-AUGUST-1976.
840      INIT01 := 3412      1INITIALIZATION SUBROUTINE
841      CON99 := 1040      1FORCE "CONSOLE-MODE HALT"
842      FET01 := 0702      1INSTR FETCH, NO OVERLAP
843      FET03 := 0700      1INSTR FETCH, OVERLAP
844      SER01 := 0701      1SERVICE ENTRY, OVERLAP
845      SER02 := 0703      1SERVICE ENTRY, NO OVERLAP
846
847      1ENTRY POINTS INTO BASE MACHINE FOR "BUTA(DIAGNOSE)":
848      1THESE ENTRY POINTS FIXED AS OF 26-OCT-76:
849      MED23 := 3200      1FOR FLPADR: D_ASPLO(D)-TOP
850      MED25 := 3020      1FOR FLPADR: D_ASPHI(D)-TOP
851      MED27 := 3210      1FOR FLPADR: ASPLO(D)-TOP_D
852      MED29 := 3214      1FOR FLPADR: ASPHI(D)-TOP_D
853      MED31 := 3044      1FOR FLPADR: D_BSPLO(D)-TOP
854      MED33 := 3230      1FOR FLPADR: D_BSPHI(D)-TOP
855      MED35 := 3234      1FOR FLPADR: BSPLO(D)-TOP_D
856      MED37 := 3064      1FOR FLPADR: BSPHI(D)-TOP_D
857
858      RYTE01 := 0032      1FOR KJENAB: DATOB*KJENAB
859
860      RTS02 := 4034      1FOR SPRYKT: DATI, BA_SP-A, SP_SP+2
861
862      DST01 := 0511      1FOR ALTER: DATIS[P]
863      DST02 := 0512      1
864      DAT20 := 0527      1

```

```

865 DAT22 := 0525 1
866
867 LOADNZW4 := 4330 1FOR FPSCC&CLKFPSCC: D_CSP(MD)
868 LOADNZWS := 4332 1FPS<310>_D<310>
869
870 -----
871
872
873
874 .TOC * MISCELLANEOUS FIELDS
875
876
877
878 .TOC * <NEXT-PAGE>-NEW PAGE ADDRESS LOADED DURING BUT(SUBROUTINE)
879 [THESE 3 BITS ARE CLOCKED INTO PAGE REGISTER DURING A BUT(SUBRA) OR
880 [BUT(SUBRA)], ONLY USED WHEN URF/BUT(SUBRA) OR URF/BUT(SUBRA),
881 .FIELD NEXT-PAGE:=<32:30>
882
883
884
885 .TOC * <MULTIPLE>-SELECT CODE FOR BUT(MULTIPLE)
886 [MUST BE SET IN BOTH PREVIOUS AND CURRENT MICROWORDS WHEN BUT(MULTIPLE) IS TO BE EMPLOYED,
887 .FIELD MULTIPLE:=<32:30>
888 MASKED-PS(T):=0 1
889 D00:=1 1
890 PS(N):=2 1
891 FLAG7:=3 1
892 EXFLAG1:=4 1
893 FLTP7:=5 1
894 EXFLAG2:=6 1
895 INIT-JAM:=7 1
896
897
898 .TOC * FMIT FIELD - IMMEDIATE DATA FROM MICROWORD
899 [USED WHENEVER LOADING IMMEDIATE DATA FROM MICROWORD
900 .FIELD EMIT1:=<47:44>*<41:30>
901 .FIELD EMIT1H:=<47:44>
902 .FIELD EMIT1L:=<41:30>
903 .FIELD EMIT1H:=<47:44>
904 .FIELD EMIT1L:=<41:30>
905 .FIELD EMIT1H:=<47:44>
906 .FIELD EMIT1L:=<41:30>
907 .FIELD EMIT1H:=<47:44>
908 .FIELD EMIT1L:=<41:30>
909 .FIELD EMIT1H:=<47:44>
910 .FIELD EMIT1L:=<41:30>
911 .FIELD EMIT1H:=<47:44>
912 .FIELD EMIT1L:=<41:30>
913 .FIELD EMIT1H:=<47:44>
914 .FIELD EMIT1L:=<41:30>
915 .FIELD EMIT1H:=<47:44>
916 .FIELD EMIT1L:=<41:30>
917 .FIELD EMIT1H:=<47:44>
918 .FIELD EMIT1L:=<41:30>

```

```

919 .FIELD EMIT02:=<32>
920 .FIELD EMIT01:=<31>
921 .FIELD EMIT00:=<30>
922
923
924
925 .TOC * RETURN ADDRESS - FOR MICROSUBROUTINE CALLS
926 [USED WITH BUT(SUBRA) AND BUT(SUBRA)
927 .FIELD RETURN:=<46:44>*<41:33> 1PAGE 8 D.I.P.
928
929
930
931 .TOC * UCON SELECTION AND CONTROL FIELDS
932
933
934 .TOC * SELECTION
935 [SELECT PARTICULAR UCON, ONLY USED IF BEGIN/YES & SELECT/UCON.
936 .FIELD UCON-SEL-EMI:=<43> 1SELECT EMI CAN ONLY BE DONE BY USING
937 1 BUTA(CLR-FLAG-RES-UCON) TO ASSERT UCON-SEL-EMI-L
938 1
939 .FIELD UCON-SEL-I-O:=<46> 1SELECT I-O (BUS) CONTROL
940 NO:=0
941 YES:=1
942 .FIELD UCON-SEL-WCS:=<45> 1SELECT WCS/ECS/DCS
943 NO:=0
944 YES:=1
945 .FIELD UCON-SEL-CACHEKT:=<44> 1SELECT CACHE/KT
946 NO:=0
947 YES:=1
948 .FIELD UCON-SEL-PROC:=<36> 1SELECT PROCESSOR CONTROL
949 NO:=0
950 YES:=1
951 .FIELD UCON-SEL-FLTP7:=<33> 1SELECT NOT FLOATING POINT
952 NO:=0
953 YES:=1
954
955
956
957 .TOC * CONTROL
958 [AFTER UCON(S) SELECTED FROM ABOVE, CONTROL COMES FROM HERE.
959 .FIELD UCON:=<32:30>*<35:34>*<47>*<42:38>
960 .FIELD UCONH:=<32:30>
961 .FIELD UCONH:=<35:34>
962 .FIELD UCONL:=<47>*<42:38>
963 .FIELD UCONH:=<32>
964 .FIELD UCONL:=<31>
965 .FIELD UCONH:=<30>
966 .FIELD UCONH:=<35>
967 .FIELD UCONH:=<34>
968 .FIELD UCONH:=<47>
969 .FIELD UCONH:=<42>
970 .FIELD UCONH:=<41>
971 .FIELD UCONH:=<40>
972 .FIELD UCONH:=<39>

```

```

973 .FIELD UCON0511=<38>
974
975 |-----|
976
977
978
979 .TOC * BASE MACHINE EXTENSION BITS
980
981 | LAYOUT IN BASE MACHINE (NOT DCS) ADDRESS SPACE:
982
983 |
984 | --- NAME --- 0 1 2 3 4
985 | ROMEX 00 H X
986 | ROMEX 01 H X X
987 | ROMEX 03 H X
988
989 | FPSEL L X X X
990 | SFCC L X X X
991 | FLPADR L X X X
992
993 | SPBYKT L X X
994 | UDAD01 L X X
995 | UDAD00 I X X
996 | UKJCONT L X X
997 | UKT01 H X X
998 | UKT00 H X X
999 | UALTER L X X
1000 | UKTEN L X X
1001
1002
1003 | NULL BIT DEFINITIONS:
1004
1005 .FIELD ROMEX0011=<60> | ACTIVE HIGH
1006 | ZERO11=0 |
1007 | ONE11=1 |
1008 .FIELD ROMEX0111=<61> | ACTIVE HIGH
1009 | ZERO11=0 |
1010 | ONE11=1 |
1011 .FIELD ROMEX0311=<62> | ACTIVE HIGH
1012 | ZERO11=0 |
1013 | ONE11=1 |
1014 .FIELD FPSEL11=<57> | ACTIVE LOW
1015 | ZERO11=1 |
1016 | ONE11=0 |
1017 .FIELD SFCC11=<58> | ACTIVE LOW
1018 | ZERO11=1 |
1019 | ONE11=0 |
1020 .FIELD FLPADR11=<59> | ACTIVE LOW
1021 | ZERO11=1 |
1022 | ONE11=0 |
1023 .FIELD SPBYKT11=<55> | ACTIVE LOW
1024 | ZERO11=1 |
1025 | ONE11=0 |
1026 .FIELD UDAD0111=<53> | ACTIVE LOW

```

```

1027 | ZERO11=1 |
1028 | ONE11=0 |
1029 .FIELD UDAD0011=<52> | ACTIVE LOW
1030 | ZERO11=1 |
1031 | ONE11=0 |
1032 .FIELD UKJCONT11=<54> | ACTIVE LOW
1033 | ZERO11=1 |
1034 | ONE11=0 |
1035 .FIELD UKT0111=<49> | ACTIVE HIGH
1036 | ZERO11=0 |
1037 | ONE11=1 |
1038 .FIELD UKT0011=<48> | ACTIVE HIGH
1039 | ZERO11=0 |
1040 | ONE11=1 |
1041 .FIELD UALTER11=<51> | ACTIVE LOW
1042 | ZERO11=1 |
1043 | ONE11=0 |
1044 .FIELD UKTEN11=<50> | ACTIVE LOW
1045 | ZERO11=1 |
1046 | ONE11=0 |
1047
1048 |-----|
1049
1050
1051
1052 .TOC * SPECIAL DCS FIELDS
1053
1054
1055
1056 .TOC * FIELDS USED IN PAGES 4, 5, OR 6 OF DCS
1057
1058
1059 .TOC * <LOAD-DCS-CTR>=LOAD DIAGNOSTIC COUNTER FROM EMITH
1060 | THIS CODE LOADS THE 4-BIT DCS COUNTER FROM THE CURRENT
1061 | MICROWORD'S EMITH FIELD. THIS COUNTER IS CLOCKED AT EVERY PO
1062 | FOLLOWING, UNTIL THE COUNTER REACHES ZERO. AT THIS POINT, THE
1063 | COMPARE IS ENABLED, CLOCKING THE RESULT OF THE CURRENT EMITHUA
1064 | COMPARE INTO THE ERROR LATCH.
1065 | ONLY USED IN PAGES 4, 5, OR 6 OF DCS.
1066 .FIELD LOAD-DCS-CTR11=<51>,0
1067 | NOP11=0 | NOP
1068 | YES11=1 | LOAD, COUNT, AND ENABLE COMPARE
1069
1070
1071
1072 .TOC * <CTR>=4 BIT DCS COUNTER VALUE FROM EMITH
1073
1074 | THIS FOUR BIT VALUE IS LOADED INTO THE COUNTER (DIAGNOSTIC1),
1075 | WHEN LOAD COUNTDOWN/YES. COMPLEMENT OF ACTUAL VALUE IS USED, FOR COUNT DOWN.
1076 | LOADING COUNTER VALUE OF 17(8) CAUSES COMPARE AT END OF THIS UNORD.
1077 .FIELD CTR11=<47:44>
1078 | C011=17
1079 | C111=16
1080 | C211=15
1081 | C311=14

```

```

1081      C4.11=13
1082      C5.11=12
1083      C6.11=11
1084      C7.11=10
1085      C8.11=07
1086      C9.11=06
1087      C10.11=05
1088      C11.11=04
1089      C12.11=03
1090      C13.11=02
1091      C14.11=01
1092      C15.11=00
1093
1094
1095
1096      .TDC *      <LOAD-ENUA-ERRCOD>-LOAD THE ENUA AND ERRCOD REGISTERS
1097      !THIS CODE LOADS THE 12-BIT ENUA REGISTER FROM THE <EMITH,EMITL> FIELD
1098      !OF THE CURRENT MICROWORD, AND LATCHES THE NUA INTO THE ERRCOD REGISTER.
1099      !ONLY USED IN PAGES 4, 5, OR 6 OF DCS.
1100      .FIELD LOAD-ENUA-ERRCOD:1=<54>,0
1101      NOP:1=0      !NOP
1102      YES:1=1      !LOAD REGISTERS AT P0
1103
1104
1105
1106      .TDC *      <ENUA>-ENUA VALUE FROM EMIT
1107      !THIS 12 BIT FIELD IS LOADED FROM <EMIT> TO THE ENUA REGISTER
1108      !WHEN LOAD-ENUA-ERRCOD=YES.
1109      .FIELD ENUA:1=<41:10>
1110
1111
1112
1113      .TDC *      <VERIFY>-VERIFY BIT FOR SELF CHECK TEST
1114      !WHEN IN SELF TEST MODE OF DCS, SETTING THIS BIT CAUSES THE VERIFY COUNTER TO BE
1115      !BUMPED AT THE START OF THIS MICROWORD. THE VERIFY BIT IS IMPLICITLY SET FOR
1116      !ANY REFERENCE TO PAGE 7 (IE, THE COUNTER IS AUTOMATICALLY BUMPED ON A REFERENCE
1117      !TO PAGE 7).
1118      !ONLY [EXPLICITLY] USED IN PAGES 4, 5, OR 6 OF DCS, WHEN IN SELF TEST MODE.
1119      .FIELD VERIFY:1=<48>,0
1120      NOP:1=0      !NO ACTION
1121      RUMP:1=1      !BUMP VERIFY COUNTER AT P0, WHEN IN SELF TEST MODE
1122
1123
1124
1125      .TDC *      FIELDS USED IN PAGE 7 OF DCS EXTENSION
1126
1127
1128      .TDC *      <EOP>-SIGNAL SUCCESSFUL END OF PASS
1129      !THIS CODE SETS THE END OF PASS LATCH, LIGHTING THE EOP LED
1130      !ONLY USED IN PAGE 7 OF DCS.
1131      .FIELD EOP:1=<49>,0
1132      NOP:1=0      !NO EOP
1133      SIGNAL:1=1    !SIGNAL SUCCESSFUL EOP AT P0
1134

```

```

1135
1136
1137      .TDC *      <DAD>-DCS CONTROL OF BASE MACHINE EXTENSION DAD BITS
1138      !THESE BITS ARE WIRE-AND'ED INTO THE BASE MACHINE DAD<110> BITS.
1139      !MUST BE SPECIFIED IN UWORD BEFORE CSP REFERENCE.
1140      !ONLY USED IN PAGE 7 OF DCS.
1141      .FIELD DAD:1=<53:52>,0
1142      NO-DAD:1=0      !
1143      FIRST-1-OR-2:1=1 !SETUP BYTE=CONST ROM INPUT
1144      SECOND-1-OR-2:1=2 !
1145      WRITE-BYTE:1=3   !SETUP FOR BYTE WRITE TO ASP/BSP
1146
1147
1148
1149      .TDC *      FIELDS USED IN ALL PAGES OF DCS EXTENSION
1150
1151
1152      .TDC *      <SCOPE>-SCOPE ON ERROR, DIAGNOSTIC BUT
1153      !THIS CODE IS A SPECIAL BUT, THAT, WHEN ENABLED, CHECKS THE ERROR
1154      !LATCH TO SEE IF IT IS SET. IF IT IS, NUA00 IS FORCED TO A ZERO.
1155      !ELSE IT IS LEFT UNCHANGED. USED TO IMPLEMENT FORCED SCOPE LOOP ON ERROR.
1156      !USED IN ALL PAGES OF DCS.
1157      .FIELD SCOPE:1=<50>,0
1158      NOP:1=0      !NOP
1159      ENABLPD:1=1   !ENABLE SCOPE LOOPING FACILITY
1160
1161
1162
1163      !-----
1164      !END OF MICROWORD FIELD DEFINITIONS
1165      !-----
1166
1167
1168      !.PAGE=====
1169
1170      .TDC *      MACRO DEFINITIONS
1171
1172
1173      .TDC *      PRIMITIVE OPERATIONS
1174
1175
1176      .TDC *      TIMING
1177      .MACRO P^      11= NULL      10 NS., UP3 VIEWED AS THE START OF A MICROCYCLE
1178
1179      .MACRO P1      11= NULL      160 NS., AT P1
1180      .MACRO P1-L    11= NULL      130 NS., AT P1 LEADING EDGE
1181
1182      .MACRO P1-T    11= NULL      160 NS., AT P1 TRAILING EDGE
1183
1184      .MACRO P2      11= NULL      1100 NS., AT P2
1185      .MACRO P2-L    11= NULL      170 NS., AT P2 LEADING EDGE
1186      .MACRO P2-T    11= WHEN/AT-P2-T 1100 NS., AT P2 TRAILING EDGE
1187      .MACRO P2-II    11= NULL      !UNSUPPRESSED P2, CLOCK CONTINUOUSLY
1188
1189      .MACRO P3      11= NULL      1150 NS., 120-150 NS., AT P3

```

```

1189 .MACRO P3-L      == NULL      1120 NS., AT P3 LEADING EDGE
1190 .MACRO P3-T      == WHEN/AT-P3-T 1150 NS., AT P3 TRAILING EDGE
1191 .MACRO P3-U      == NULL      UNSUPPRESSED P3, CLOCK CONTINUOUSLY
1192
1193 .MACRO UP3        == NULL      IP3 DELAYED BY 5 NS., P0 VIEWED AS THE END OF A
1194                                MICROCYCLE. LATCHES NEW MICROINSTRUCTION INTO
1195                                THE MICROWORD BUFFER REGISTER.
1196
1197 .MACRO DEFER       == NULL      ICONTROL IS ISSUED AT THIS TIME,
1198                                I ANY REQUIRED CLOCKING OCCURS LATER
1199
1200 .MACRO NEXT        == NULL      IWHERE TO GO NEXT, CLOCKED AT UP3
1201 .MACRO SETUP       == NULL      ISETUP DATA/CONTROL
1202 .MACRO SELECT       == NULL      IMAKE A HOT-BOX SELECTION
1203 .MACRO ISSUE        == NULL      ISET/CLEAR HOT-BOX FLAG
1204 .MACRO ENABLE       == NULL      I DITTO
1205 .MACRO EMITC        == NULL      ISPECIFY AN EMIT-CONSTANT VALUE
1206
1207
1208 .PAGE=====
1209
1210 .TOC *      WRITING THE A AND B SCRATCH PADS
1211
1212 I      WRITING THE APPROPRIATE SCRATCH PADS:
1213
1214 I      (NOP      )
1215 I      (A, L, A)
1216 I      WR (B, H, B)
1217 I      (AB      )
1218 I      /1\ /1\ /1\
1219 I      I      I
1220 I      ASP, BSP, BOTH, NEITHER----- I
1221 I      LO[10-17], OR HI[20-37]----- I
1222 I      USE "A" SIDE OR "B" SIDE ADDRESS-----
1223
1224 I      WRITES CONTENTS OF D-REGISTER INTO ADDRESSED SCRATCH PADS (SEE
1225 I      BELOW) DURING P3
1226
1227 .MACRO WP(AB,HI,ADDP) == MOD/CLKSP,      ICLOCK SP MODE
1228                                WRSP/SPAB,      INOP, A, ASP, B, BSP, AB, ADSP, BA, BASP, BOTH ARE CHOICES
1229                                HILO/SHL,      INI, LO, H, L ARE CHOICES
1230                                WRSEL/SPADDP      IA, B, A-ADDR, B-ADDR ARE CHOICES
1231
1232
1233
1234 .TOC *      ASP AND BSP PHYSICAL REGISTER ADDRESSES
1235
1236 I      ENABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD ONTO
1237 I      EITHER BUS-A OR BUS-B VIA EXACT PHYSICAL ADDRESS
1238
1239 .MACRO ASPLO(XX)      == AEN/ASPLO,      ISELECT
1240                                ASEL/0XX,      IREGISTER 6
1241                                RIF/0XX      IENABLE ON BUS-A
1242

```

```

1243 .MACRO ASPHI(XX)      == AEN/ASPHI,      ISELECT
1244                                ASEL/0XX,      IREGISTER 6
1245                                RIF/0XX      IENABLE ON BUS-A
1246
1247 .MACRO ASP(XX)         == ASEL/0XX,      ISELECT REGISTER,
1248                                RIF/0XX      INO ENABLE
1249
1250
1251 .MACRO BSPLO(XX)        == BEN/BSPLO,      ISELECT
1252                                BSEL/0XX,      IREGISTER 6
1253                                RIF/0XX      IENABLE ON BUS-B
1254
1255 .MACRO BSPHI(XX)        == BEN/BSPHI,      ISELECT
1256                                BSEL/0XX,      IREGISTER 6
1257                                RIF/0XX      IENABLE ON BUS-B
1258
1259 .MACRO BSP(XX)          == BSEL/0XX,      ISELECT REGISTER,
1260                                RIF/0XX      INO ENABLE
1261
1262
1263
1264
1265
1266 .TOC *      ASP AND BSP BASE MACHINE FUNCTIONAL REGISTER ADDRESSES
1267
1268 I      ENABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD ONTO
1269 I      EITHER BUS-A "A" OR BUS-B "B" VIA FUNCTIONAL REGISTER DESIGNATION
1270
1271 .MACRO R0-A      == ASPLO(R00)
1272 .MACRO R0-B      == BSPLO(R00)
1273 .MACRO R1-A      == ASPLO(R01)
1274 .MACRO R1-B      == BSPLO(R01)
1275 .MACRO R2-A      == ASPLO(R02)
1276 .MACRO R2-B      == BSPLO(R02)
1277 .MACRO R3-A      == ASPLO(R03)
1278 .MACRO R3-B      == BSPLO(R03)
1279 .MACRO R4-A      == ASPLO(R04)
1280 .MACRO R4-B      == BSPLO(R04)
1281 .MACRO R5-A      == ASPLO(R05)
1282 .MACRO R5-B      == BSPLO(R05)
1283 .MACRO SP-A      == ASPLO(R06)
1284 .MACRO SP-B      == BSPLO(R06)
1285 .MACRO PC-A      == ASPLO(R07)
1286 .MACRO PC-B      == BSPLO(R07)
1287 .MACRO FACA[0]-B == BSPHI(R10)
1288 .MACRO FACB[0]-A == ASPHI(R10)
1289 .MACRO FACC[0]-B == BSPLO(R10)
1290 .MACRO FACC[0]-A == ASPLO(R10)
1291 .MACRO FACA[1]-B == BSPHI(R11)
1292 .MACRO FACB[1]-A == ASPHI(R11)
1293 .MACRO FACC[1]-B == BSPLO(R11)
1294 .MACRO FACC[1]-A == ASPLO(R11)
1295 .MACRO FACA[2]-B == BSPHI(R12)
1296 .MACRO FACB[2]-A == ASPHI(R12)

```

```

1297 .MACRO FACC(2)-B      ;;= BSPLO(R12)
1298 .MACRO FACC(2)-A      ;;= ASPLO(R12)
1299 .MACRO FACA(3)-B      ;;= BSPHI(R13)
1300 .MACRO FACA(3)-A      ;;= ASPHI(R13)
1301 .MACRO FACC(3)-B      ;;= BSPLO(R13)
1302 .MACRO FACC(3)-A      ;;= ASPLO(R13)
1303 .MACRO FACA(4)-B      ;;= BSPHI(R14)
1304 .MACRO FACA(4)-A      ;;= ASPHI(R14)
1305 .MACRO FACC(4)-B      ;;= BSPLO(R14)
1306 .MACRO FACC(4)-A      ;;= ASPLO(R14)
1307 .MACRO FACA(5)-B      ;;= BSPHI(R15)
1308 .MACRO FACC(5)-A      ;;= ASPHI(R15)
1309 .MACRO FACC(5)-B      ;;= BSPLO(R15)
1310 .MACRO FACC(5)-A      ;;= ASPLO(R15)
1311 .MACRO FOSTA-B        ;;= BSPHI(R17)
1312 .MACRO FOSTB-A        ;;= ASPHI(R17)
1313 .MACRO FOSTC-B        ;;= BSPLO(R17)
1314 .MACRO FOSTD-A        ;;= ASPLO(R17)
1315 .MACRO FPSHI#FEC-A    ;;= ASPHI(P16)
1316 .MACRO FEA-B          ;;= BSPHI(R16)
1317 .MACRO USER-SP-A      ;;= ASPLO(P16)
1318 .MACRO USER-SP-R      ;;= BSPLO(R16)
1319 .MACRO WHAWI-A        ;;= ASPHI(R02)
1320 .MACRO P[ZERO]-B      ;;= BSPHI(R03)
1321 .MACRO R[IR]-A        ;;= ASPHI(R17)
1322 .MACRO R[SR]-R        ;;= BSPHI(R04)
1323 .MACRO R[SR]-A        ;;= ASPHI(R04)
1324 .MACRO R[IST]-B       ;;= BSPHI(R05)
1325 .MACRO R[IST]-A       ;;= ASPHI(R05)
1326 .MACRO R[IVECT]-R     ;;= BSPHI(P02)
1327 .MACRO WCSR(0)-B      ;;= BSPHI(P00)
1328 .MACRO WCSR(1)-R     ;;= BSPHI(R01)
1329 .MACRO WCSA(0)-A      ;;= ASPHI(R00)
1330 .MACRO WCSA(1)-A      ;;= ASPHI(R01)
1331 .MACRO FPA-R          ;;= BSPHI(R06)
1332 .MACRO CN5L-CNTL-B    ;;= BSPHI(R07)
1333 .MACRO CN5L-CADP-A    ;;= ASPHI(R07)
1334 .MACRO CN5L-SW-A      ;;= ASPHI(R06)
1335 .MACRO CN5L-TMP5W-A   ;;= ASPHI(R03)
1336
1337
1338
1339 .TOC *      ASP AND BSP INDIRECT REGISTER ADDRESSES
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404

```

IFBABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD
 ON BUS-A (A) OR BUS-B (B) USING INDIRECT ADDRESSING WITH THE IR,
 WHERE:
 SF<3,0>H = [FLPADR H + KTSRADR83 H] # [FLTPT L + IR6 H] # [IR7 H] # [IR6 H + ROR1 H]
 DF<3,0>H = [FLPADR H + KTDSTADR83 H] # [IR2 H] # [IR1 H] # [IR0 H]

.MACRO R[SF]-LO-A ;;= AEN/ASPLO,ASEL/SF
 .MACRO R[SF]-LO-R ;;= BEN/BSPLO,BSEL/SF

```

1351 .MACRO R[SF]-HI-A      ;;= AEN/ASPHI,ASEL/SF
1352 .MACRO R[SF]-HI-R      ;;= BEN/BSPHI,BSEL/SF
1353 .MACRO R[DF]-LO-A      ;;= AEN/ASPLO,ASEL/DF
1354 .MACRO R[DF]-LO-R      ;;= BEN/BSPLO,BSEL/DF
1355 .MACRO R[DF]-HI-A      ;;= AEN/ASPHI,ASEL/DF
1356 .MACRO R[DF]-HI-B      ;;= BEN/BSPHI,BSEL/DF
1357 .MACRO R[SF]-A         ;;= R[SF]-LO-A
1358 .MACRO R[SF]-B         ;;= R[SF]-LO-B
1359 .MACRO R[DF]-A         ;;= R[DF]-LO-A
1360 .MACRO R[DF]-B         ;;= R[DF]-LO-B
1361
1362
1363
1364 .TOC *      ASP, BSP INDIRECT ADDRESSING
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404

```

THESE MACROS ONLY SELECT THE ADDRESS MODE FOR THE ASP AND BSP;
 THE SELECTED SP IS NOT ENABLED ONTO THE BUS

.MACRO ASP-ADDRS-R[DF] ;;= ASEL/DF
 .MACRO ASP-ADDRS-R[SF] ;;= ASEL/SF
 .MACRO BSP-ADDRS-R[DF] ;;= BSEL/DF
 .MACRO BSP-ADDRS-R[SF] ;;= BSEL/SF

.TOC * ASP AND BSP DCS SPECIFIC FUNCTIONAL REGISTER ADDRESSES

ENABLE INPUT/OUTPUT (FOR READ AND/OR WRITE) OF THE APPROPRIATE SCRATCH PAD ONTO
 EITHER BUS-A "-A" OR BUS-B "-B" VIA FUNCTIONAL REGISTER DESIGNATION

.MACRO C000000-A ;;= ASPHI(C000000) IIN R01
 .MACRO C000000-B ;;= BSPHI(C000000) IIN R01
 .MACRO C177777-A ;;= ASPHI(C177777) IIN R03
 .MACRO C177777-R ;;= BSPHI(C177777) IIN R03
 .MACRO C125252-A ;;= ASPHI(C125252) IIN R05
 .MACRO C125252-B ;;= BSPHI(C125252) IIN R05
 .MACRO C052525-A ;;= ASPHI(C052525) IIN R07
 .MACRO C052525-B ;;= BSPHI(C052525) IIN R07
 .MACRO C000001-A ;;= ASPHI(C000001) IIN R11
 .MACRO C000001-R ;;= BSPHI(C000001) IIN R11
 .MACRO C100000-A ;;= ASPHI(C100000) IIN R13
 .MACRO C100000-R ;;= BSPHI(C100000) IIN R13
 .MACRO C000200-A ;;= ASPHI(C000200) IIN R15
 .MACRO C000200-B ;;= BSPHI(C000200) IIN R15

.PAGE=====

.TOC * WRITING THE C SCRATCH PAD

WRITE DATA ON BUS0IN (ACTUALLY DMUX OUTPUT) INTO ADDRESSED CSP LOCATION
 (SEE BELOW) DURING P3

```

1405 |
1406 |.MACRO WR-CSP          ::= WRCSP/YES
1407 |
1408 |
1409 |
1410 |.TOC *      CSP IMPLIED ADDRESSING
1411 |
1412 |  ENABLE FOR INPUT/OUTPUT [READ AND/OR WRITE] ONTO BUS-B ONLY A SPECIFIC CSP LOCATION,
1413 |  WHERE THE ADDRESS IS DETERMINED AS FOLLOWS:
1414 |
1415 |      CSPADDR<31:0>H = -f 0 0 0 0 RSEL<1>H 0 RSEL<0>H ]
1416 |
1417 |.MACRO CSPB(XX)        ::= BEN/BASCON,          IUSE IMMEDIATE MODE
1418 |                        RSEL/0XX                IWHICH ONE
1419 |
1420 |
1421 |
1422 |.TOC *      CSP DIRECT ADDRESSING
1423 |
1424 |  ENABLE FOR INPUT/OUTPUT [READ AND/OR WRITE] ONTO BUS-B ONLY A SPECIFIC CSP LOCATION,
1425 |  WHERE THE ADDRESS IS DETERMINED AS FOLLOWS:
1426 |
1427 |      CSPADDR<31:0>H = -WORD<23:20> H
1428 |
1429 |.MACRO CSPD(XX)        ::= BEN/CSP,              IUSE CSP-ADDR MODE
1430 |                        CSPADDR/0XX             IWHICH ONE
1431 |
1432 |
1433 |
1434 |.PAGE=====
1435 |
1436 |.TOC *      SHIFT TREE SPECIFICATION
1437 |  IN-B, MAY REQUIRE PRIOR SETUP OF RES-REGISTER FOR SHIFT END MUX SELECTION CONTROL
1438 |  (EG, WHEN ASEL/LEFT-1 IS USED).
1439 |
1440 |.TOC *      ENABLED ONTO BUS A
1441 |.MACRO D-RIGHT-14      ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/RIGHT-4,ASEL/RIGHT-2
1442 |.MACRO D-RIGHT-13      ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/RIGHT-4,ASEL/RIGHT-1
1443 |.MACRO D-RIGHT-12      ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/RIGHT-4,ASEL/DIRECT
1444 |.MACRO D-RIGHT-11      ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/RIGHT-4,ASEL/LEFT-1      ISENDMUX SETUP
1445 |.MACRO D-RIGHT-10      ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/DIRECT,ASEL/RIGHT-2
1446 |.MACRO D-RIGHT-9       ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/DIRECT,ASEL/RIGHT-1
1447 |.MACRO D-RIGHT-8       ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/DIRECT,ASEL/DIRECT
1448 |.MACRO D-RIGHT-7       ::= AEN/CMUX,AMUX/RIGHT-8,BMUX/DIRECT,ASEL/LEFT-1      ISENDMUX SETUP
1449 |.MACRO D-RIGHT-6       ::= AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT-4,ASEL/RIGHT-2
1450 |.MACRO D-RIGHT-5       ::= AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT-4,ASEL/RIGHT-1
1451 |.MACRO D-RIGHT-4       ::= AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT-4,ASEL/DIRECT
1452 |.MACRO D-RIGHT-3       ::= AEN/CMUX,AMUX/DIRECT,BMUX/RIGHT-4,ASEL/LEFT-1      ISENDMUX SETUP
1453 |.MACRO D-RIGHT-2       ::= AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/RIGHT-2
1454 |.MACRO D-RIGHT-1       ::= AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/RIGHT-1
1455 |.MACRO D-NO-SHIFT      ::= AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/DIRECT
1456 |.MACRO D-DIRECT        ::= D-NO-SHIFT
1457 |.MACRO D-LEFT-1        ::= AEN/CMUX,AMUX/DIRECT,BMUX/DIRECT,ASEL/LEFT-1      ISENDMUX SETUP
1458 |.MACRO D-SWAB          ::= AEN/CMUX,AMUX/SWAB,BMUX/DIRECT,ASEL/DIRECT

```

```

1459 |.MACRO D-SWAB-RIGHT-3  ::= AEN/CMUX,AMUX/SWAB,BMUX/RIGHT-4,ASEL/LEFT-1      ISENDMUX SETUP
1460 |.MACRO D-SWAB-LEFT-1   ::= AEN/CMUX,AMUX/SWAB,BMUX/DIRECT,ASEL/LEFT-1      ISENDMUX SETUP
1461 |.MACRO D-SIGNEXT        ::= AEN/CMUX,AMUX/SIGNEXT,BMUX/DIRECT,ASEL/DIRECT
1462 |.MACRO D-SIGNEXT-RIGHT-1 ::= AEN/CMUX,AMUX/SIGNEXT,BMUX/DIRECT,ASEL/RIGHT-1
1463 |.MACRO D-SIGNEXT-LEFT-1 ::= AEN/CMUX,AMUX/SIGNEXT,BMUX/DIRECT,ASEL/LEFT-1      ISENDMUX SETUP
1464 |.MACRO D-NO-SHIFT      ::= AEN/CMUX,BMUX/DIRECT,ASEL/DIRECT
1465 |.MACRO DIRECT          ::= NO-SHIFT
1466 |.MACRO COUNT#D(H1)     ::= AEN/CMUX,AMUX/COUNTER#D(H1),BMUX/DIRECT,ASEL/DIRECT
1467 |.MACRO COUNT#D(L0)     ::= AEN/CMUX,AMUX/COUNTER#D(L0),BMUX/DIRECT,ASEL/DIRECT
1468 |
1469 |
1470 |
1471 |.TOC *      FIRST TWO LEVELS ONLY (AMUX,BMUX)
1472 |  IN-B,1 FOR USE WHEN SHIFTING SR RIGHT, SR<15> <- BMUX<00>
1473 |.MACRO D-DIRECT(BMUX)  ::= AMUX/DIRECT,BMUX/DIRECT
1474 |
1475 |
1476 |
1477 |.PAGE=====
1478 |
1479 |.TOC *      ALU FUNCTIONS
1480 |  [SEE FIELD DESCRIPTION OF "ALU" FOR FULL DESCRIPTION]
1481 |.MACRO ZERO            ::= ALU/ZERO
1482 |.MACRO A-XOR-B         ::= ALU/A-XOR-B
1483 |.MACRO B               ::= ALU/B
1484 |.MACRO A-AND-B         ::= ALU/A-AND-B
1485 |.MACRO A-IOR-B         ::= ALU/A-IOR-B
1486 |.MACRO A               ::= ALU/A
1487 |.MACRO NOT-A           ::= ALU/NOT-A
1488 |.MACRO NOT-A-AND-B     ::= ALU/NOT-A-AND-B
1489 |.MACRO A-AND-NOT-B     ::= ALU/A-AND-NOT-B
1490 |
1491 |.MACRO DIVIDE          ::= ALU/DIVIDE
1492 |.MACRO A-PLUS-B        ::= ALU/A-PLUS-B
1493 |.MACRO A-MINUS-B       ::= ALU/A-MINUS-B
1494 |.MACRO A-PLUS-B-PLUS-PS[C] ::= ALU/A-PLUS-B-PLUS-PS[C]
1495 |.MACRO A-PLUS-B-PLUS-D[C] ::= ALU/A-PLUS-B-PLUS-D[C]
1496 |.MACRO A-PLUS-NOT-B-PLUS-D[C] ::= ALU/A-PLUS-NOT-B-PLUS-D[C]
1497 |.MACRO A-PLUS-B-PLUS-1 ::= ALU/A-PLUS-B-PLUS-1
1498 |
1499 |
1500 |.TOC *      COUT GENERATION
1501 |  [SEE FIELD DESCRIPTION OF "COUT" FOR FULL DESCRIPTION]
1502 |.MACRO COUT_CIN        ::= COUT/CIN
1503 |.MACRO COUT_PS[C]      ::= COUT/PS[C]
1504 |.MACRO COUT_ALU00      ::= COUT/ALU00
1505 |.MACRO COUT_ALU07      ::= COUT/ALU07
1506 |.MACRO COUT_ALU15      ::= COUT/ALU15
1507 |.MACRO COUT_COUT07     ::= COUT/COUT07
1508 |.MACRO COUT_COUT15     ::= COUT/COUT15
1509 |.MACRO COUT_D[C]       ::= COUT/D[C]
1510 |
1511 |
1512 |

```

```
1513 !.PAGE=====
1514
1515 .TOC *   CLOCKS
1516
1517
1518
1519 .TOC *   BASIC REGISTER CLOCKS (D, SR, BA, CC)
1520 .MACRO CLK-D          == CLKD/YES  !MUST SPECIFY P2 T OR P3 T
1521 .MACRO CLK-SR         == CLKS/YES  !MUST SPECIFY P2 T OR P3 T
1522 .MACRO CLK-BA         == CLKBA/YES !AT P1 T ONLY
1523 .MACRO SET-CC         == SCC/YES   !SETUP HERE, CLOCKED AT P2 T **OF NEXT UNWORD** ONLY
1524 .MACRO CLK-CC         == NULL      !IN NEXT UNWORD, FOR DOCUMENTATION
1525
1526
1527
1528 .TOC *   REDEFINED FROM SP RENAME FIELD (RES, COUNTER)
1529 .MACRO LOAD-RES        == MOD/LOADREG,LOADRES/YES !AT P2 T ONLY, FROM B-BUS<14:11>
1530 .MACRO LOAD-COUNTER   == MOD/LOADREG,LOADCOUNT/YES !DURING ENTIRE UNWORD, FROM B-BUS<7:0>
1531
1532
1533
1534 .TOC *   RES REGISTER CONTROL VALUES (FROM EMIT)
1535 !LOADED VIA: EMIT<14:11> -> CSP[XX]<14:11> -> B-BUS<14:11> -> RES<2:0>
1536 .MACRO SENDMUX-0123-SEL == EMIT14/1          !FOR SHIFT TREE
1537 .MACRO SENDMUX-4567-SEL == EMIT14/0          !FOR SHIFT TREE
1538 .MACRO SR-LOAD         == EMIT13/0,EMIT12/0    !FOR SR/GUARD
1539 .MACRO SP-LEFT         == EMIT13/0,EMIT12/1    !FOR SR/GUARD
1540 .MACRO SP-RIGHT        == EMIT13/1,EMIT12/0    !FOR SR/GUARD
1541 .MACRO SR-NOP          == EMIT13/1,EMIT12/1    !FOR SR/GUARD
1542 .MACRO GUARD-EN        == EMIT11/1            !FOR SR/GUARD
1543 .MACRO GUARD-DIS       == EMIT11/0            !FOR SR/GUARD
1544
1545
1546
1547
1548 .TOC *   CC CONTROL (FROM EMIT)
1549 !USED VIA: BUS-U37-H -> EMIT07-H -> MODIFY-V(1)-H
1550 .MACRO MODIFY-VBIT     == EMIT07/1
1551 .MACRO NOT-MODIFY-VBIT == EMIT07/0
1552
1553
1554
1555 !.PAGE=====
1556
1557 .TOC *   BUS CONTROL MACROS
1558 .MACRO DATI-CLKIP      == BEGIN/YES,SELECT/BUS,BUSCODE/DATI-CLKIP
1559 .MACRO DATI-NOINT      == BEGIN/YES,SELECT/BUS,BUSCODE/DATI-NOINT
1560 .MACRO DATI            == BEGIN/YES,SELECT/BUS,BUSCODE/DATI
1561 .MACRO DATI[P]         == BEGIN/YES,SELECT/BUS,BUSCODE/DATI[P]    !WITH ALTER/ALLOWED
1562 .MACRO DATO            == BEGIN/YES,SELECT/BUS,BUSCODE/DATO
1563 .MACRO DATIB           == BEGIN/YES,SELECT/BUS,BUSCODE/DATIB
1564 .MACRO DATIB[P]        == BEGIN/YES,SELECT/BUS,BUSCODE/DATIB[P]    !WITH ALTER/ALLOWED
1565 .MACRO DATIP           == BEGIN/YES,SELECT/BUS,BUSCODE/DATIP
1566 .MACRO DATOP           == BEGIN/YES,SELECT/BUS,BUSCODE/DATOP
```

```
1567 .MACRO INVALIDATE    == BEGIN/YES,SELECT/BUS,BUSCODE/INVALIDATE
1568
1569
1570
1571 .TOC *   KI/KJ CONTROL FUNCTIONS
1572
1573 !THESE BITS ACTUALLY ARISE OUT OF THE BASE MACHINE EXTENSION ROMS,
1574 !AND AS SUCH ARENT DIRECTLY ACCESSIBLE FROM THE DCS. THEY ARE
1575 !INCLUDED HERE ONLY FOR DOCUMENTATION PURPOSES.
1576
1577 .MACRO KJ-ENABLE      == KJ/ONE
1578
1579 .MACRO MAINTENANCE    == UKTEN/ONE
1580 .MACRO CURRENT-MODE   == UKT01/ONE,UKT00/ZERO
1581 .MACRO KERNAL-MODE    == UKT01/ONE,UKT00/ONE
1582 .MACRO MT-MODE        == UKT01/ZERO,UKT00/ONE
1583 .MACRO MF-MODE        == UKT01/ZERO,UKT00/ZERO
1584
1585
1586
1587 !.PAGE=====
1588
1589 .TOC *   UCON CONTROL MACROS
1590 .MACRO SET-UCON-CONTROL == BEGIN/YES,SELECT/UCON,UCON-LOAD/YES !LOAD UCON CONTROL REGISTER AT P0
1591 .MACRO UCON-OPERATION  == BEGIN/YES,SELECT/UCON,UCON-XFER/YES !PERFORM UCON OPERATION
1592
1593
1594
1595 .TOC *   PROCESSOR UCON CONTROL SETUP
1596 .MACRO UCON-PROC       == UCON-SEL-PROC/YES          !SELECT PROCESSOR
1597 .MACRO EN-CLK-TR[15-00] == UCON15/1                !ENABLE OPERATIONS
1598 .MACRO EN-CLK-PS[15-12] == UCON14/1
1599 .MACRO EN-CLK-FLAG[9-0] == UCON13/1
1600 .MACRO EN-CLK-FPS[7-4]  == UCON12/1
1601 .MACRO EN-CLK-PS[7-4]   == UCON11/1
1602 .MACRO EN-CLK-PS[3-0]   == UCON10/1
1603 .MACRO EN-CLK-UBREAK[11-00] == UCON09/1
1604 !UCON<8:7> ARE NOT USED IN PROCESSOR CONTROL
1605 .MACRO BUSDIN_EMIT[15-00] == UCON06/0,UCON05/0        !HBMUX SELECT
1606 .MACRO BUSDIN_CUA[14-03] == UCON06/0,UCON05/1
1607 .MACRO BUSDIN_PS[15-00]  == UCON06/1,UCON05/0
1608 .MACRO BUSDIN_FLAG[8-0]#FPS[7-0] == UCON06/1,UCON05/1
1609
1610
1611
1612
1613 .TOC *   DCS/WCS/ECS CONTROL
1614 .MACRO UCON-DCS        == UCON-SEL-WCS/YES          !SELECT DCS
1615 .MACRO BUSDIN_TNHA[11-00] == UCON14/0
1616 .MACRO BUSDIN_ERR#PROP#ERRCOD[11-00] == UCON14/1
1617 .MACRO START-DCS       == UCON15/1
1618
1619
1620
```

```

1621 .TOC * CACHE/KT UCON CONTROL
1622 .MACRO UCON-CACHE-KT          ;; UCON-SEL-CACHEKT/YES      ;SELECT CACHE / KT UCON FUNCTION
1623     UCON<15> NOT USED HERE
1624 .MACRO EN-KT-NO-RELOCATE      ;; UCON14/1                ;INHIBIT KT FROM ANY RELOCATION OF BA -> PBA
1625 .MACRO BUSDIN_BUS-INTERNAL-ADDR[15-00] ;; UCON13/0,UCON12/1 ;FROM INTERNAL ADDR ROM
1626 .MACRO BUSDIN_CPU-INTERNAL-ADDR[15-00] ;; UCON13/1,UCON12/1 ;DITTO ...
1627 .MACRO BUSDIN_MMP2[15-00]      ;; UCON11/1,UCON09/0 ;VIRTUAL PC
1628 .MACRO BUSDIN_CACHE-STATUS[15-00] ;; UCON11/1,UCON09/1 ;CACHE INFO
1629 .MACRO BUSDIN_KT-SEL          ;; UCON10/1                ;FOR PAR-8, PDR-8 ETC
1630 .MACRO KT-WRITE-HIGH          ;; UCON08/1                ;WRITE REGISTER <15:08>
1631 .MACRO KT-WRITE-LOW          ;; UCON07/1                ;WRITE REGISTER <07:00>
1632 .MACRO KT-WRITEP             ;; UCON08/1,UCON07/1         ;WRITE REGISTER <15:00>
1633 .MACRO KT-SEL-SLR/CCP        ;; UCON06/0,UCON05/0         ;SELECT KT-MUX OUTPUT
1634 .MACRO KT-SEL-MMR0           ;; UCON06/0,UCON05/1         ;
1635 .MACRO KT-SEL-PDR            ;; UCON06/1,UCON06/0         ;
1636 .MACRO KT-SEL-PAR            ;; UCON06/1,UCON06/1         ;
1637
1638
1639
1640
1641 .TOC * I/O UCON CONTROL
1642 .MACRO UCON-I-O              ;; UCON-SEL-I-O/YES          ;SELECT I-O CONTROL
1643
1644
1645
1646 .TOC * BUS CONTROL
1647 .MACRO EN-LOAD-DBUF[15-00]    ;; UCON15/1                ;EN LOAD DBUF AT P3
1648 .MACRO BUSDIN_DBUF[15-00]    ;; UCON15/1                ;DBUF ON BUSDIN
1649 .MACRO EN-STATUS-MUX         ;; UCON15/0                ;STATUS-MUX ENABLE ON BUSDIN
1650     UCON<14:11> ARE NOT USED IN UCON BUS CONTROL
1651 .MACRO BUSDIN_SERVICE[15-00] ;; UCON10/0,UCON09/1         ;
1652 .MACRO BUSDIN_JAM[15-00]     ;; UCON10/1,UCON09/0         ;
1653 .MACRO BUSDIN_PBA[15-00]     ;; UCON10/1,UCON09/1         ;
1654 .MACRO NMUX_CACHEDATA[15-00] ;; UCON09/1                ;
1655 .MACRO EN-RC-FCN=0           ;; UCON07/0,UCON06/0,UCON05/0 ;SELECT BUS CONTROL FUNCTION
1656 .MACRO EN-START-DELAY        ;; UCON07/0,UCON06/0,UCON05/1 ;
1657 .MACRO EN-CLR-JAM-ERRORS     ;; UCON07/0,UCON06/1,UCON05/0 ;
1658 .MACRO EN-CLR-NPR-TIMEOUT    ;; UCON07/0,UCON06/1,UCON05/1 ;
1659 .MACRO EN-CLR-PWR-FAIL       ;; UCON07/1,UCON06/0,UCON05/0 ;
1660 .MACRO EN-CLR-YELLOW-ZONE    ;; UCON07/1,UCON06/0,UCON05/1 ;
1661 .MACRO EN-ALLOW-BG[1:1]H    ;; UCON07/1,UCON06/1,UCON05/0 ;
1662 .MACRO EN-BUS-INVT-UCON      ;; UCON07/1,UCON06/1,UCON05/1 ;
1663
1664
1665
1666 .TOC * CONSOLE I-O
1667 .MACRO EN-CONSOLE-COMMAND    ;; UCON15/0,UCON14/0         ;SETS UP UCON I-O BITS FOR CONSOLE COMMANDS
1668                                     ;ALSO SELECTS STATUS-MUX ON BUSDIN
1669 .MACRO EN-CNSL-NOP           ;; UCON13/0,UCON12/0,UCON11/0 ;ENABLE CONSOLE NO OPERATION
1670 .MACRO EN-CLR-COUNTP         ;; UCON13/0,UCON12/0,UCON11/1 ;ENABLE CLEAR DIGIT PAIR COUNTER
1671 .MACRO EN-INC-COUNTP         ;; UCON13/0,UCON12/1,UCON11/0 ;ENABLE BUMP TO NEXT DIGIT PAIR
1672 .MACRO EN-CLR-CNSL-SRVC      ;; UCON13/0,UCON12/1,UCON11/1 ;ENABLE CLEAR CONSOLE SERVICE RST FLOP
1673 .MACRO EN-STPB-DTSP          ;; UCON13/1,UCON12/0,UCON11/0 ;ENABLE WRITE DIGIT PAIR TO DISPLAY LATCH
1674 .MACRO EN-CLR-CNSL          ;; UCON13/1,UCON12/0,UCON11/1 ;ENABLE CLEAR CONSOLE LED

```

```

1675 .MACRO EN-SET-CNSL          ;; UCON13/1,UCON12/1,UCON11/0 ;ENABLE SET CONSOLE LED
1676 .MACRO EN-SET-DP            ;; UCON13/1,UCON12/1,UCON11/1 ;ENABLE SET ALL DP LEDS
1677 .MACRO BUSDIN_CONSOLE[06-00] ;; UCON10/0,UCON09/0         ;STATUS-MUX SELECT
1678     UCON<8:5> ARE NOT USED IN UCON CONSOLE CONTROL
1679
1680
1681
1682 .TOC * REMOTE CONSOLE INTERFACE
1683 IN P.S. "EN CONSOLE COMMAND" DOES NOT APPLY TO REMOTE CONSOLE
1684 .MACRO EN-REMSRST            ;; UCON14/1                ;EN REMOTE CONSOLE STROBE
1685 .MACRO EN-REMCODE1          ;; UCON12/1                ;EN SPECIAL CODE 1
1686 .MACRO EN-REMCODE0          ;; UCON11/1                ;EN SPECIAL CODE 0
1687
1688
1689
1690
1691
1692
1693 .TOC * DCS ROM EXTENSION MACROS
1694
1695 .TOC * GENERAL FUNCTIONS
1696 .MACRO LOAD-ENUA(MICROADDR)   ;; LOAD-ENUA-ERRCOD/YES, ;SPECIFY LOAD
1697                                     ;ENUA/0MICROADDR ;AND VALUE
1698 .MACRO LOAD-ERROR(MICROADDR) ;; LOAD-ENUA-ERRCOD/YES ;FOR EFFECT ONLY - ALWAYS ACCOMPANIES ABOVE
1699 .MACRO BUMP-VERIFY            ;; VERIFY/BUMP           ;BUMP VERIFY COUNTER WHEN IN SELF TEST MODE
1700 .MACRO SIGNAL-EOP             ;; EOP/SIGNAL            ;SIGNAL END OF PASS
1701 .MACRO DCS-CTR(X)             ;; LOAD-DCS-CTR/YES, ;SPECIFY LOAD COUNTER (DCS1)
1702                                     ;CTR/0XX ;AND VALUE
1703
1704
1705
1706 .TOC * DAD<1:0> BIT FUNCTIONS
1707 .MACRO NO-DAD                 ;; DAD/NO-DAD             ;DON'T ASSERT DAD BITS
1708 .MACRO FIRST-1-OR-2           ;; DAD/FIRST-1-OR-2        ;SELECT CSP CONSTANT 1/2, FIRST USE
1709 .MACRO SECOND-1-OR-2          ;; DAD/SECOND-1-OR-2       ;SELECT CSP CONSTANT 1/2, SECOND USE
1710 .MACRO WRITE-BYTE             ;; DAD/WRITE-BYTE          ;BYTE WRITE ENABLE TO ASP/BSP
1711 .MACRO BYTE-WRITE             ;; DAD/WRITE-BYTE          ;
1712
1713
1714
1715 .TOC * DIAGNOSTIC MODE BUT ENABLES
1716 .MACRO BUTD(SCOPE)            ;; SCOPE/ENABLED         ;ENABLE SCOPE LOOPING CHECK
1717 .MACRO BUTD(ERROR)            ;; SCOPE/ENABLED         ;FORCES NUA00=0 IF ERROR[1]H SET
1718
1719 .MACRO BUTD(VERIFY-MODE)      ;; NULL                ;CHECK IMPLICITLY FOR VERIFY MODE
1720                                     ;"SIGNAL EOP" MUST BE PRESENT IN SAME WORD
1721
1722
1723 .MACRO BUTD(EOP-OVERFLOW)     ;; NULL                ;FORCES NUA01=0 IF VERIFY SWITCH SET
1724
1725 .MACRO BUTD(EOP-OVERFLOW)     ;; NULL                ;AFTER "SIGNAL EOP" GIVEN, ADDRESS IS FORCED
1726                                     ;TO (4000) IF HARDWARE EOP/VERIFY COUNTER
1727                                     ;HAS NOT YET OVERFLOWED, FORCING ANOTHER
1728                                     ;PASS, ELSE ADDRESS IN UPF IS TAKEN UNMODIFIED.

```

```
1729
1730 [.PAGE=====
1731
1732 .TOC * MICROBRANCH FIELD MACROS
1733 [ISFF <URF> FIELD DESCRIPTION FOR FULL INFO]
1734
1735 .MACRO BUT(XX)          ;; UBF/0XX          INACTIVE, FULL WIDTH
1736 .MACRO BUTR(XX)         ;; UBF/0XX          INACTIVE, RESTRICTED WIDTH
1737
1738 .MACRO BUTA(XX)         ;; UBF/0XX          IACTIVE, FULL WIDTH
1739 .MACRO BUTRA(XX)        ;; UBF/0XX          IACTIVE, RESTRICTED WIDTH
1740
1741 .MACRO TEST(XX)         ;; MULTIPLE/0XX     IFOR BUTR(MULTIPLE) SETUP
1742 .MACRO BUTM(XX)         ;; MULTIPLE/0XX,UBF/0XX IA MULTIPLE BUTR
1743
1744
1745
1746 [.PAGE=====
1747
1748 .TOC * MISCELLANEOUS
1749
1750 .TOC * OTHER SOURCES ENABLED FOR A-BUS
1751 .MACRO SR               ;; AEN/XMUX,ASELO/SR
1752 .MACRO FLTPT            ;; AEN/XMUX,ASELO/FLTPT
1753
1754
1755
1756 .TOC * PAGING, RETURN REGISTER
1757
1758 [PAGE FIELD ONLY:
1759 .MACRO PAGE(X)          ;; NEXT-PAGE/0X
1760
1761 [PAGE FIELD AND BUT(SUBR B):
1762 .MACRO GOTO-PAGE(X)     ;; NEXT-PAGE/0X,UBF/SUBR-B
1763
1764 [RETURN REGISTER <- D<14:03>, PAGE <- EMIT<02:00> ON BUTA(SUBR-A)
1765 .MACRO RETURN_D(14-03) ;; UBF/SUBR-A
1766
1767
1768
1769 [.PAGE=====
1770
1771 .TOC * ADVANCED OPERATIONS
1772
1773
1774
1775 .TOC * DATA INTO CSP, AT P3 ONLY
1776
1777 [N,R,1 BUSDIN IS ANY BUT EMIT (OVERLAPS BSEL<1:0>)]
1778 .MACRO CSPR[14]_RUSDIN ;; CSPR(B14),WR-CSP
1779 .MACRO CSPR[15]_RUSDIN ;; CSPR(B15),WR-CSP
1780 .MACRO CSPR[16]_RUSDIN ;; CSPR(B16),WR-CSP
1781 .MACRO CSPR[17]_RUSDIN ;; CSPR(B17),WR-CSP
1782
```

```
1783 [N,R,1 GETS WHATEVER IS ON BUSDIN
1784 .MACRO CSPD[00]_RUSDIN ;; CSPD(D00),WR-CSP
1785 .MACRO CSPD[01]_RUSDIN ;; CSPD(D01),WR-CSP
1786 .MACRO CSPD[02]_RUSDIN ;; CSPD(D02),WR-CSP
1787 .MACRO CSPD[03]_RUSDIN ;; CSPD(D03),WR-CSP
1788 .MACRO CSPD[04]_RUSDIN ;; CSPD(D04),WR-CSP
1789 .MACRO CSPD[05]_RUSDIN ;; CSPD(D05),WR-CSP
1790 .MACRO CSPD[06]_RUSDIN ;; CSPD(D06),WR-CSP
1791 .MACRO CSPD[07]_RUSDIN ;; CSPD(D07),WR-CSP
1792 .MACRO CSPD[08]_RUSDIN ;; CSPD(D08),WR-CSP
1793 .MACRO CSPD[09]_RUSDIN ;; CSPD(D09),WR-CSP
1794 .MACRO CSPD[10]_RUSDIN ;; CSPD(D10),WR-CSP
1795 .MACRO CSPD[11]_RUSDIN ;; CSPD(D11),WR-CSP
1796 .MACRO CSPD[12]_RUSDIN ;; CSPD(D12),WR-CSP
1797 .MACRO CSPD[13]_RUSDIN ;; CSPD(D13),WR-CSP
1798 .MACRO CSPD[14]_RUSDIN ;; CSPD(D14),WR-CSP
1799 .MACRO CSPD[15]_RUSDIN ;; CSPD(D15),WR-CSP
1800 .MACRO CSPD[16]_RUSDIN ;; CSPD(D16),WR-CSP
1801 .MACRO CSPD[17]_RUSDIN ;; CSPD(D17),WR-CSP
1802
1803 [N,R,1 REQUIP THAT BUSDIN_EMIT[15-00] PREVIOUSLY SET UP
1804 .MACRO CSPD[00]_EMIT ;; CSPD(D00),WR-CSP
1805 .MACRO CSPD[01]_EMIT ;; CSPD(D01),WR-CSP
1806 .MACRO CSPD[02]_EMIT ;; CSPD(D02),WR-CSP
1807 .MACRO CSPD[03]_EMIT ;; CSPD(D03),WR-CSP
1808 .MACRO CSPD[04]_EMIT ;; CSPD(D04),WR-CSP
1809 .MACRO CSPD[05]_EMIT ;; CSPD(D05),WR-CSP
1810 .MACRO CSPD[06]_EMIT ;; CSPD(D06),WR-CSP
1811 .MACRO CSPD[07]_EMIT ;; CSPD(D07),WR-CSP
1812 .MACRO CSPD[08]_EMIT ;; CSPD(D08),WR-CSP
1813 .MACRO CSPD[09]_EMIT ;; CSPD(D09),WR-CSP
1814 .MACRO CSPD[10]_EMIT ;; CSPD(D10),WR-CSP
1815 .MACRO CSPD[11]_EMIT ;; CSPD(D11),WR-CSP
1816 .MACRO CSPD[12]_EMIT ;; CSPD(D12),WR-CSP
1817 .MACRO CSPD[13]_EMIT ;; CSPD(D13),WR-CSP
1818 .MACRO CSPD[14]_EMIT ;; CSPD(D14),WR-CSP
1819 .MACRO CSPD[15]_EMIT ;; CSPD(D15),WR-CSP
1820 .MACRO CSPD[16]_EMIT ;; CSPD(D16),WR-CSP
1821 .MACRO CSPD[17]_EMIT ;; CSPD(D17),WR-CSP
1822
1823 .TOC * MISC CONSTANTS INTO ASP, BSP, AT P2-T * P3
1824
1825 .MACRO A#BSPHI(C1000001)_D ;; ASP(C1000000),BSP(C1000000),WR(AB,H,B)
1826 .MACRO A#BSPHI(C0002001)_D ;; ASP(C0002000),BSP(C0002000),WR(AB,H,B)
1827 .MACRO A#BSPHI(C0000001)_D ;; ASP(C0000000),BSP(C0000000),WR(AB,H,B)
1828 .MACRO A#BSPHI(C177777)_D ;; ASP(C177777),BSP(C177777),WR(AB,H,B)
1829 .MACRO A#BSPHI(C000001)_D ;; ASP(C000001),BSP(C000001),WR(AB,H,B)
1830 .MACRO A#BSPHI(C052525)_D ;; ASP(C052525),BSP(C052525),WR(AB,H,B)
1831
1832 .MACRO A#BSPHI(C125252)_D ;; ASP(C125252),BSP(C125252),WR(AB,H,B)
1833
1834 .MACRO A#BSPHI(C100000)_D-[B] ;; BSP(C100000),WR(AB,H,B)
1835 .MACRO A#BSPHI(C000200)_D-[B] ;; BSP(C000200),WR(AB,H,B)
1836 .MACRO A#BSPHI(C100000)_D-[B] ;; BSP(C000000),WR(AB,H,B)
1837 .MACRO A#BSPHI(C177777)_D-[R] ;; BSP(C177777),WR(AB,H,B)
1838 .MACRO A#BSPHI(C000001)_D-[B] ;; BSP(C000001),WR(AB,H,B)
1839 .MACRO A#BSPHI(C052525)_D-[R] ;; BSP(C052525),WR(AB,H,B)
```

```
1837 .MACRO A#BSPHI(C125252)_D=[B] 11= BSP(C125252),WR(AB,H,B)
1838
1839 .MACRO A#BSPHI(C100000)_D=[A] 11= ASP(C100000),WR(AB,H,A)
1840 .MACRO A#BSPHI(C000200)_D=[A] 11= ASP(C000200),WR(AB,H,A)
1841 .MACRO A#BSPHI(C000000)_D=[A] 11= ASP(C000000),WR(AB,H,A)
1842 .MACRO A#BSPHI(C177777)_D=[A] 11= ASP(C177777),WR(AB,H,A)
1843 .MACRO A#BSPHI(C000001)_D=[A] 11= ASP(C000001),WR(AB,H,A)
1844 .MACRO A#BSPHI(C052525)_D=[A] 11= ASP(C052525),WR(AB,H,A)
1845 .MACRO A#BSPHI(C125252)_D=[A] 11= ASP(C125252),WR(AB,H,A)
1846
1847
1848
1849 .T0C * DATA INTO ASP, BSP, AT P2-T * P3
1850
1851 .MACRO ASPLO[17]_CSPD(XX) 11= B,ASPLO(R17),CSPD(0XX),CLK=D,P2-T,WR(A,L,A)
1852 .MACRO ASPLO[17]_CSPD(XX) 11= B,ASPLO(R17),CSPD(0XX),CLK=D,P2-T,WR(A,L,A)
1853 .MACRO PC_D 11= PC=A,WR(AB,L,A)
1854 .MACRO RS_D 11= RS=A,WR(AB,L,A)
1855
1856 .MACRO ASPLO[00]_D 11= ASP(R00),WR(A,L,A)
1857 .MACRO ASPLO[01]_D 11= ASP(R01),WR(A,L,A)
1858 .MACRO ASPLO[02]_D 11= ASP(R02),WR(A,L,A)
1859 .MACRO ASPLO[03]_D 11= ASP(R03),WR(A,L,A)
1860 .MACRO ASPLO[04]_D 11= ASP(R04),WR(A,L,A)
1861 .MACRO ASPLO[05]_D 11= ASP(R05),WR(A,L,A)
1862 .MACRO ASPLO[06]_D 11= ASP(R06),WR(A,L,A)
1863 .MACRO ASPLO[07]_D 11= ASP(R07),WR(A,L,A)
1864 .MACRO ASPLO[10]_D 11= ASP(R10),WR(A,L,A)
1865 .MACRO ASPLO[11]_D 11= ASP(R11),WR(A,L,A)
1866 .MACRO ASPLO[12]_D 11= ASP(R12),WR(A,L,A)
1867 .MACRO ASPLO[13]_D 11= ASP(R13),WR(A,L,A)
1868 .MACRO ASPLO[14]_D 11= ASP(R14),WR(A,L,A)
1869 .MACRO ASPLO[15]_D 11= ASP(R15),WR(A,L,A)
1870 .MACRO ASPLO[16]_D 11= ASP(R16),WR(A,L,A)
1871 .MACRO ASPLO[17]_D 11= ASP(R17),WR(A,L,A)
1872
1873 .MACRO ASPHI[00]_D 11= ASP(R00),WR(A,H,A)
1874 .MACRO ASPHI[01]_D 11= ASP(R01),WR(A,H,A)
1875 .MACRO ASPHI[02]_D 11= ASP(R02),WR(A,H,A)
1876 .MACRO ASPHI[03]_D 11= ASP(R03),WR(A,H,A)
1877 .MACRO ASPHI[04]_D 11= ASP(R04),WR(A,H,A)
1878 .MACRO ASPHI[05]_D 11= ASP(R05),WR(A,H,A)
1879 .MACRO ASPHI[06]_D 11= ASP(R06),WR(A,H,A)
1880 .MACRO ASPHI[07]_D 11= ASP(R07),WR(A,H,A)
1881 .MACRO ASPHI[10]_D 11= ASP(R10),WR(A,H,A)
1882 .MACRO ASPHI[11]_D 11= ASP(R11),WR(A,H,A)
1883 .MACRO ASPHI[12]_D 11= ASP(R12),WR(A,H,A)
1884 .MACRO ASPHI[13]_D 11= ASP(R13),WR(A,H,A)
1885 .MACRO ASPHI[14]_D 11= ASP(R14),WR(A,H,A)
1886 .MACRO ASPHI[15]_D 11= ASP(R15),WR(A,H,A)
1887 .MACRO ASPHI[16]_D 11= ASP(R16),WR(A,H,A)
1888 .MACRO ASPHI[17]_D 11= ASP(R17),WR(A,H,A)
1889
1890 .MACRO BSPLO[00]_D 11= BSP(R00),WR(B,L,B)
```

```
1891 .MACRO BSPLO[01]_D 11= BSP(R01),WR(B,L,B)
1892 .MACRO BSPLO[02]_D 11= BSP(R02),WR(B,L,B)
1893 .MACRO BSPLO[03]_D 11= BSP(R03),WR(B,L,B)
1894 .MACRO BSPLO[04]_D 11= BSP(R04),WR(B,L,B)
1895 .MACRO BSPLO[05]_D 11= BSP(R05),WR(B,L,B)
1896 .MACRO BSPLO[06]_D 11= BSP(R06),WR(B,L,B)
1897 .MACRO BSPLO[07]_D 11= BSP(R07),WR(B,L,B)
1898 .MACRO BSPLO[10]_D 11= BSP(R10),WR(B,L,B)
1899 .MACRO BSPLO[11]_D 11= BSP(R11),WR(B,L,B)
1900 .MACRO BSPLO[12]_D 11= BSP(R12),WR(B,L,B)
1901 .MACRO BSPLO[13]_D 11= BSP(R13),WR(B,L,B)
1902 .MACRO BSPLO[14]_D 11= BSP(R14),WR(B,L,B)
1903 .MACRO BSPLO[15]_D 11= BSP(R15),WR(B,L,B)
1904 .MACRO BSPLO[16]_D 11= BSP(R16),WR(B,L,B)
1905 .MACRO BSPLO[17]_D 11= BSP(R17),WR(B,L,B)
1906
1907 .MACRO BSPHI[00]_D 11= BSP(R00),WR(B,H,B)
1908 .MACRO BSPHI[01]_D 11= BSP(R01),WR(B,H,B)
1909 .MACRO BSPHI[02]_D 11= BSP(R02),WR(B,H,B)
1910 .MACRO BSPHI[03]_D 11= BSP(R03),WR(B,H,B)
1911 .MACRO BSPHI[04]_D 11= BSP(R04),WR(B,H,B)
1912 .MACRO BSPHI[05]_D 11= BSP(R05),WR(B,H,B)
1913 .MACRO BSPHI[06]_D 11= BSP(R06),WR(B,H,B)
1914 .MACRO BSPHI[07]_D 11= BSP(R07),WR(B,H,B)
1915 .MACRO BSPHI[10]_D 11= BSP(R10),WR(B,H,B)
1916 .MACRO BSPHI[11]_D 11= BSP(R11),WR(B,H,B)
1917 .MACRO BSPHI[12]_D 11= BSP(R12),WR(B,H,B)
1918 .MACRO BSPHI[13]_D 11= BSP(R13),WR(B,H,B)
1919 .MACRO BSPHI[14]_D 11= BSP(R14),WR(B,H,B)
1920 .MACRO BSPHI[15]_D 11= BSP(R15),WR(B,H,B)
1921 .MACRO BSPHI[16]_D 11= BSP(R16),WR(B,H,B)
1922 .MACRO BSPHI[17]_D 11= BSP(R17),WR(B,H,B)
1923
1924 .MACRO A#BSPLO[00]_D 11= ASP(R00),BSP(R00),WR(AB,L,A)
1925 .MACRO A#BSPLO[01]_D 11= ASP(R01),BSP(R01),WR(AB,L,A)
1926 .MACRO A#BSPLO[02]_D 11= ASP(R02),BSP(R02),WR(AB,L,A)
1927 .MACRO A#BSPLO[03]_D 11= ASP(R03),BSP(R03),WR(AB,L,A)
1928 .MACRO A#BSPLO[04]_D 11= ASP(R04),BSP(R04),WR(AB,L,A)
1929 .MACRO A#BSPLO[05]_D 11= ASP(R05),BSP(R05),WR(AB,L,A)
1930 .MACRO A#BSPLO[06]_D 11= ASP(R06),BSP(R06),WR(AB,L,A)
1931 .MACRO A#BSPLO[07]_D 11= ASP(R07),BSP(R07),WR(AB,L,A)
1932 .MACRO A#BSPLO[10]_D 11= ASP(R10),BSP(R10),WR(AB,L,A)
1933 .MACRO A#BSPLO[11]_D 11= ASP(R11),BSP(R11),WR(AB,L,A)
1934 .MACRO A#BSPLO[12]_D 11= ASP(R12),BSP(R12),WR(AB,L,A)
1935 .MACRO A#BSPLO[13]_D 11= ASP(R13),BSP(R13),WR(AB,L,A)
1936 .MACRO A#BSPLO[14]_D 11= ASP(R14),BSP(R14),WR(AB,L,A)
1937
1938 .MACRO A#BSPLO[15]_D 11= ASP(R15),BSP(R15),WR(AB,L,A)
1939 .MACRO A#BSPLO[16]_D 11= ASP(R16),BSP(R16),WR(AB,L,A)
1940 .MACRO A#BSPLO[17]_D 11= ASP(R17),BSP(R17),WR(AB,L,A)
1941
1942 .MACRO A#BSPHI[00]_D 11= ASP(R00),BSP(R00),WR(AB,H,A)
1943 .MACRO A#BSPHI[01]_D 11= ASP(R01),BSP(R01),WR(AB,H,A)
1944 .MACRO A#BSPHI[02]_D 11= ASP(R02),BSP(R02),WR(AB,H,A)
1945 .MACRO A#BSPHI[03]_D 11= ASP(R03),BSP(R03),WR(AB,H,A)
```

```
1945 .MACRO A=BSPI(04)_D      11= ASP(P04),BSP(R04),WR(AB,H,A)
1946 .MACRO A=BSPI(05)_D      11= ASP(P05),BSP(R05),WR(AB,H,A)
1947 .MACRO A=BSPI(06)_D      11= ASP(P06),BSP(R06),WR(AB,H,A)
1948 .MACRO A=BSPI(07)_D      11= ASP(P07),BSP(R07),WR(AB,H,A)
1949 .MACRO A=BSPI(10)_D      11= ASP(R10),BSP(R10),WR(AB,H,A)
1950 .MACRO A=BSPI(11)_D      11= ASP(R11),BSP(R11),WR(AB,H,A)
1951 .MACRO A=BSPI(12)_D      11= ASP(R12),BSP(R12),WR(AB,H,A)
1952 .MACRO A=BSPI(13)_D      11= ASP(R13),BSP(R13),WR(AB,H,A)
1953 .MACRO A=BSPI(14)_D      11= ASP(R14),BSP(R14),WR(AB,H,A)
1954 .MACRO A=BSPI(15)_D      11= ASP(R15),BSP(R15),WR(AB,H,A)
1955 .MACRO A=BSPI(16)_D      11= ASP(R16),BSP(R16),WR(AB,H,A)
1956 .MACRO A=BSPI(17)_D      11= ASP(R17),BSP(R17),WR(AB,H,A)
1957
1958 .MACRO A=BSPI(00)_D-[A]    11= ASP(R00),WR(AB,L,A)
1959 .MACRO A=BSPI(01)_D-[A]    11= ASP(R01),WR(AB,L,A)
1960 .MACRO A=BSPI(02)_D-[A]    11= ASP(R02),WR(AB,L,A)
1961 .MACRO A=BSPI(03)_D-[A]    11= ASP(R03),WR(AB,L,A)
1962 .MACRO A=BSPI(04)_D-[A]    11= ASP(R04),WR(AB,L,A)
1963 .MACRO A=BSPI(05)_D-[A]    11= ASP(R05),WR(AB,L,A)
1964 .MACRO A=BSPI(06)_D-[A]    11= ASP(R06),WR(AB,L,A)
1965 .MACRO A=BSPI(07)_D-[A]    11= ASP(R07),WR(AB,L,A)
1966 .MACRO A=BSPI(10)_D-[A]    11= ASP(R10),WR(AB,L,A)
1967 .MACRO A=BSPI(11)_D-[A]    11= ASP(R11),WR(AB,L,A)
1968 .MACRO A=BSPI(12)_D-[A]    11= ASP(R12),WR(AB,L,A)
1969 .MACRO A=BSPI(13)_D-[A]    11= ASP(R13),WR(AB,L,A)
1970 .MACRO A=BSPI(14)_D-[A]    11= ASP(R14),WR(AB,L,A)
1971 .MACRO A=BSPI(15)_D-[A]    11= ASP(R15),WR(AB,L,A)
1972 .MACRO A=BSPI(16)_D-[A]    11= ASP(R16),WR(AB,L,A)
1973 .MACRO A=BSPI(17)_D-[A]    11= ASP(R17),WR(AB,L,A)
1974
1975 .MACRO A=BSPI(00)_D-[A]    11= ASP(R00),WR(AB,H,A)
1976 .MACRO A=BSPI(01)_D-[A]    11= ASP(R01),WR(AB,H,A)
1977 .MACRO A=BSPI(02)_D-[A]    11= ASP(R02),WR(AB,H,A)
1978 .MACRO A=BSPI(03)_D-[A]    11= ASP(R03),WR(AB,H,A)
1979 .MACRO A=BSPI(04)_D-[A]    11= ASP(R04),WR(AB,H,A)
1980 .MACRO A=BSPI(05)_D-[A]    11= ASP(R05),WR(AB,H,A)
1981 .MACRO A=BSPI(06)_D-[A]    11= ASP(R06),WR(AB,H,A)
1982 .MACRO A=BSPI(07)_D-[A]    11= ASP(R07),WR(AB,H,A)
1983 .MACRO A=BSPI(10)_D-[A]    11= ASP(R10),WR(AB,H,A)
1984 .MACRO A=BSPI(11)_D-[A]    11= ASP(R11),WR(AB,H,A)
1985 .MACRO A=BSPI(12)_D-[A]    11= ASP(R12),WR(AB,H,A)
1986 .MACRO A=BSPI(13)_D-[A]    11= ASP(R13),WR(AB,H,A)
1987 .MACRO A=BSPI(14)_D-[A]    11= ASP(R14),WR(AB,H,A)
1988 .MACRO A=BSPI(15)_D-[A]    11= ASP(R15),WR(AB,H,A)
1989 .MACRO A=BSPI(16)_D-[A]    11= ASP(R16),WR(AB,H,A)
1990 .MACRO A=BSPI(17)_D-[A]    11= ASP(R17),WR(AB,H,A)
1991
1992 .MACRO A=BSPI(00)_D-[B]    11= BSP(R00),WR(AB,L,B)
1993 .MACRO A=BSPI(01)_D-[B]    11= BSP(R01),WR(AB,L,B)
1994 .MACRO A=BSPI(02)_D-[B]    11= BSP(R02),WR(AB,L,B)
1995 .MACRO A=BSPI(03)_D-[B]    11= BSP(R03),WR(AB,L,B)
1996 .MACRO A=BSPI(04)_D-[B]    11= BSP(R04),WR(AB,L,B)
1997 .MACRO A=BSPI(05)_D-[B]    11= BSP(R05),WR(AB,L,B)
1998 .MACRO A=BSPI(06)_D-[B]    11= BSP(R06),WR(AB,L,B)
```

```
1999 .MACRO A=BSPI(07)_D-[B]    11= BSP(R07),WR(AB,L,B)
2000 .MACRO A=BSPI(10)_D-[B]    11= BSP(R10),WR(AB,L,B)
2001 .MACRO A=BSPI(11)_D-[B]    11= BSP(R11),WR(AB,L,B)
2002 .MACRO A=BSPI(12)_D-[B]    11= BSP(R12),WR(AB,L,B)
2003 .MACRO A=BSPI(13)_D-[B]    11= BSP(R13),WR(AB,L,B)
2004 .MACRO A=BSPI(14)_D-[B]    11= BSP(R14),WR(AB,L,B)
2005 .MACRO A=BSPI(15)_D-[B]    11= BSP(R15),WR(AB,L,B)
2006 .MACRO A=BSPI(16)_D-[B]    11= BSP(R16),WR(AB,L,B)
2007 .MACRO A=BSPI(17)_D-[B]    11= BSP(R17),WR(AB,L,B)
2008
2009 .MACRO A=BSPI(00)_D-[B]    11= BSP(R00),WR(AB,H,B)
2010 .MACRO A=BSPI(01)_D-[B]    11= BSP(R01),WR(AB,H,B)
2011 .MACRO A=BSPI(02)_D-[B]    11= BSP(R02),WR(AB,H,B)
2012 .MACRO A=BSPI(03)_D-[B]    11= BSP(R03),WR(AB,H,B)
2013 .MACRO A=BSPI(04)_D-[B]    11= BSP(R04),WR(AB,H,B)
2014 .MACRO A=BSPI(05)_D-[B]    11= BSP(R05),WR(AB,H,B)
2015 .MACRO A=BSPI(06)_D-[B]    11= BSP(R06),WR(AB,H,B)
2016 .MACRO A=BSPI(07)_D-[B]    11= BSP(R07),WR(AB,H,B)
2017 .MACRO A=BSPI(10)_D-[B]    11= BSP(R10),WR(AB,H,B)
2018 .MACRO A=BSPI(11)_D-[B]    11= BSP(R11),WR(AB,H,B)
2019 .MACRO A=BSPI(12)_D-[B]    11= BSP(R12),WR(AB,H,B)
2020 .MACRO A=BSPI(13)_D-[B]    11= BSP(R13),WR(AB,H,B)
2021 .MACRO A=BSPI(14)_D-[B]    11= BSP(R14),WR(AB,H,B)
2022 .MACRO A=BSPI(15)_D-[B]    11= BSP(R15),WR(AB,H,B)
2023 .MACRO A=BSPI(16)_D-[B]    11= BSP(R16),WR(AB,H,B)
2024 .MACRO A=BSPI(17)_D-[B]    11= BSP(R17),WR(AB,H,B)
2025
2026 .MACRO ASPLO(DF)_D      11= ASP-ADDRS-R(DF),WR(A,L,A)
2027 .MACRO ASPHI(DF)_D      11= ASP-ADDRS-R(DF),WR(A,H,A)
2028 .MACRO BSPLO(DF)_D      11= BSP-ADDRS-R(DF),WR(B,L,B)
2029 .MACRO BSPHI(DF)_D      11= BSP-ADDRS-R(DF),WR(B,H,B)
2030
2031 .MACRO ASPLO(SF)_D      11= ASP-ADDRS-R(SF),WR(A,L,A)
2032 .MACRO ASPHI(SF)_D      11= ASP-ADDRS-R(SF),WR(A,H,A)
2033 .MACRO BSPLO(SF)_D      11= BSP-ADDRS-R(SF),WR(B,L,B)
2034 .MACRO BSPHI(SF)_D      11= BSP-ADDRS-R(SF),WR(B,H,B)
2035
2036 .MACRO A=BSPI(DF)_D-[A]    11= ASP-ADDRS-R(DF),WR(AB,L,A)
2037 .MACRO A=BSPI(DF)_D-[A]    11= ASP-ADDRS-R(DF),WR(AB,H,A)
2038 .MACRO A=BSPI(DF)_D-[B]    11= BSP-ADDRS-R(DF),WR(AB,L,B)
2039 .MACRO A=BSPI(DF)_D-[B]    11= BSP-ADDRS-R(DF),WR(AB,H,B)
2040
2041 .MACRO A=BSPI(DF)_D-[A]    11= ASP-ADDRS-R(SF),WR(AB,L,A)
2042 .MACRO A=BSPI(DF)_D-[A]    11= ASP-ADDRS-R(SF),WR(AB,H,A)
2043 .MACRO A=BSPI(DF)_D-[B]    11= BSP-ADDRS-R(SF),WR(AB,L,B)
2044 .MACRO A=BSPI(DF)_D-[B]    11= BSP-ADDRS-R(SF),WR(AB,H,B)
2045
2046 .MACRO A=BSPI(DF)_D      11= ASP-ADDRS-R(SF),BSP-ADDRS-R(SF),WR(AB,L,A)
2047 .MACRO A=BSPI(DF)_D      11= ASP-ADDRS-R(DF),BSP-ADDRS-R(DF),WR(AB,L,A)
2048
2049 .MACRO A=BSPI(SF)_D      11= ASP-ADDRS-R(SF),BSP-ADDRS-R(SF),WR(AB,H,A)
2050 .MACRO A=BSPI(DF)_D      11= ASP-ADDRS-R(DF),BSP-ADDRS-R(DF),WR(AB,H,A)
2051
2052
```

```

2053 1.PAGE=====
2054
2055 .TNC * D AND SR <= (BUS-A FCN BUS-B), AT P2-T OR P3-T
2056
2057 !LOGIC FUNCTIONS:
2058 .MACRO SR_ZERO          ;; ZERO,CLK-SR
2059 .MACRO SR_A-XOR-B       ;; A-XOR-B,CLK-SR
2060 .MACRO SR_B             ;; B,CLK-SR
2061 .MACRO SR_A-AND-B       ;; A-AND-B,CLK-SR
2062 .MACRO SR_A-IOR-B       ;; A-IOR-B,CLK-SR
2063 .MACRO SR_A             ;; A,CLK-SR
2064 .MACRO SR_NOT-A         ;; NOT-A,CLK-SR
2065 .MACRO SR_NOT-A-AND-B   ;; NOT-A-AND-B,CLK-SR
2066 .MACRO SR_A-AND-NOT-B   ;; A-AND-NOT-B,CLK-SR
2067 .MACRO D_ZERO           ;; ZERO,CLK-D
2068 .MACRO D_A-XOR-B        ;; A-XOR-B,CLK-D
2069 .MACRO D_B              ;; B,CLK-D
2070 .MACRO D_A-AND-B        ;; A-AND-B,CLK-D
2071 .MACRO D_A-IOR-B        ;; A-IOR-B,CLK-D
2072 .MACRO D_A              ;; A,CLK-D
2073 .MACRO D_NOT-A          ;; NOT-A,CLK-D
2074 .MACRO D_NOT-A-AND-B    ;; NOT-A-AND-B,CLK-D
2075 .MACRO D_A-AND-NOT-B    ;; A-AND-NOT-B,CLK-D
2076
2077 !ARITH FUNCTIONS:
2078 .MACRO D_DIVIDE-STEP     ;; DIVIDE,CLK-D
2079 .MACRO D_A-PLUS-B        ;; A-PLUS-B,CLK-D
2080 .MACRO D_A-PLUS-B-PLUS-0 ;; A-PLUS-B,CLK-D
2081 .MACRO D_A-MINUS-B       ;; A-MINUS-B,CLK-D
2082 .MACRO D_A-PLUS-B-PLUS-PS(C) ;; A-PLUS-B-PLUS-PS(C),CLK-D
2083 .MACRO D_A-PLUS-B-PLUS-D(C) ;; A-PLUS-B-PLUS-D(C),CLK-D
2084 .MACRO D_A-PLUS-NOT-B-PLUS-D(C) ;; A-PLUS-NOT-B-PLUS-D(C),CLK-D
2085 .MACRO D_A-PLUS-B-PLUS-1 ;; A-PLUS-B-PLUS-1,CLK-D
2086 .MACRO SR_DIVIDE-STEP   ;; DIVIDE,CLK-SR
2087 .MACRO SR_A-PLUS-B       ;; A-PLUS-B,CLK-SR
2088 .MACRO SR_A-PLUS-B-PLUS-0 ;; A-PLUS-B,CLK-SR
2089 .MACRO SR_A-MINUS-B      ;; A-MINUS-B,CLK-SR
2090 .MACRO SR_A-PLUS-B-PLUS-PS(C) ;; A-PLUS-B-PLUS-PS(C),CLK-SR
2091 .MACRO SR_A-PLUS-B-PLUS-D(C) ;; A-PLUS-B-PLUS-D(C),CLK-SR
2092 .MACRO SR_A-PLUS-NOT-B-PLUS-D(C) ;; A-PLUS-NOT-B-PLUS-D(C),CLK-SR
2093 .MACRO SR_A-PLUS-B-PLUS-1 ;; A-PLUS-B-PLUS-1,CLK-SR
2094
2095
2096
2097
2098 .TNC * D(C) GETS SET
2099
2100 .MACRO D(C)_CINMUX      ;; CLK-D,COUT_CIN
2101 .MACRO D(C)_I           ;; CLK-D,COUT_CIN      INEDS SPECIFIC ALU/---
2102 .MACRO D(C)_O           ;; CLK-D,COUT_CIN      INEDS SPECIFIC ALU/---
2103 .MACRO D(C)_PS(C)       ;; CLK-D,COUT_PS(C)
2104 .MACRO D(C)_ALU00        ;; CLK-D,COUT_ALU00
2105 .MACRO D(C)_ALU07        ;; CLK-D,COUT_ALU07
2106 .MACRO D(C)_ALU15        ;; CLK-D,COUT_ALU15

```

```

2107 .MACRO D(C)_COUT07      ;; CLK-D,COUT_COUT07
2108 .MACRO D(C)_COUT15      ;; CLK-D,COUT_COUT15
2109 .MACRO D(C)_D(C)        ;; CLK-D,COUT_D(C)
2110 .MACRO D(C)_D(C)        ;; CLK-D,COUT_D(C)
2111
2112
2113
2114
2115
2116 .TNC * D-REGISTER <= (BUS = ABUS), BITWISE, AT P2-T OR P3-T
2117
2118 !N.B.: SHIFT TREE ENABLED SEPARATELY
2119 .MACRO D_D-SHIFTED-XOR-CSPB(XX) ;; A-XOR-B,CSPB(XX),CLK-D
2120 .MACRO D_D-SHIFTED-XOR-BSPHI(XX) ;; A-XOR-B,BSPHI(XX),CLK-D
2121
2122 .MACRO D_FLTPT-XOR-CSPB(XX)      ;; A-XOR-B,FLTPT,CSPB(XX),CLK-D
2123 .MACRO D_FLTPT-XOR-CSPD(XX)      ;; A-XOR-B,FLTPT,CSPD(XX),CLK-D
2124 .MACRO D_FLTPT-XOR-BSPHI(XX)     ;; A-XOR-B,FLTPT,BSPHI(XX),CLK-D
2125
2126 .MACRO D_SR-XOR-CSPB(XX)         ;; A-XOR-B,SR,CSPB(XX),CLK-D
2127 .MACRO D_SR-XOR-CSPD(XX)         ;; A-XOR-B,SR,CSPD(XX),CLK-D
2128 .MACRO D_SR-XOR-BSPHI(XX)        ;; A-XOR-B,SR,BSPHI(XX),CLK-D
2129
2130 .MACRO D_ASPL0(17)-XOR-CSPD(XX)  ;; A-XOR-B,ASPL0(R17),CSPD(XX),CLK-D
2131 .MACRO D_ASPL0(07)-XOR-BSPHI(XX) ;; A-XOR-B,ASPL0(R07),BSPHI(XX),CLK-D
2132 .MACRO D_ASPL0(05)-XOR-BSPHI(XX) ;; A-XOR-B,ASPL0(R05),BSPHI(XX),CLK-D
2133
2134 .MACRO D_SR-XOR-BSPLO(SF)        ;; A-XOR-B,SR,R(SF)-LO-B,CLK-D
2135 .MACRO D_SR-XOR-BSPHI(DF)        ;; A-XOR-B,SR,R(DF)-HI-B,CLK-D
2136
2137 .MACRO D_ASPL0(DF)-XOR-BSPHI(SF) ;; A-XOR-B,R(DF)-LO-A,R(SF)-HI-B,CLK-D
2138 .MACRO D_ASPL0(SF)-XOR-BSPLO(DF) ;; A-XOR-B,R(SF)-HI-A,R(DF)-LO-B,CLK-D
2139
2140 .MACRO D_CSPD(05)-XOR-ASPL0(XX)  ;; A-XOR-B,CSPD(D05),ASPL0(XX),CLK-D
2141 .MACRO D_CSPD(05)-XOR-ASPHI(XX)  ;; A-XOR-B,CSPD(D05),ASPHI(XX),CLK-D
2142 .MACRO D_CSPD(06)-XOR-ASPL0(XX)  ;; A-XOR-B,CSPD(D06),ASPL0(XX),CLK-D
2143 .MACRO D_CSPD(06)-XOR-ASPHI(XX)  ;; A-XOR-B,CSPD(D06),ASPHI(XX),CLK-D
2144 .MACRO D_CSPD(17)-XOR-ASPHI(XX)  ;; A-XOR-B,CSPD(D17),ASPHI(XX),CLK-D
2145
2146 .MACRO D_ASPL0(R21)-XOR-BSPLO(XX) ;; A-XOR-B,ASPL0(R02),BSPLO(XX),CLK-D
2147 .MACRO D_ASPL0(R03)-XOR-BSPLO(XX) ;; A-XOR-B,ASPL0(R03),BSPLO(XX),CLK-D
2148 .MACRO D_ASPL0(R04)-XOR-BSPLO(XX) ;; A-XOR-B,ASPL0(R04),BSPLO(XX),CLK-D
2149 .MACRO D_ASPL0(R05)-XOR-BSPLO(XX) ;; A-XOR-B,ASPL0(R05),BSPLO(XX),CLK-D
2150
2151
2152
2153 .TNC * D-REGISTER <= D-REGISTER THRU SHIFT-TREE
2154
2155 .MACRO D_D-RIGHT-14          ;; A,D-RIGHT-14,CLK-D
2156 .MACRO D_D-RIGHT-13          ;; A,D-RIGHT-13,CLK-D
2157 .MACRO D_D-RIGHT-12          ;; A,D-RIGHT-12,CLK-D
2158 .MACRO D_D-RIGHT-11          ;; A,D-RIGHT-11,CLK-D
2159 .MACRO D_D-RIGHT-10          ;; A,D-RIGHT-10,CLK-D
2160 .MACRO D_D-RIGHT-9           ;; A,D-RIGHT-9,CLK-D

```

```

2161 .MACRO D_D=RIGHT=8          ;; A,D=RIGHT=8,CLK=D
2162 .MACRO D_D=RIGHT=7          ;; A,D=RIGHT=7,CLK=D
2163 .MACRO D_D=RIGHT=6          ;; A,D=RIGHT=6,CLK=D
2164 .MACRO D_D=RIGHT=5          ;; A,D=RIGHT=5,CLK=D
2165 .MACRO D_D=RIGHT=4          ;; A,D=RIGHT=4,CLK=D
2166 .MACRO D_D=RIGHT=3          ;; A,D=RIGHT=3,CLK=D
2167 .MACRO D_D=RIGHT=2          ;; A,D=RIGHT=2,CLK=D
2168 .MACRO D_D=RIGHT=1          ;; A,D=RIGHT=1,CLK=D
2169 .MACRO D_D=NO-SHIFT          ;; A,D=NO-SHIFT,CLK=D
2170 .MACRO D_D=DIRECT           ;; A,D=DIRECT,CLK=D
2171 .MACRO D_D                  ;; A,D=DIRECT,CLK=D
2172 .MACRO D_SAVF=D              ;; A,D=DIRECT,CLK=D
2173 .MACRO D_D=LEFT=1           ;; A,D=LEFT=1,CLK=D
2174 .MACRO D_D=SWAB              ;; A,D=SWAB,CLK=D
2175 .MACRO D_D=SWAB=RIGHT=3      ;; A,D=SWAB=RIGHT=3,CLK=D
2176 .MACRO D_D=SWAB=LEFT=1       ;; A,D=SWAB=LEFT=1,CLK=D
2177 .MACRO D_D=SIGNEXT           ;; A,D=SIGNEXT,CLK=D
2178 .MACRO D_D=SIGNEXT=RIGHT=1   ;; A,D=SIGNEXT=RIGHT=1,CLK=D
2179 .MACRO D_D=SIGNEXT=LEFT=1     ;; A,D=SIGNEXT=LEFT=1,CLK=D
2180 .MACRO D_D=NO-SHIFT          ;; A,D=NO-SHIFT,CLK=D
2181 .MACRO D_D=DIRECT            ;; A,D=DIRECT,CLK=D
2182 .MACRO D_COUNT#D(HI)         ;; A,COUNT#D(HI),CLK=D
2183 .MACRO D_COUNT#D(LO)         ;; A,COUNT#D(LO),CLK=D
2184
2185
2186
2187 .TOC *      D <- WHATEVER'S LEFT, AT P2-T OR P3-T
2188
2189 .MACRO D_NOT=ASPHI(XX)        ;; NOT=A,ASPHI(0XX),CLK=D
2190 .MACRO D_NOT=ASPLO(XX)        ;; NOT=A,ASPLO(0XX),CLK=D
2191 .MACRO D_NOT=CSPB(XX)         ;; A=AND=NOT=B,C177777=A,CSPB(0XX),CLK=D
2192 .MACRO D_NOT=CSPD(XX)         ;; A=AND=NOT=B,C177777=A,CSPD(0XX),CLK=D
2193
2194 .MACRO D_CSPD(XX)             ;; B,CSPD(0XX),CLK=D
2195 .MACRO D_CSPB(XX)             ;; B,CSPB(0XX),CLK=D
2196 .MACRO D_CSPB(161#D(C)_1)     ;; A=IOR=B,C000000=A,CSPB(B16),CLK=D,D(C)_CINMUX
2197
2198 .MACRO D_BSPHI(XX)            ;; B,BSPHI(0XX),CLK=D
2199 .MACRO D_BSPLO(XX)            ;; B,BSPLO(0XX),CLK=D
2200 .MACRO D_ASPHI(XX)            ;; A,ASPHI(0XX),CLK=D
2201 .MACRO D_ASPLO(XX)            ;; A,ASPLO(0XX),CLK=D
2202
2203 .MACRO D_ASPLN(DF)            ;; A,R(DF)=LO=A,CLK=D
2204 .MACRO D_ASPHI(DF)            ;; A,R(DF)=HI=A,CLK=D
2205 .MACRO D_BSPLO(DF)            ;; B,R(DF)=LO=B,CLK=D
2206 .MACRO D_BSPHI(DF)            ;; B,R(DF)=HI=B,CLK=D
2207 .MACRO D_ASPLN(SF)            ;; A,R(SF)=LO=A,CLK=D
2208 .MACRO D_ASPHI(SF)            ;; A,R(SF)=HI=A,CLK=D
2209 .MACRO D_BSPLO(SF)            ;; B,R(SF)=LO=B,CLK=D
2210 .MACRO D_BSPHI(SF)            ;; B,R(SF)=HI=B,CLK=D
2211
2212 .MACRO D_CSPD(14)=AND=ASPHI(XX) ;; A=AND=B,CSPD(D14),ASPHI(0XX),CLK=D
2213 .MACRO D_CSPD(15)=AND=ASPHI(XX) ;; A=AND=B,CSPD(D15),ASPHI(0XX),CLK=D
2214

```

```

2215 .MACRO SR_ASPHI(17)=AND=007700 ;; A=AND=B,ASPHI(R17),CSPB(B17),CLK=SR
2216 .MACRO D_SR=IOR=170000          ;; A=IOR=B,SR,CSPB(B16),CLK=D
2217 .MACRO SR_ASPHI(17)=AND=000077 ;; A=AND=B,ASPHI(R17),CSPB(B15),CLK=SR
2218 .MACRO D_SR=IOR=000100          ;; A=IOR=B,SR,CSPB(B14),CLK=D
2219
2220 .MACRO D_ASPLN(17)=AND=CSPD(XX)  ;; A=AND=B,ASPLO(R17),CSPD(0XX),CLK=D
2221 .MACRO D_ASPLN(00)=IOR=CSPD(XX) ;; A=IOR=B,ASPHI(R00),CSPD(0XX),CLK=D
2222 .MACRO D_ASPLN(00)=IOR=CSPB(XX) ;; A=IOR=B,ASPHI(R00),CSPB(0XX),CLK=D
2223
2224 .MACRO D_SR                  ;; A,SR,CLK=D
2225 .MACRO D_ALL=ONES           ;; A,C177777=A,CLK=D
2226 .MACRO D_D=PLUS=1           ;; A=PLUS=B,D=DIRECT,C000001=B,CLK=D
2227 .MACRO D_JUNK                ;; ZERO,CLK=D
2228 .MACRO D_TWO                 ;; A=PLUS=B,C000001=A,C000001=B,CLK=D
2229
2230
2231
2232 1.PAGE=====
2233
2234 .TOC *      SR <- DATA, AT P2 T OR P3 T
2235
2236 14.R.1 THE PARTICULAR FUNCTION SELECTED REQUIRES THE RESIDUAL
2237 1 CONTROL REGISTER ("RES-REG") TO HAVE THE APPROPRIATE
2238 1 FUNCTION SETUP FOR THE SR OPERATION.
2239 1
2240 1 POSSIBLE FUNCTIONS: LOAD, LEFT, RIGHT, NOP
2241
2242 .MACRO SR_ASPHI(XX)            ;; A,ASPHI(0XX),CLK=SR
2243 .MACRO SR_NOT=ASPHI(XX)        ;; NOT=A,ASPHI(0XX),CLK=SR
2244 .MACRO SR_CSPB(XX)             ;; B,CSPB(0XX),CLK=SR
2245 .MACRO SR_CSPD(XX)             ;; B,CSPD(0XX),CLK=SR
2246 .MACRO SR_NOT=BSPHI(XX)        ;; A=AND=NOT=B,BSPHI(0XX),C177777=A,CLK=SR
2247 .MACRO SR_ASPHI(XX)            ;; B,BSPHI(0XX),CLK=SR
2248 .MACRO SR_SR=PLUS=1           ;; A=PLUS=B,C000001=B,SR,CLK=SR
2249 .MACRO SR_ALL=ONES            ;; A,C177777=A,CLK=SR
2250 .MACRO SR_NOT=CSPB(XX)         ;; A=AND=NOT=B,C177777=A,CSPB(0XX),CLK=SR
2251 .MACRO SR_NOT=CSPD(XX)         ;; A=AND=NOT=B,C177777=A,CSPD(0XX),CLK=SR
2252 .MACRO SR_SR=RIGHT=1          ;; D=DIRECT(BMUX),CLK=SR
2253 .MACRO SR_SR=LEFT=1           ;; CLK=SR
2254 .MACRO SR_JUNK                 ;; ZERO,CLK=SR
2255 .MACRO SR_D                    ;; A,D=DIRECT,CLK=SR
2256 .MACRO SR_ASPLN(DF)           ;; A,R(DF)=LO=A,CLK=SR
2257 .MACRO SR_ASPHI(DF)            ;; A,R(DF)=HI=A,CLK=SR
2258 .MACRO SR_BSPLO(DF)           ;; B,R(DF)=LO=B,CLK=SR
2259 .MACRO SR_BSPHI(DF)           ;; B,R(DF)=HI=B,CLK=SR
2260 .MACRO SR_ASPLN(SF)           ;; A,R(SF)=LO=A,CLK=SR
2261
2262 .MACRO SR_ASPLN(SF)           ;; A,R(SF)=HI=A,CLK=SR
2263 .MACRO SR_BSPHI(SF)           ;; B,R(SF)=LO=B,CLK=SR
2264 .MACRO SR_BSPHI(SF)           ;; B,R(SF)=HI=B,CLK=SR
2265
2266
2267
2268

```

```
2269 .TOC * RES-REG OPERATION MACROS
2270
2271 .MACRO RES_CSPD(XX)          ;; CSPD(XX),LOAD-RES
2272 .MACRO RES_CSPB(XX)          ;; CSPB(XX),LOAD-RES
2273
2274
2275
2276
2277
2278 .TOC * BASE MACHINE COUNTER
2279
2280 .MACRO COUNTER_CSPD(XX)      ;; LOAD-COUNTER,CSPD(XX)
2281 .MACRO COUNTER_BSPHI(XX)    ;; LOAD-COUNTER,BSPHI(XX)
2282
2283
2284
2285
2286 .TOC * ENARLF ON BUS-A/B ONLY
2287
2288 .MACRO BUS-A_ASPLO(SF)       ;; R(SF)-LO-A
2289 .MACRO BUS-A_ASPLO(DF)       ;; R(DF)-LO-A
2290 .MACRO BUS-A_ASPHI(SF)       ;; R(SF)-HI-A
2291 .MACRO BUS-A_ASPHI(DF)       ;; R(DF)-HI-A
2292 .MACRO BUS-A                ;; NULL
2293 .MACRO BUS-A_ASPLO(XX)       ;; ASPLO(XX)
2294 .MACRO BUS-A_ASPHI(XX)       ;; ASPHI(XX)
2295 .MACRO BUS-A_SR              ;; SR
2296 .MACRO BUS-A_FLTPT           ;; FLTPT
2297
2298 .MACRO BUS-B_BSPLO(SF)       ;; R(SF)-LO-B
2299 .MACRO BUS-B_BSPLO(DF)       ;; R(DF)-LO-B
2300 .MACRO BUS-B_BSPHI(SF)       ;; R(SF)-HI-B
2301 .MACRO BUS-B_BSPHI(DF)       ;; R(DF)-HI-B
2302 .MACRO BUS-B                ;; NULL
2303 .MACRO BUS-B_BSPLO(XX)       ;; BSPLO(XX)
2304 .MACRO BUS-B_BSPHI(XX)       ;; BSPHI(XX)
2305 .MACRO BUS-B_CSPD(XX)        ;; CSPD(XX)
2306 .MACRO BUS-B_CSPB(XX)        ;; CSPB(XX)
2307
2308
2309
2310
2311
2312 .TOC * LOADING BA REGISTER
2313 ;LOADED AT P1-T ONLY, FROM BUS-B<01:00>#BUS-A<15:00> -> BA<17:00>
2314
2315 .MACRO BA_BSPLO(XX)          ;; CLK-BA,BSPLO(XX)
2316 .MACRO BA_BSPHI(XX)          ;; CLK-BA,BSPHI(XX)
2317 .MACRO BA_SR                 ;; CLK-BA,SR
2318 .MACRO BA_ASPLO(XX)          ;; CLK-BA,ASPLO(XX)
2319 .MACRO BA_ASPHI(XX)          ;; CLK-BA,ASPHI(XX)
2320
2321
2322
```

```
2323
2324
2325 .TOC * D AND SR TOGETHER
2326
2327 .MACRO SR#D_SR-PLUS-CSPD(XX) ;; A-PLUS-B,SR,CSPD(XX),CLK-D,CLK-SR
2328
2329
2330
2331 ;.PAGE=====
2332
2333 .TOC * UCON FUNCTIONS
2334
2335
2336
2337 .TOC * PROCESSOR UCON FUNCTIONS
2338
2339 ;PREVIOUSLY SFT UP (UCON-PROC, SET-UCON-CONTROL, EN "FUNCTION")
2340 .MACRO IR_EMIT                ;; UCON-OPERATION
2341 .MACRO PS(15-12)_D(15-13)-[I] ;; UCON-OPERATION
2342 .MACRO FLAG(8-0)_D(15-8)      ;; UCON-OPERATION
2343 .MACRO FPS(7-4)_D(7-4)-[I]    ;; UCON-OPERATION
2344 .MACRO PS(7-4)_D(7-4)-[I]     ;; UCON-OPERATION
2345 .MACRO PS(3-0)_D(3-0)-[I]     ;; UCON-OPERATION
2346 .MACRO PS_D                  ;; UCON-OPERATION
2347 .MACRO UBRKAK_BUSDIN(11-00)    ;; UCON-OPERATION
2348
2349 ;SETUP UCON AND EXECUTE IN 1 MICROWORD:
2350 .MACRO PS(15-12)_D(15-13)-[I] ;; UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-PS(15-12)
2351 .MACRO FLAG(8-0)_D(15-8)-[I]  ;; UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-FLAG(8-0)
2352 .MACRO FPS(7-4)_D(7-4)-[I]    ;; UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-FPS(7-4)
2353 .MACRO PS(7-4)_D(7-4)-[I]     ;; UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-PS(7-4)
2354 .MACRO PS(3-0)_D(3-0)-[I]     ;; UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,EN-CLK-PS(3-0)
2355 .MACRO PS_D-[I]               ;; UCON-PROC,SET-UCON-CONTROL,UCON-OPERATION,
2356                                     EN-CLK-PS(15-12),EN-CLK-PS(7-4),EN-CLK-PS(3-0)
2357 .MACRO BUSDIN_CUA-[I]          ;; UCON-PROC,SET-UCON-CONTROL,BUSDIN_CUA[14-03]
2358 .MACRO BUSDIN_FLAGS#FPS-[I]   ;; UCON-PROC,SET-UCON-CONTROL,BUSDIN_FLAG(8-0)#FPS(7-0)
2359 .MACRO BUSDIN_PS-[I]           ;; UCON-PROC,SET-UCON-CONTROL,BUSDIN_PS[15-00]
2360 .MACRO BUSDIN_FMIT-[I]         ;; UCON-PROC,SET-UCON-CONTROL,BUSDIN_EMIT[15-00]
2361
2362
2363
2364 .TOC * CACHE/KT UCON FUNCTIONS
2365
2366 ;SETUP, EXECUTE IN 1 MICROWORD
2367 .MACRO KT-NO-RELOCATE-[I]      ;; UCON-CACHE-KT,SET-UCON-CONTROL,EN-KT-NO-RELOCATE
2368 .MACRO BUSDIN_BUS-INTERNAL-ADDR-[I] ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_BUS-INTERNAL-ADDR[15-00]
2369
2370 .MACRO BUSDIN_CPU-INTERNAL-ADDR-[I] ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_CPU-INTERNAL-ADDR[15-00]
2371 .MACRO BUSDIN_MMR2-[I]          ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_MMR2[15-00]
2372 .MACRO BUSDIN_CACHE-STATUS-[I] ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_CACHE-STATUS[15-00]
2373 .MACRO BUSDIN_SLR#CCR-[I]       ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-SLR#CCR
2374 .MACRO BUSDIN_MMRO-[I]          ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-MMRO
2375 .MACRO BUSDIN_PDR-[I]           ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-PDR
2376 .MACRO BUSDIN_PAR-[I]           ;; UCON-CACHE-KT,SET-UCON-CONTROL,BUSDIN_KT-SEL,KT-SEL-PAR
2377 .MACRO SLR[15-08]_[I]           ;; UCON-CACHE-KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL-SLR#CCR,KT-WRITE-HIGH
```

```

2377 .MACRO CCP[07-02]_D[07-02]-[I]      ;; UCON-CACHE=KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=BLR&CCR,KT-WRITE=LOW
2378 .MACRO MMRO_D-[I]                    ;; UCON-CACHE=FT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=MMRO,KT-WRITE
2379 .MACRO MMRO[00]_D[00]-[I]            ;; UCON-CACHE=KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=MMRO,KT-WRITE=LOW
2380 .MACRO MMRO[15-01]_D[15-01]-[I]      ;; UCON-CACHE=FT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=MMRO,KT-WRITE-HIGH
2381 .MACRO PDR_D-[I]                      ;; UCON-CACHE=KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=PDR,KT-WRITE
2382 .MACRO PDR[03-01]_D[03-01]-[I]       ;; UCON-CACHE=KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=PDR,KT-WRITE=LOW
2383 .MACRO PDR[14-08]_D[14-08]-[I]       ;; UCON-CACHE=KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=PDR,KT-WRITE-HIGH
2384 .MACRO PAR_D-[I]                      ;; UCON-CACHE=KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=PAR,KT-WRITE
2385 .MACRO PAR[07-00]_D[07-00]-[I]       ;; UCON-CACHE=FT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=PAR,KT-WRITE=LOW
2386 .MACRO PAR[11-08]_D[11-08]-[I]      ;; UCON-CACHE=KT,SET-UCON-CONTROL,UCON-OPERATION,KT-SEL=PAR,KT-WRITE-HIGH
2387
2388
2389
2390
2391 .TOC *      I-O UCON FUNCTIONS
2392
2393 IN_R_1 SETUP IN 1 MICROWORD
2394 .MACRO BUSDIN_JAM-[I]                  ;; UCON-I-O,EN-STATUS-MUX,SET-UCON-CONTROL,BUSDIN_JAM[15-00]
2395 .MACRO BUSDIN_SERVICE-[I]             ;; UCON-I-O,EN-STATUS-MUX,SET-UCON-CONTROL,BUSDIN_SERVICE[15-00]
2396 .MACRO BUSDIN_PBA-[I]                 ;; UCON-I-O,EN-STATUS-MUX,SET-UCON-CONTROL,BUSDIN_PBA[15-00]
2397 .MACRO BC_FCN_D-[I]                   ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-BC_FCN_D
2398 .MACRO START_DELAY-[I]                ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-START_DELAY
2399 .MACRO CLR_JAM_ERRORS-[I]              ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-CLR_JAM_ERRORS
2400 .MACRO CLR_NPR_TIMEOUT-[I]            ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-CLR_NPR_TIMEOUT
2401 .MACRO CLR_PWP_FAIL-[I]                ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-CLR_PWP_FAIL
2402 .MACRO CLR_YELLOW_ZONE-[I]            ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-CLR_YELLOW_ZONE
2403 .MACRO ALLOW_BG[1]H-[I]                ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-ALLOW_BG[1]H
2404 .MACRO BUS_INIT_UCON-[I]               ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-BUS_INIT_UCON
2405
2406 .TOC *      DCS UCON FUNCTIONS
2407
2408 ISETUP IN 1 MICROWORD:
2409 .MACRO BUSDIN_TNVA-[I]                 ;; UCON-DCS,SET-UCON-CONTROL,BUSDIN_TNVA[11-00]
2410 .MACRO BUSDIN_ERROR_CODE-[I]           ;; UCON-DCS,SET-UCON-CONTROL,BUSDIN_ERROR_CODE[11-00]
2411
2412 .TOC *      CONSOLE UCON FUNCTIONS
2413
2414 ISETS UP AND PERFORMS INDICATED OPERATION IN 1 MICROWORD
2415 .MACRO CONSOLE_NOP                     ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-CONSL_NOP
2416 .MACRO CLR_CONSOLE_COUNTER             ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-CLR_COUNTER
2417 .MACRO INCREMENT_CONSOLE_COUNTER       ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-INCR_COUNTER
2418 .MACRO CLEAR_CONSOLE_SERVICE           ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-CLR_CONSL_SRVC
2419 .MACRO STROBE_CONSOLE_DISPLAY          ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-STRB_DISP
2420 .MACRO CLEAR_CONSOLE_LED               ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-CLR_CONSL
2421 .MACRO SET_CONSOLE_LED                 ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-SET_CONSL
2422 .MACRO SET_CONSOLE_DP_LEDS             ;; UCON-I-O,EN-CONSOLE-COMMAND,SET-UCON-CONTROL,UCON-OPERATION,EN-SET_DP
2423 .MACRO BUSDIN_CONSOLE-[I]              ;; UCON-I-O,EN-STATUS-MUX,SET-UCON-CONTROL,BUSDIN_CONSOLE[06-00]
2424
2425
2426
2427
2428 .TOC *      DRUF UCON FUNCTIONS
2429
2430 IPREVIOUSLY SETUP UCON-I-O, EN LOAD DRUF

```

```

2431 .MACRO DRUF_D                        ;; UCON-OPERATION
2432
2433 ISETUP AND EXECUTE IN 1 MICROWORD:
2434 .MACRO DRUF_D-[I]                     ;; UCON-I-O,SET-UCON-CONTROL,UCON-OPERATION,EN-LOAD-DRUF[15-00]
2435
2436
2437 .TOC *      MULTIPLE UCON FUNCTIONS
2438
2439 ITHSE ARE FUNCTIONS OF MORE THAN 1 UCON ENABLED SIMULTANEOUSLY
2440
2441 IPREVIOUSLY SETUP:
2442 .MACRO TR_DBUF                         ;; UCON-OPERATION
2443
2444 ISETUP AND EXECUTE IN 1 MICROWORD:
2445 .MACRO TR_DBUF-[I]                     ;; UCON-PROC,UCON-I-O,SET-UCON-CONTROL,
2446                                     UCON-OPERATION,EN-CLK-IR[15-00],BUSDIN_DBUF[15-00]
2447
2448
2449
2450
2451 I.PAGE=====
2452 .TOC *      SPECIFIC MACROS FOR PREFETCH/OVERLAP/SP-INHIBIT TESTS
2453
2454 .MACRO CSPD[17]_020010                 ;; ENIT/020010,CSPD[17]_EMIT
2455 .MACRO A_BSPLO[OVERLAP]_D              ;; BSPLO(R10),WR(A,B,L,A)
2456 .MACRO A_BSPHI[PATTERN]_D              ;; BSPHI(R17),WR(A,B,H,A)
2457 .MACRO A_BSPHI[PREFETCH]_D             ;; BSPHI(R10),WR(A,B,H,A)
2458 .MACRO A_BSPLO[OVERLAP]_D              ;; BSPLO(R10),WR(A,L,A)
2459 .MACRO A_BSPLO[OVERLAP]_D              ;; BSPLO(R10),WR(B,L,B)
2460 .MACRO A_BSPHI[PREFETCH]_D             ;; BSPHI(R10),WR(A,H,A)
2461
2462 .MACRO D_ASPLO[OVERLAP]_PLUS-1          ;; A-PLUS-B,BSPLO(R10),C000001-B,CLK-D
2463 .MACRO D_BSPLO[OVERLAP]_PLUS-1          ;; A-PLUS-B,BSPLO(R10),C000001-A,CLK-D
2464 .MACRO D_ASPHI[PREFETCH]_PLUS-1        ;; A-PLUS-B,BSPHI(R10),C000001-B,CLK-D
2465 .MACRO D_ASPHI[PATTERN]_PLUS-020010-PLUS-1 ;; A-PLUS-B-PLUS-1,BSPHI(R17),CSPB(B17),CLK-D
2466 .MACRO D_ASPHI[PATTERN]_AND-NOT-020010 ;; A-AND-NOT-B,BSPHI(R17),CSPB(B17),CLK-D
2467
2468 .MACRO D_BSPHI[PREFETCH]                ;; B,BSPHI(R10),CLK-D
2469 .MACRO D_ASPLO[OVERLAP]_MINUS-CSPB[EXPEC] ;; A-MINUS-B,BSPLO(R10),CSPB(B17),CLK-D
2470 .MACRO D_ASPHI[PREFETCH]_MINUS-CSPB[EXPEC] ;; A-MINUS-B,BSPHI(R10),CSPB(B17),CLK-D
2471 .MACRO D_ASPLO[OVERLAP]_MINUS-BSPLO[OVERLAP] ;; A-MINUS-B,BSPLO(R10),BSPLO(R10),CLK-D
2472
2473 .MACRO CSPD[EXPEC]_EMIT                 ;; CSPD[17]_EMIT
2474
2475
2476
2477 I.PAGE=====
2478 .TOC *      SPECIFIC MACROS FOR BYTE/BYTE CONSTANT/D=ZERO TESTS
2479
2480
2481 .MACRO D_ASPLO[DNONZERO]_PLUS-1         ;; A-PLUS-B,BSPLO(R11),C000001-B,CLK-D
2482 .MACRO D_BSPLO[DZERO]_PLUS-1           ;; A-PLUS-B,C000001-A,BSPLO(R11),CLK-D
2483 .MACRO D_ASPHI[IR-DATA]_PLUS-2         ;; A-PLUS-B-PLUS-1,C000001-A,BSPLO(R10),CLK-D
2484 .MACRO D_ASPHI[WORD]_PLUS-1             ;; A-PLUS-B,C000001-A,BSPHI(R10),CLK-D

```

```

2485 .MACRO D_BSPLO(CTR-DATA)          ;; B,BSPLO(R10),CLK-D
2486 .MACRO D_BSPLO(DZERO)              ;; B,BSPLO(R11),CLK-D
2487 .MACRO D_BSP-MINUS-BSPHI(WORD)     ;; A-MINUS-B,BR,BSPHI(R10),CLK-D,P3-T
2488 .MACRO D_BSPLO(DNONZERO)-MINUS-CSPD[17] ;; A-MINUS-B,BSPLO(R11),CSPD(D17),CLK-D,P3-T
2489 .MACRO D_BSPHI(BYTE-FIRST)-MINUS-CSPD[17] ;; A-MINUS-B,BSPHI(R10),CSPD(D17),CLK-D,P3-T
2490 .MACRO D_BSPLO(BYTE-SECOND)-MINUS-CSPD[17] ;; A-MINUS-B,BSPLO(R10),CSPD(D17),CLK-D,P3-T
2491 .MACRO D_BSPHI(BYTE-FIRST)-PLUS-CSP[1-0] ;; A-PLUS-B,BSPHI(R10),CSPD(D13),CLK-D
2492 .MACRO D_BSPLO(BYTE-SECOND)-PLUS-CSP[1-0] ;; A-PLUS-B,BSPLO(R10),CSPD(D13),CLK-D
2493
2494
2495
2496 !.PAGE=====
2497
2498 .TOC *      SUBROUTINE CALL MACROS
2499
2500 .MACRO CALL(DISPLAY)                ;; GOTO-PAGE(7),J/DISPLAY
2501
2502 .MACRO CALL(DINTOIR)                ;; GOTO-PAGE(7),J/DINTOIR
2503 .MACRO CALL(DINTOIR-5)              ;; GOTO-PAGE(7),J/DINTOIRS
2504 .MACRO CALL(SRINTOIR)               ;; GOTO-PAGE(7),J/SRINTOIR
2505 .MACRO CALL(SRINTOIR-5)            ;; GOTO-PAGE(7),J/SRINTOIRS
2506
2507 .MACRO CALL(FLAGFPSTOD)             ;; GOTO-PAGE(7),J/FLAGFPSTOD
2508 .MACRO CALL(PSTOD)                 ;; GOTO-PAGE(7),J/PSTOD
2509 .MACRO CALL(CUATOD)                ;; GOTO-PAGE(7),J/CUATOD
2510 .MACRO CALL(CLRJAMTOD)             ;; GOTO-PAGE(7),J/CLRJAMTOD
2511 .MACRO CALL(ODDJAMTOD)             ;; GOTO-PAGE(7),J/ODDJAMTOD
2512 .MACRO CALL(JAMTOD)                ;; GOTO-PAGE(7),J/JAMTOD
2513 .MACRO CALL(CLRSERVICETOD)         ;; GOTO-PAGE(7),J/CLRSERVICETOD
2514 .MACRO CALL(DATISERVICETOD)       ;; GOTO-PAGE(7),J/DATISERVICETOD
2515 .MACRO CALL(DATOSERVICETOD)       ;; GOTO-PAGE(7),J/DATOSERVICETOD
2516 .MACRO CALL(CJESERVICETOD)        ;; GOTO-PAGE(7),J/CJESERVICETOD
2517 .MACRO CALL(SERVICETOD)           ;; GOTO-PAGE(7),J/SERVICETOD
2518 .MACRO CALL(PBATOD)               ;; GOTO-PAGE(7),J/PBATOD
2519 .MACRO CALL(PSSSEQLOD)             ;; GOTO-PAGE(7),J/PSSSEQLOD
2520 .MACRO CALL(FLAGFPSSSEQLOD)       ;; GOTO-PAGE(7),J/FLAGFPSSSEQLOD
2521 .MACRO CALL(FLAGFLOD)             ;; GOTO-PAGE(7),J/FLAGFLOD
2522 .MACRO CALL(GETPROC DAT)           ;; GOTO-PAGE(7),J/GETPROC DAT
2523
2524 .MACRO CALL(CLEAR-I-O-A)           ;; GOTO-PAGE(7),J/CLEAR-I-O-A
2525 .MACRO CALL(CLEAR-I-O-B)           ;; GOTO-PAGE(7),J/CLEAR-I-O-B
2526
2527 .MACRO CALL(BUSDINXOR125252)       ;; GOTO-PAGE(7),J/BDX12
2528 .MACRO CALL(BUSDINXOR052525)       ;; GOTO-PAGE(7),J/BDX05
2529 .MACRO CALL(CBP17XOR125252)       ;; GOTO-PAGE(7),J/C17X12
2530 .MACRO CALL(CBP17XOR052525)       ;; GOTO-PAGE(7),J/C17X05
2531
2532 .MACRO CALL(D15-12)                ;; GOTO-PAGE(7),J/D[15-12]
2533 .MACRO CALL(D11-06)                ;; GOTO-PAGE(7),J/D[11-06]
2534 .MACRO CALL(D05-00)                ;; GOTO-PAGE(7),J/D[05-00]
2535 .MACRO CALL(DZERO)                ;; GOTO-PAGE(7),J/DZERO
2536
2537 .MACRO CALL(ZEROSF04DF02)          ;; GOTO-PAGE(7),J/ZEROSF04DF02
2538 .MACRO CALL(ZEROSF02DF04)          ;; GOTO-PAGE(7),J/ZEROSF02DF04

```

```

2539 .MACRO CALL(ZEROSFDF)             ;; GOTO-PAGE(7),J/ZEROSFDF
2540 .MACRO CALL(ZERODF)               ;; GOTO-PAGE(7),J/ZERODF
2541
2542 .MACRO CALL(SFDFTO8R)             ;; GOTO-PAGE(8),J/SFDFTO8R
2543
2544 .MACRO CALL(ALUCARRY1)             ;; GOTO-PAGE(7),J/ALUCARRY1
2545 .MACRO CALL(ALUCARRY2)            ;; GOTO-PAGE(7),J/ALUCARRY2
2546
2547 .MACRO CALL(LOADFP8CC)             ;; GOTO-PAGE(7),J/LOADFP8CC
2548 .MACRO XPP-TO-RM(LOADNZW4)        ;; GOTO-PAGE(4),J/LOADNZW4
2549
2550 .MACRO CALL(SETUPP8CC+DC)          ;; GOTO-PAGE(7),J/SETUPP8CC+DC
2551 .MACRO CALL(P8CCTO8R3-0)          ;; GOTO-PAGE(7),J/P8CCTO8R3-0
2552
2553 .MACRO CALL(CSP16XOR8RTOIR-5)     ;; GOTO-PAGE(7),J/CSP16XOR8RTOIRS
2554 .MACRO CALL(CSP16XORFLTTOIR-5)    ;; GOTO-PAGE(7),J/CSP16XORFLTTOIRS
2555
2556 .MACRO CALL(MF88-TEST)            ;; GOTO-PAGE(6),J/MF8801
2557
2558 .MACRO CALL(KTSRCDST)              ;; GOTO-PAGE(7),J/KTSRCDST01
2559 .MACRO CALL(KTSRCDSTSP)           ;; GOTO-PAGE(7),J/KTSRCDST08
2560 .MACRO CALL(KTSRCD8SP)            ;; GOTO-PAGE(7),J/KTSRCDST07
2561
2562 .MACRO CALL(COUNT-TEST)           ;; GOTO-PAGE(4),J/COUNTER01
2563
2564
2565
2566 !.PAGE=====
2567
2568 .TOC *      JAM UPP LOG MACROS
2569
2570 !MACROS CONCERNED WITH CSP LOG AFTER UNEXPECTED JAMUPP
2571 !MACROS REQUIRE APPROPRIATE REGISTER ENABLED ON BUSDIN
2572
2573 .MACRO CSPD[00]-LOG-CUA            ;; CSPD(D00),WR-CSP
2574 .MACRO CSPD[01]-LOG-SERVICE       ;; CSPD(D01),WR-CSP
2575 .MACRO CSPD[02]-LOG-JAM           ;; CSPD(D02),WR-CSP
2576
2577
2578
2579 !**** END OF MACRO DEFINITIONS ****
2580
2581
2582
2583 !.PAGE=====
2584
2585 .TnC * - - - MICRODIAGNOSTIC CODE - - -
2586
2587 .CODE
2588
2589
2590
2591
2592

```

```

2593
2594
2595 1.PAGE=====
2596
2597 1*** VERSION /V10140/ ***
2598
2599 1***** MICROMDIAGNOSTIC INITIAL STARTUP LOCATION *****
2600
2601
2602 .TDC * TEST001-007: NUA SEQUENCING
2603
2604 1*****
2605 1*
2606 1* TESTS: 001 - 007 UWORDS: 010 + 000
2607 1*
2608 1* FUNCTIONS: TESTS 001 - 007 TEST THE NUA SEQUENCING LOGIC.
2609 1* PATTERNS ARE RUN THRU THE NUA LOGIC ESTABLISHING
2610 1* THAT ALL BITS CAN BE SET AND CLEARED, AND THAT THE
2611 1* STATE (SET, CLEARED) OF NO ONE BIT AFFECTS THE
2612 1* ABILITY TO SET/CLEAR ANY OTHER BIT. THE PAGE
2613 1* CHANGING FUNCTIONS OF BUTA(SUBR-1) AND BUTA(SUBR-5)
2614 1* ARE ALSO TESTED FOR PAGES 4, 5, 6, & 7. NOTE
2615 1* THAT THE RETURN ADDRESS SIMULTANEOUSLY LOADED
2616 1* IS NOT CHECKED FOR VALIDITY AT THIS POINT.
2617 1*
2618 1* NOTFS: TFST[N] DOES THE SETUP FOR TESTIN+1]. THE ACTUAL TEST CONSISTS
2619 1* OF BEING IN THE RIGHT PLACE (MICROWORD) AT THE RIGHT TIME.
2620 1*
2621 1*****
2622
2623
2624
2625
2626
2627
2628 1 - - - - -
2629
2630 1*** TFST 001 ***
2631 1TFST NUA LOGIC WITH PATTERN "100 000 000 000"
2632
2633 40001:
2634 TFST001:
2635 PO, LOAD=ENUA(6252),
2636 LOAD=ERROR(7EST001),
2637 DCS=CTR(C1),
2638 NFXT, PAGE(2), BUTA(SUBR-5),
2639 J/TEST002
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2999
3000

```

```

2644
2645
2646
2647
2648
2649 | - - - - -
2650
2651 |*** TEST 002 ***
2652 |TEST NUA LOGIC WITH PATTERN "010 010 101 010"
2653 |62521
2654 |TEST002:
2655 |PO, LOAD=ENUA(6631), |LOAD ENUA WITH ADDR[NEXT UWORD]
2656 |LOAD=ERROR(TEST002), |ERROR DIRECTORY KEY
2657 |DCS=CTR(C1.), |COMPARE AT NEXT UWORD
2658 |BIMP=VERIFY, |COUNT
2659 |NEXT, J/TEST003 |
(6252) DCS[1,00,1,0,0,0] BM[1110..00,11..01,10..011..001...0,0,0..0..0..0,0000...0..0000,0...11,000...110,011,001]
2660
2661
2662
2663
2664
2665
2666
2667 | - - - - -
2668
2669 |*** TEST 003 ***
2670 |TEST NUA LOGIC WITH PATTERN "010 110 011 001"
2671 |66311
2672 |TEST003:
2673 |PO, LOAD=ENUA(5525), |LOAD ENUA WITH ADDR[NEXT UWORD]
2674 |LOAD=ERROR(TEST003), |ERROR DIRECTORY KEY
2675 |DCS=CTR(C1.), |COMPARE AT NEXT UWORD
2676 |NEXT, PAGE(5), BUTA(SUBR-A), |CHANGING TO PAGE 5, VIA SUBR-A
2677 |J/TEST004 |NOTE OVERLAP: NEXTPAGE<2;0>=ENUA<2;0>
2678 (6631) DCS[1,00,1,0,0,0] BM[1110..00,10..11,01..010..101...0,0,0..0..0..0,0000...0..0000,0...11,101...101,010,101]
2679
2680
2681
2682
2683
2684
2685
2686 | - - - - -
2687
2688 |*** TEST 004 ***
2689 |TEST NUA LOGIC WITH PATTERN "101 101 010 101"
2690 |55251
2691 |TEST004:
2692 |PO, LOAD=ENUA(5146), |LOAD ENUA WITH ADDR[NEXT UWORD]
2693 |LOAD=ERROR(TEST004), |ERROR DIRECTORY KEY
2694

```

```

2695          DCS-CTR(C1.),           !COMPARE AT NEXT UWORD
2696          BUMP=VERIFY,             !COUNT
2697          NEXT, J/TEST005            |
(5525) DCS[1,00,1,0,0,1] BM[1110,,00,10,,10,01,,100,,110,,,0,0,0,,0,,0...0.0000...0..0000,0,...11,000...001,100,110]

2698
2699
2700
2701
2702
2703
2704
2705
2706      |-----|
2707
2708      !*** TEST 005 ***
2709      !TEST   NUA LOGIC    WITH PATTERN "101 001 100 110"
2710      5146!
2711      TEST005:
2712          PO,        LOAD=ENUA(4474),           !LOAD ENUA WITH ADDR[NEXT UWORD]
2713                  LOAD=ERROR(TEST005),         !ERROR DIRECTORY KEY
2714                  DCS-CTR(C1.),               !COMPARE AT NEXT UWORD
2715                  NEXT, PAGE(4), BUTA(SUBR-B), !CHANGING TO PAGE 4, VIA SUBR-B
2716                  J/TEST006                   !NOTE OVERLAP! NEXTPAGE<210>=ENUA<210>
(5146) DCS[1,00,1,0,0,0] BM[1110,,00,10,,01,00,,111,,100,,,0,0,0,,0,,0...0.0000...0..0000,0,...11,100...100,111,100]

2717
2718
2719
2720
2721
2722
2723
2724
2725      |-----|
2726
2727      !*** TEST 006 ***
2728      !TEST   NUA LOGIC    WITH PATTERN "100 100 111 100"
2729      4474!
2730      TEST006:
2731          PO,        LOAD=ENUA(4377),           !LOAD ENUA WITH ADDR[NEXT UWORD]
2732                  LOAD=ERROR(TEST006),         !ERROR DIRECTORY KEY
2733                  DCS-CTR(C1.),               !COMPARE AT NEXT UWORD
2734                  BUMP=VERIFY,             !COUNT
2735                  NEXT, J/TEST007            |
(4474) DCS[1,00,1,0,0,1] BM[1110,,00,10,,00,11,,111,,111,,,0,0,0,,0,,0...0.0000...0..0000,0,...11,000...011,111,111]

2736
2737
2738
2739
2740
2741
2742
2743
2744      |-----|

```

```

2745 1*** TEST 007 ***
2746 |TEST NUA LOGIC WITH PATTERN "100 011 111 111"
2747 4377:
2748 TEST007:
2749     PO,          LOAD=ENUA(7303),          |LOAD ENUA WITH ADDR[NEXT UWORD]
2750     LOAD=ERROR(TEST007),          |ERROR DIRECTORY KEY
2751     DCS=CTRC(1,),          |COMPARE AT NEXT UWORD
2752     NFXT,        PAGE(3), BUTA(SUBR-A),          |CHANGING TO PAGE 3 (ACTUALLY 7), VIA SUBR-A
2753     J/NEXT007          |NOTE OVERLAP: NEXTPAGE<2:0>=ENUA<2:0>
2754 (4377) DCS{1,0,0,1,0,0,0} BM{1110,00,11,10,11,000,0,11,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,101,0,011,000,011}
2755                                     |NOTE ALSO DCS MODULE FORCES NUA<11>=1, SO
2756                                     |PAGES 3/7 ARE EQUIVALENT REFERENCES.
2757
2758
2759
2760
2761
2762 |-----
2763
2764 1*** TEST 007-1/2 ***
2765 |TEST NUA LOGIC WITH PATTERN "011 011 000 011"
2766 7303:
2767 NEXT007:
2768     NEXT,        PAGE(4), BUTA(SUBR-B),          |CHANGING TO PAGE 4, VIA SUBR B
2769     J/TEST010          |OFF TO NEXT TEST
2770 (7303) DCS{0,0,0,0,0,0,0} BM{0000,00,00,00,00,0,100,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,100,0,111,111,101}
2771
2772
2773 1.PAGE=====
2774
2775 .TNC * TEST010-011: MICROSUBROUTINE OPERATION
2776
2777 1=====
2778 1*
2779 1* TESTS: 010 - 011                                UWORDS: 007 + 002
2780 1*
2781 1* FUNCTIONS:  THESE TWO TESTS DETERMINE THAT THE RETURN REGISTER AND
2782 1* ASSOCIATED DECODE, MUX, AND ENABLING LOGIC IS ABLE TO
2783 1* LOAD THE 12-BIT NUA RETURN REGISTER FROM THE EMIT
2784 1* FIELD, AND THEN ENTER THE REGISTER CONTENTS ONTO THE
2785 1* UWORD ADDRESS BUS [WHEN ENABLED BY A BUTA[RETURN]] IN
2786 1* TIME TO FETCH THE NEXT MICROWORD. TWO ALTERNATING
2787 1* BIT TESTS ARE USED TO CHECK THAT EACH BIT CAN BE SET
2788 1* AND CLEARED, INDEPENDENT OF ADJACENT BITS.
2789
2790 1*
2791 1=====
2792
2793
2794

```

```

2795
2796
2797 | -----
2798
2799 |*** TEST 010 ***
2800 |TEST RETURN LOGIC WITH PATTERN "1010 1010 0101" (5245)
2801 4775:
2802 TEST010:
2803     PO,          LOAD=ENUA(NEXT010),          |LOAD ENUA WITH EXPECTED RETURN ADDRESS
2804     NEXT,        LOAD=ERROR(TEST010),          |ERROR DIRECTORY KEY
2805     NEXT,        DCS=CTR(CA.),                |COMPARE ENUA;TNUA IN 4. MICROWORDS
2806     NEXT,        J/LOAD010                    |DO LOAD
(4775) DCS[1,0,0,1,0,0,0] BM[1011,00,10,10,10,100,101,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,000,000,000]
2807
2808 4100: |(FREE)
2809 LOAD010:
2810     SETUP,      RETURN/NEXT010,              |SET RETURN ADDRESS FROM EMIT
2811     NEXT,       PAGE(7),                     |"SUBR" IS ON PAGE 7
2812     NEXT,       BUTA(SUBR=B),                 |WITH B VERSION
2813     NEXT,       J/SUBR010                     |"SUBR" DISP. ON PAGE
(4100) DCS[0,0,0,0,0,0,0] BM[0101,00,01,01,00,101,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,000,000,000,000]
2814
2815 7000: |(FREE)
2816 SUBR010:
2817     SETUP,      RETURN/ERROR010,              |INQISE BITS IN EMIT-RETURN FIELD
2818     P3,         BUTA(CLR=FLAG-RES=UCON),       |NOISE IN UBF FIELD
2819     NEXT,       J/ZTARGET777                  |BEFORE DO BUTA[RETURN]
(7000) DCS[0,0,0,0,0,0,0] BM[0110,00,10,11,00,011,000,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,010,000,000,000,000]
2820
2821 | REMAINDER OF SUBROUTINE IS AT ZTARGET777 [LOCATION 7777(0)], WHICH IMMEDIATELY DOES A BUTA[RETURN]
2822 | TO THE ADDRESS IN THE RETURN REGISTER, WHICH SHOULD BE THE VALUE LOADED IN WORD
2823 | LOAD010. THE RETURN REGISTER WAS SET TO POINT TO THE START OF THE NEXT TEST
2824 | [NEXT010], AT WHICH POINT THE ENUA;TNUA COMPARE IS SET TO TAKE PLACE.
2825
2826 6543:
2827 ERROR010:
2828     PO,         BUMP=VERIFY,                  |COUNT
2829     NEXT,       PAGE(4),                      |IF WE END UP HERE, THE NOISE BITS
2830     NEXT,       J/TEST010                     |WERE LOADED INTO THE RETURN REGISTER INSTEAD
(6543) DCS[0,0,0,0,0,0,1] BM[0000,00,00,00,00,000,100,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,000,000,000,000]
2831
2832 5245:
2833 NEXT010:
2834     PO,         BUMP=VERIFY,                  |COUNT
2835     NEXT,       J/TEST011                     |RETURNED OK, COMPARE ENUA;TNUA DONE HERE
(5245) DCS[0,0,0,0,0,0,1] BM[0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,000,000,000,000]
2836
2837
2838
2839
2840
2841

```

```

2847 1
2848
2844 1*** TEST 011 ***
2845 1TEST RETURN LOGIC WITH PATTERN "1101 0101 1010" (6532)
2846 5777:
2847 TEST011:
2848     PO,          LOAD=ENUA(INITIALIZE01),          ILOAD ENUA WITH EXPECTED RETURN ADDRESS
2849     LOAD=ERROR(TEST011),          IERROR DIRECTORY KEY
2850     OCS=CTR(C3.),          ICOMPARE ENUA;TNUA IN 3, MICROWORDS
2851     NEXT,          J/LOAD011          IGO LOAD
(4777) DCS(1,00,1,0,0,0) BM(1100,00,11,01,01,011,010,00,0,0,0,0,0,0,0000,00,0000,0,0,11,000,000,000,001)

2852 4001: 1(FREF)
2853 LOAD011:
2854     SFTUP,          RETURN/INTYIALIZE01,          ISET RETURN ADDRESS IN EMIT
2855     NEXT,          PAGE(7),          I"SUBR" IS ON PAGE 7
2856     BUTA(SUBR=8),          IVIA 8 VERSION
2857     I/RESETUCOMP          I"SUBR" DISP. ON PAGE
(5001) DCS(0,00,0,0,0,0) BM(0110,00,10,10,11,010,111,00,0,0,0,0,0,0,0000,00,0000,0,0,11,100,010,111,001)

2859 1
2860 1 "ONE-WORD SUBROUTINE" IS AT LABEL "RESETUCOMP", WHICH IMMEDIATELY DOES A BUTA(RETURN)
2861 1 TO THE ADDRESS IN THE RETURN REGISTER, WHICH SHOULD BE THE VALUE LOADED AT SYMBOLIC LABEL
2862 1 "LOAD011". THE RETURN REGISTER WAS SET TO POINT TO THE INITIALIZATION CODE ON THE NEXT PAGE;
2863 1 "INITIALIZE01", AT WHICH POINT THE ENUA;TNUA COMPARE IS SET TO TAKE PLACE.
2864 1 THIS SUBROUTINE ALSO "ATTEMPTS" TO SET THE UCON CONTROL TO: "PROC", "BUSDIN_EMIT", AND
2865 1 "EN-CLK-IP", FOR USE IN THE FOLLOWING INITIALIZATION ROUTINE.
2866
2867
2868
2869 1 * * * * *
2870
2871 1 *** BUTA(RETURN) ERROR LOOP WORDS:
2872 1
2873 1 IF AT ANY TIME, A "BUTA(RETURN)" FAILS, ONE OF THE FOLLOWING MICROWORDS WILL BE
2874 1 CONTINUOUSLY LOOPEED UPON. THIS IS DUE TO THE FACT THAT FOR EACH MICROWORD THAT
2875 1 CONTAINS THIS "BUTA(RETURN)", THE UPF FIELD (WHICH WOULD NORMALLY NOT BE USED IN THIS
2876 1 MICROWORD EXECUTION), CONTAINS THE VALUE "J/BUTERRORS", WHERE NUMBER (8) IS THE
2877 1 CURRENT MICROWORD'S PAGE NUMBER (4-7).
2878
2879 4376:
2880 BUTERROR4:
2881     PO,          DCS=CTR(C0.),          IFORCE AN ERROR, ERROR-CODE=LAST LOADED
2882     NEXT,          J/BUTERROR4          IHANG UP
(4376) DCS(0,00,1,0,0,0) BM(1111,00,00,00,00,000,000,00,0,0,0,0,0,0,0000,00,0000,0,0,11,000,001,111,110)

2883
2884
2885 4376:
2886 BUTERROR5:
2887     PO,          DCS=CTR(C0.),          IFORCE AN ERROR, ERROR-CODE=LAST LOADED
2888     NEXT,          J/BUTERROR5          IHANG UP
(5376) DCS(0,00,1,0,0,0) BM(1111,00,00,00,00,000,000,00,0,0,0,0,0,0,0000,00,0000,0,0,11,000,001,111,110)

2889
2890

```

```

2891 6376:
2892 BUTERROR6:
2893 P0, DCS-CTR(C0.), |FORCE AN ERROR, ERROR-CODE=LAST LOADED
2894 NEXT, J/BUTERROR6 |HANG UP
(6376) DCS(0.00,1.0,0,0) BM(1111.00,00.00,00.00,00.00,0.0,0.0,0.0,0.0,0.0000...0.0000,0.11,000,0.011,111,110)

2895
2896
2897 7376:
2898 BUTERROR7:
2899 P0, |DCS-CTR(C0.), |FORCE AN ERROR, ERROR-CODE=LAST LOADED
2900 NEXT, J/BUTERROR7 |HANG UP
(7376) DCS(0.00,0.0,0,0) BM(0000.00,00.00,00.00,00.00,0.0,0.0,0.0,0.0,0.0000...0.0000,0.11,000,0.011,111,110)

2901
2902
2903
2904 1.PAGE=====
2905
2906 .T0C * INIT REGISTERS, CONSOLE DEFAULT ERROR DISPLAY, REVISION CODE
2907
2908
2909 |=====
2910 |*
2911 |* TESTS: INITIALIZATION UNWORDS: 011 + 011
2912 |*
2913 |* FUNCTIONS:
2914 |*
2915 |* TRY TO PUT AN OCTAL (000000) IN THE CONSOLE DISPLAY AS AN ERROR INDICATOR
2916 |*
2917 |* PUT REVISION CODE, BIT15 CLEAR, IN GPR RS
2918 |*
2919 |* SFT FLAGS, FPS, PS, UBREAK REGISTERS TO ALL ZEROS TO DISABLE AS MUCH
2920 |* AS POSSIBLE ANY SPURIOUS HOT FLOATING POINT STARTUPS, UBREAKS, ETC.
2921 |*
2922 |* TURN OFF CACHE (SET BOTH FORCE MISS BITS), AND TURN OFF KT (MEMORY
2923 |* MANAGEMENT (BY CLEARING ENABLE BIT).
2924 |*
2925 |=====
2926
2927
2928 |RETURNED OK, COMPARE OF ENUA:TNUA
2929 |FOR PREV TEST DONE HERE
2930
2931 6532:
2932 INITIALIZE01:
2933 SETUP, RETURN/INITIALIZE03, |GO TO SUBR THAT PUTS REVISION NUMBER,
2934 NEXT, GOTO-PAGE(7), | WITH B<15>=0, INTO B.M. GPR "RS"
| J/INSERTREVNO
(6532) DCS(0.00,0.0,0,0) BM(0111.00,00.00,00.00,00.01,111.0.0,0.0,0.0,0.0,0.0000...0.0000,0.11,100,0.010,001,110)

2935
2936 7001: 1(FREE)
2937 INITIALIZE03:
2938 P1, CSPD(17)_EMIT, EMIT/000014, |EMITCOM TO DISABLE CACHE, KT
2939 NEXT, J/INITIALIZE04
(7001) DCS(0.00,0.0,0,0) BM(0000.10,00.00,00.00,00.01,100.0.0,0.0,0.0,0.0,0.0000...1.0000,0.11,000,0.000,000,011)

```

```

2940
2941 7003: 1(FREE)
2942 INITIALIZE04:
2943 P2-T, D_CSPD(D17), D(C)_0, |GET ABOVE CONSTANT INTO D
2944 P2, RES_CSPD(D17), |BITS<14:11>=0/00/0 FOR SR-LOAD, GUARD-DIS
2945 NEXT, J/INITIALIZE05
(7003) DCS(0.00,0.0,0,0) BM(1010.10,00.00,00.00,00.00,0.0,1.0,0.0,0.0,0.0000...0.1000,1.11,000,0.000,000,100)

2946
2947 7004: 1(FREE)
2948 INITIALIZE05:
2949 P2-T, CCR(07-02)_D(07-02)-[1], |LOAD CCR, DISABLING CACHE
2950 NEXT, J/INITIALIZE06 | BITS<3:2> SET FORCE MISS
(7004) DCS(0.00,0.0,0,0) BM(0001.00,01.00,00.00,00.00,0.0,0.0,0.0,0.0,0.0000...1.1011,0.0000,0.11,000,0.000,000,101)

2951
2952 7005: 1(FREE)
2953 INITIALIZE06:
2954 P2-T, MMRO_D-[1], |LOAD MMRO, DISABLING KT11
2955 NEXT, J/INITIALIZE07 | BIT<00> CLEAR DISABLES KT11
(7005) DCS(0.00,0.0,0,0) BM(0001.00,11.01,00.00,00.00,0.0,0.0,0.0,0.0,0.0000...1.1011,0.0000,0.11,000,0.000,000,110)

2956
2957 7006: 1(FREE)
2958 INITIALIZE07:
2959 P2-T, D_ZERO, |DEFAULT TO ALL ZEROS FOR ERROR
2960 P2, SR_ZERO, |ZERO SR FOR JAMUPP IS ERROR
2961 NEXT, GOTO-PAGE(4), |XFER
2962 J/INITIALIZE10
(7006) DCS(0.00,0.0,0,0) BM(0011.00,00.00,00.00,00.00,100.0.1,1.0,0.0,0.0,0.0000...0.0000,0.11,100,0.001,000,001)

2963
2964 4101: 1(FREE)
2965 INITIALIZE10:
2966 SELECT, UCON-PROC, |ENABLE CLOCKING THE FOLLOWING
2967 ENABLE, EN-CLK-PS(15-12), EN-CLK-PS(7-4), | PROCESSOR REGISTERS;
| EN-CLK-PS(3-0), EN-CLK-FLAG(8-0),
| FW-CLK-FPS(7-4), EN-CLK-UBREAK(11-00),
| RUSDTN_EMIT(15-00),
|FOR UBREAK CONSTANT
2970 P0, SET-UCON-CONTROL, |WRITE CONTROLS
2971 RUMP-VERIFY, |COUNT
2972 P1, RUTA(CUA-TRACK), |RESET CUA TRACKING
2973 NEXT, J/INITIALIZE11
(4101) DCS(0.00,0.0,0,1) BM(1000.01,00.00,01.110.011.0.0,0.0,0.0,0.0,0.0000...1.1001,0.0000,0.11,001,0.001,000,010)

2975
2976 4102: 1(FREE)
2977 INITIALIZE11:
2978 P0, DCS-CTR(C15.), |DISABLE DCS-CTR FOR NOW
2979 EMITC, EMIT/000000, |FOR UBREAK REGISTER LOAD
2980 P2, PS(3-0)_D(3-0),
|
2981 UBREAK_BUSDIN(11-00),
|
2982 PS(15-12)_D(15:13), PS(7-4)_D(7-4),
|
2983 PS(7-4)_D(7-4), FLAG(8-0)_D(15-8),
|
2984 NEXT, RUTD(SCNFE), |NO ERROR: "INITIALIZE12" (+1. WORDS)
2985 J/INITIALIZE12 | ERROR: "TEST001" (BACK AT START)
(4102) DCS(0.00,1.1,0,0) BM(0000.00,00.00,00.00,00.00,0.0,0.0,0.0,0.0,0.0000...1.1010,0.0000,0.11,000,0.000,000,001)
2986

```

```

2987 4001:
2988 INITIALIZE1:
2989     SETUP, RETURN/TEST012A, ;RETURN TO NEXT TEST START
2990     NEXT, CALL[DISPLAY] ;GO DISPLAY CONTENTS OF D-REGISTER IN LIGHTS
(4001) DCS[0.00.0.0.0.0] BM[0101..00.11..11.11..101..111..0.0.0..0.0000...0.0000.0...11.100...010.010.001]

2991
2992
2993
2994 1.PAGE=====
2995
2996 .TOC * TEST012-050: IR DECODE (INSTR1, INSTR5, FLTPT, RELATED "BUTS")
2997
2998 1=====
2999 1*
3000 1* TESTS: 012 - 050 UWORDB: 274 + 206
3001 1*
3002 1* FUNCTIONS:
3003 1*
3004 1* THE FOLLOWING TESTS EXERCISE THE IR-DECODE RELATED LOGIC:
3005 1*
3006 1* INSTR=1, INSTR=5, FLOATING-POINT DECODE "BUTS"
3007 1* IR<15:12>, IR<11> (TWO), IR<9:6>, IR<5:3> BIT "BUTS"
3008 1* MOV/DR7, BYTE/DW0/8MO, DR6=7 DECODE-RELATED "BUTS"
3009 1*
3010 1* NOTE ALSO THAT THE FIRST TIME THE PROCESSOR "UCON":
3011 1* BUSDIN_EMIT, AND EN-CLK-IR, IS EMPLOYED IS IN TEST-012-A.
3012 1*
3013 1=====
3014
3015
3016
3017
3018
3019
3020
3021 1 - - - - -
3022
3023 1*** TEST 012 ***
3024 1TEST-012 USES A DATA PATTERN OF: "1 111 111 111 111 111" (177777)
3025
3026 1 - - - - -
3027
3028 1* PART A *
3029 1TEST-012-A CHECKS THAT BUT[IR<15:12>] READS THE "1111" IN IR<15:12>H CORRECTLY
3030 5775:
3031 TEST012A:
3032     P0, LOAD-ENUA(ZTARGET417), ;LOAD EXPECTED ADDRESS AFTER "BUT"
3033     LOAD-ERROR(TEST012A), ;ERROR DIRECTORY KEY
3034     DCS-CTR(C5.), ;COMPARE ENUA:TWUA IN 5. MICROWORDS
3035     NEXT, J/SETIRA ;SETUP FOR IR TESTS
(5775) DCS[1.00.1.0.0.0] BM[1010..00.11..11.00..001..111..0.0.0..0.0000...0.0000.0...11.000...111.110.000]
3036

```

```

3037 1*** SETUP PROCESSOR UCON FOR BUSDIN <- EMIT, CLOCKING INSTRUCTION REGISTER ***
3038 5760:
3039 SFTIRA:
3040     SELECT, UCON-PROC, ;SELECT PROCESSOR UCON CONTROL:
3041     ENABLF, EN-CLK-IR[15:00], ;ENABLE CLOCK IR OPERATION
3042     BUSDIN_EMIT[15:00], ;PUT EMIT[15:00] ONTO BUSDIN
3043     P0, SET-UCON-CONTROL, ;LOAD UCON REGISTER AT P0
3044     BUMP-VERIFY, ;COUNT
3045     NEXT, J/LOAD012A ;GO TO FIRST TEST, PART A
(5760) DCS[0.00.0.0.0.1] BM[0000..00.00..00.01..000..100...0.0.0..0.0...1.1001...0.0000.0...11.000...000.000.010]
3046
3047
3048 5002: 1(FREE)
3049 LOAD012A:
3050     P2-U, IR_EMIT, ;LOAD IR WITH TEST PATTERN
3051     EMIT/177777, ;(177777)
3052     NEXT, J/G0BUT012A ;GO SETUP FOR "BUT"
(5002) DCS[0.00.0.0.0.0] BM[1111..00.11..11.11..111..111...0.0.0..0.0...1.1010...0.0000.0...11.000...000.000.011]
3053
3054
3055 5003: 1(FREE)
3056 G0BUT012A:
3057     SETUP, RETURN/TEST012B, ;RETURN TO START OF NEXT SUBTEST
3058     NEXT, GOTO-PAGE(7), ;BUT TABLE IS ON PAGE 7
3059     J/BUTIR15-12 ;GO DO "BUT" ON IR<15:12>H
(5003) DCS[0.00.0.0.0.0] BM[0101..00.11..11.11..011..111...0.0.0..0.0...0.0000...0.0000.0...11.100...011.000.000]
3060
3061
3062 1 - - - - -
3063
3064 1* PART B *
3065 1TEST-012-B CHECKS THAT BUT[IR<11>:FLTPT<3:0>] READS THE "1" IN IR<11>H CORRECTLY,
3066 1AND THE FLTPT DECODE ROM GETS ADDRESS (776), WHICH IS A CMPF/D INSTR;
3067 1DATA OUTPUT SHOULD BE (17)
3068 5773:
3069 TEST012B:
3070     P0, LOAD-ENUA(ZTARGET437), ;LOAD EXPECTED ADDRESS AFTER "BUT"
3071     LOAD-ERROR(TEST012B), ;ERROR DIRECTORY KEY
3072     DCS-CTR(C3.), ;COMPARE ENUA:TWUA IN 3. MICROWORDS
3073     NEXT, J/G0BUT012B ;GO SETUP FOR "BUT"
(5773) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..011..111...0.0.0..0.0...0.0000...0.0000.0...11.000...000.000.100]
3074
3075
3076 5004: 1(PREF)
3077 G0BUT012B:
3078     SETUP, RETURN/TEST012C, ;RETURN TO START OF NEXT SUBTEST
3079     NEXT, GOTO-PAGE(7), ;BUT TABLE IS ON PAGE 7
3080     J/BUTIR11:FLTPT3=0 ;GO DO "BUT" ON IR<11>H:FLTPT<3:0>H
(5004) DCS[0.00.0.0.0.0] BM[0101..00.11..11.11..001..111...0.0.0..0.0...0.0000...0.0000.0...11.100...011.000.010]
3081
3082
3083 1 - - - - -

```

```
3084
3085 1* PART C *
3086 1TEST-012-C CHECKS THAT BUTR[IR<11>B] READS THE "1" IN IR<11>H CORRECTLY
3087 57711
3088 TEST012C:
3089     PO,          LOAD-ENUA(ZTARGET403),          |LOAD EXPECTED ADDRESS AFTER "BUT"
3090                  LOAD-ERROR(TEST012C),          |ERROR DIRECTORY KEY
3091                  DCS-CTR(C3),                    |COMPARE ENUA:TWUA IN 3, MICROWORDS
3092                  BUMP-VERIFY,                    |COUNT
3093                  NEXT, J/GOBUTO12C              |GO SETUP FOR "BUT"
3094 (5771) DCS(1,00,1,0,0,1) BM(1100,00,11,11,00,000,001,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,000,101)
3095
3096 5005: 1(FREE)
3097 GOBUTO12C:
3098     SETUP,        RETURN/TEST012D,              |RETURN TO START OF NEXT SUBTEST
3099     NEXT,         GOTO-PAGE(7),                  |BUT TABLE IS ON PAGE 7
3100                  J/BUTR11B                      |GO DO "BUT" ON IR<11>H
3101 (5005) DCS(0,00,0,0,0,0) BM(0101,00,11,11,10,111,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,100,010)
3102
3103 1 - - - - -
3104
3105 1* PART D *
3106 1TEST-012-D CHECKS THAT BUT[IR<9:6>] READS THE "1111" IN IR<9:6>H CORRECTLY
3107 57671
3108 TEST012D:
3109     PO,          LOAD-ENUA(ZTARGET417),          |LOAD EXPECTED ADDRESS AFTER "BUT"
3110                  LOAD-ERROR(TEST012D),          |ERROR DIRECTORY KEY
3111                  DCS-CTR(C3),                    |COMPARE ENUA:TWUA IN 3, MICROWORDS
3112                  NEXT, J/GOBUTO12D              |GO SETUP FOR "BUT"
3113 (5767) DCS(1,00,1,0,0,0) BM(1100,00,11,11,00,001,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,000,110)
3114
3115 5006: 1(FREE)
3116 GOBUTO12D:
3117     SETUP,        RETURN/TEST012E,              |RETURN TO START OF NEXT SUBTEST
3118     NEXT,         GOTO-PAGE(7),                  |BUT TABLE IS ON PAGE 7
3119                  J/BUTR9-6                      |GO DO "BUT" ON IR<9:6>H
3120 (5006) DCS(0,00,0,0,0,0) BM(0101,00,11,11,10,101,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,100)
3121
3122 1 - - - - -
3123
3124 1* PART E *
3125 1TEST-012-E CHECKS THAT BUT(MOV-DR7:IR<5:3>) READS THE (-FLTPT+MOV+FLTPT+DR7) AND "11" IN IR<5:3>H CORRECTLY
3126 57651
3127 TEST012E:
3128     PO,          LOAD-ENUA(ZTARGET417),          |LOAD EXPECTED ADDRESS AFTER "BUT"
3129                  LOAD-ERROR(TEST012E),          |ERROR DIRECTORY KEY
3130                  DCS-CTR(C3),                    |COMPARE ENUA:TWUA IN 3, MICROWORDS
3131                  NEXT, J/GOBUTO12E              |GO SETUP FOR "BUT"
```

```
(5765) DCS(1,00,1,0,0,0) BM(1100,00,11,11,00,001,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,000,111)
3132
3133 5007: 1(FREE)
3134 GOBUTO12E:
3135     SETUP,        RETURN/TEST012F,              |RETURN TO START OF NEXT SUBTEST
3136     NEXT,         GOTO-PAGE(7),                  |BUT TABLE IS ON PAGE 7
3137                  J/BUTMOVDR7IRS=3              |GO DO "BUT" ON (-FLTPT+MOV+FLTPT+DR7) & IR<5:3>H
3138 (5007) DCS(0,00,0,0,0,0) BM(0101,00,11,11,10,100,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,101)
3139
3140 1 - - - - -
3141
3142 1* PART F *
3143 1TEST-012-F CHECKS THAT BUT(DR6-7 L) READS THE "11" IN IR<2:1>H CORRECTLY
3144 1AND DOES ASSERT THE SIGNAL
3145 57641
3146 TEST012F:
3147     PO,          LOAD-ENUA(ZTARGET402),          |LOAD EXPECTED ADDRESS AFTER "BUT"
3148                  LOAD-ERROR(TEST012F),          |ERROR DIRECTORY KEY
3149                  DCS-CTR(C3),                    |COMPARE ENUA:TWUA IN 3, MICROWORDS
3150                  NEXT, J/GOBUTO12F              |GO SETUP FOR "BUT"
3151 (5764) DCS(1,00,1,0,0,0) BM(1100,00,11,11,00,000,010,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,001,000)
3152
3153 5010: 1(FREE)
3154 GOBUTO12F:
3155     SETUP,        RETURN/TEST012G,              |RETURN TO START OF NEXT SUBTEST
3156     PO,          BUMP-VERIFY,                    |COUNT
3157     NEXT,         GOTO-PAGE(7),                  |BUT TABLE IS ON PAGE 7
3158                  J/BUTDR6-7L                    |GO DO "BUT" ON DR 6/7 L
3159 (5010) DCS(0,00,0,0,0,1) BM(0101,00,11,11,10,011,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,100,111)
3160
3161 1 - - - - -
3162
3163 1* PART G *
3164 1TEST-012G CHECKS THAT BUT(INSTR1) READS THE ONES IN IR<15:00>H CORRECTLY
3165 1AS CLASS-A-FLTPT, AND CORRECTLY TARGETS TO INSTR1 FLTPT (474)
3166 1 BIT<1:0> = FLAG44:5>H = "00" FROM INITIALIZATION CODE
3167 57631
3168 TEST012G:
3169     PO,          LOAD-ENUA(ZTARGET474),          |LOAD EXPECTED ADDRESS AFTER "BUT"
3170                  LOAD-ERROR(TEST012G),          |ERROR DIRECTORY KEY
3171                  DCS-CTR(C3),                    |COMPARE ENUA:TWUA IN 3, MICROWORDS
3172                  NEXT, J/GOBUTO12G              |GO SETUP FOR "BUT"
3173 (5763) DCS(1,00,1,0,0,0) BM(1100,00,11,11,00,111,100,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,001,001)
3174
3175 5011: 1(FREE)
3176 GOBUTO12G:
3177     SETUP,        RETURN/SCOPE012,              |RETURN TO SCOPE LOOP TEST WORD
3178
```

```

3179      NEXT,      GOTO=PAGE(7),      IBUT TABLE IS ON PAGE 7
3180      J/BUTINSTR1      IGD DO INSTR1 "BUT"
(5011) DCS[0.00,0.0,0.0] BNI[0101..00.00..00.01..010..111..0,0,0..0,0...0.0000...0..0000,0...11,100...011,000,110]
3181
3182
3183
3184      5012: 1(FREE)
3185      SCOPE012:
3186      PO,      BUMP=VERIFY,      ICOUNT
3187      NEXT,      BUTD(SCOPE),      IWD ERROR: "TEST013A" [+1, WORDS]
3188      J/TEST013A      I ERROR: "SETIRA" [-16, WORDS]
(5012) DCS[0.00,0.1,0.1] BNI[0000..00.00..00.00..000..000..0,0,0..0,0...0.0000...0..0000,0...11,000...111,110,001]
3189
3190
3191
3192
3193
3194
3195      1 - - - - -
3196
3197      I** TEST 013 **
3198      ITEST=013 USES AN IR DATA PATTERN OF: "0 000 000 000 000 000" (0000000)
3199
3200      1 - - - - -
3201
3202      I* PART A *
3203      ITEST=013=A CHECKS THAT BUT (IR<15:12>) READS THE "0000" IN IR<15:12>N CORRECTLY
3204      5761:
3205      TEST013A:
3206      PO,      LOAD=ENUA(ZTARGET400),      ILOAD EXPECTED ADDRESS AFTER "BUT"
3207      LOAD=ERROR(TEST013A),      IERROR DIRECTORY KEY
3208      DCS=CTR(C4.),      ICOMPARE ENUA:THUA IN 4, MICROWORDS
3209      NEXT,      J/LOAD013A      IGD LOAD PATTERN
(5761) DCS[1.00,1.0,0.0] BNI[1011..00.11..11,00..000..000..0,0,0..0,0...0.0000...0..0000,0...11,000...111,100,100]
3210
3211
3212      5744:
3213      LOAD013A:
3214      P2="I,      IR_EMIT,      ILOAD IR WITH TEST PATTERN
3215      EMIT/0000000,      I(000000)
3216      NEXT,      J/GDBUT013A      IGD SETUP FOR "BUT"
(5744) DCS[0.00,0.0,0.0] BNI[0000..00.00..00.00..000..000..0,0,0..0,0...1.1010...0..0000,0...11,000...000,001,011]
3217
3218
3219      5013: 1(FREE)
3220      GORBUT013A:
3221      RPTUP,      RETURN/TEST013B,      IRETURN TO START OF NEXT SUBTEST
3222      NEXT,      GOTO=PAGE(7),      IBUT TABLE IS ON PAGE 7
3223      J/BUTIR15=12      IGD DO "BUT" ON IR<15:12>N
(5013) DCS[0.00,0.0,0.0] BNI[0101..00.11..11,01..111..111..0,0,0..0,0...0.0000...0..0000,0...11,100...011,000,000]
3224
3225

```

```

3276      *
3277      *
3278      *
3279      *
3280      *
3281      *
3282      *
3283      *
3284      *
3285      *
3286      *
3287      *
3288      *
3289      *
3290      *
3291      *
3292      *
3293      *
3294      *
3295      *
3296      *
3297      *
3298      *
3299      *
3300      *
3301      *
3302      *
3303      *
3304      *
3305      *
3306      *
3307      *
3308      *
3309      *
3310      *
3311      *
3312      *
3313      *
3314      *
3315      *
3316      *
3317      *
3318      *
3319      *
3320      *
3321      *
3322      *
3323      *
3324      *
3325      *
3326      *
3327      *
3328      *
3329      *
3330      *
3331      *
3332      *
3333      *
3334      *
3335      *
3336      *
3337      *
3338      *
3339      *
3340      *
3341      *
3342      *
3343      *
3344      *
3345      *
3346      *
3347      *
3348      *
3349      *
3350      *
3351      *
3352      *
3353      *
3354      *
3355      *
3356      *
3357      *
3358      *
3359      *
3360      *
3361      *
3362      *
3363      *
3364      *
3365      *
3366      *
3367      *
3368      *
3369      *
3370      *
3371      *
3372      *
3373      *
3374      *
3375      *
3376      *
3377      *
3378      *
3379      *
3380      *
3381      *
3382      *
3383      *
3384      *
3385      *
3386      *
3387      *
3388      *
3389      *
3390      *
3391      *
3392      *
3393      *
3394      *
3395      *
3396      *
3397      *
3398      *
3399      *

```

```

3274          DCS-CTR(C3.),          |ERROR DIRECTORY KEY
3275          DCS-CTR(C3.),          |COMPARE ENUA;ENUA IN 3, MICROWORDS
3276          NEXT, J/GOBUTO13D      |GO SETUP FOR "BUT"
(5753) DCS[1,00,1,0,0,0] BM[1100,00,11,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,000,001,110]

3277
3278
3279          5016: 1(FREE)
3280          GOBUTO13D:
3281          SETUP, RETURN/TEST013E, |RETURN TO START OF NEXT SUBTEST
3282          NEXT, GOTO=PAGE(7),      |BUT TABLE IS ON PAGE 7
3283          J/BUTIRS=6               |GO DO "BUT" ON IR<5>6H
(5016) DCS[0,00,0,0,0,0] BM[0101,00,11,01,001,001,0,0,0,0,0,0,0,0000,0,0,11,100,0,011,000,100]

3284
3285
3286          | - - - - -
3287
3288          |* PART F *
3289          |TEST-013-E CHECKS THAT BUT(MOV=DR7;IR<5>3) READS THE -(FLTP=MOV=FLTP=DR7) AND "000" IN IR<5>3>H CORRECTLY
3290          5751:
3291          TEST013E:
3292          PO,                      |LOAD EXPECTED ADDRESS AFTER "BUT"
3293          LOAD=ENUA(ZTARGET400),   |LOAD=ERROR(TEST013E),
3294          DCS-CTR(C3.),           |ERROR DIRECTORY KEY
3295          NEXT, J/GOBUTO13E       |COMPARE ENUA;ENUA IN 3, MICROWORDS
3296          |GO SETUP FOR "BUT"
(5751) DCS[1,00,1,0,0,0] BM[1100,00,11,00,000,000,0,0,0,0,0,0,0,0000,0,0,11,000,0,000,001,111]

3296
3297
3298          5017: 1(FREE)
3299          GOBUTO13E:
3300          SETUP, RETURN/TEST013F, |RETURN TO START OF NEXT SUBTEST
3301          PO, BUMP=VERIFY,         |COUNT
3302          NEXT, GOTO=PAGE(7),      |BUT TABLE IS ON PAGE 7
3303          J/BUTMOVDR7IRS=3         |GO DO "BUT" ON (-FLTP=MOV=FLTP=DR7) & IR<5>3>H
(5017) DCS[0,00,0,0,0,1] BM[0101,00,11,01,000,001,0,0,0,0,0,0,0,0000,0,0,11,100,0,011,000,101]

3304
3305
3306          | - - - - -
3307
3308          |* PART F *
3309          |TEST-013-F CHECKS THAT BUT(INSTR5) READS THE ZERO5 IN IR<5>00>H CORRECTLY
3310          |AS (000000)=HALT, AND CORRECTLY TARGETS TO (434)
3311          5750:
3312          TEST013F:
3313          PO,                      |LOAD EXPECTED ADDRESS AFTER "BUT"
3314          LOAD=ENUA(ZTARGET434),   |LOAD=ERROR(TEST013F),
3315          DCS-CTR(C3.),           |ERROR DIRECTORY KEY
3316          NEXT, J/GOBUTO13F       |COMPARE ENUA;ENUA IN 3, MICROWORDS
3317          |GO SETUP FOR "BUT"
(5750) DCS[1,00,1,0,0,0] BM[1100,00,11,00,011,100,0,0,0,0,0,0,0,0000,0,0,11,000,0,000,010,000]

3317
3318
3319          5020: 1(FREE)
3320          GOBUTO13F:

```

```

3371          SETUP, RRETURN/TEST013G,                |RETURN TO START OF NEXT SUBTEST
3372          NEXT,  GOTO=PAGE(7),                      |BUT TABLE IS ON PAGE 7
3373          J/BUTINSTRS                                |GO DO INSTRS "BUT"
(5070) DCS(0.00,0.0,0.0) BM[0101..00,11..11,00..111..111..0,0,0..0,0,0,0,0,0000...0,0000,0...11,100...011,000,001]
3374
3375          | - - - - -
3376
3377          |* PART G *
3378          |TEST-013-G CHECKS THAT BUT[INSTR1] READS THE ZEROS IN IR<15:06>H CORRECTLY
3379          |AS NOT[CLASS=A THRU G], AND CORRECTLY TARGETS TO (417) [OTHER]
3380          5747:
3381          TEST013G:
3382          PO,          LOAD=ENUA(ZTARGET417),          |LOAD EXPECTED ADDRESS AFTER "BUT"
3383          LOAD=ERROR(TEST013G),                      |ERROR DIRECTORY KEY
3384          DCS=CTR(C3,),                                |COMPARE ENUA;THUA IN 3, MICROWORDS
3385          NEXT,        J/GOBUTO13G                    |GO SETUP FOR "BUT"
(5747) DCS(1.00,1.0,0.0) BM[1100..00,11..11,00..001..111..0,0,0..0,0,0,0,0,0000...0,0000,0...11,000...000,010,001]
3386
3387          5021: I(FREE)
3388          GOBUTO13G:
3389          SETUP,      RETURN/SCOPE013,                |RETURN TO SCOPE LOOP TEST WORD
3390          NEXT,        GOTO=PAGE(7),                  |BUT TABLE IS ON PAGE 7
3391          J/BUTINSTR1 |GO DO INSTR1 "BUT"
(5021) DCS(0.00,0.0,0.0) BM[0101..00,00..00,10..010..111..0,0,0..0,0,0,0,0,0000...0,0000,0...11,100...011,000,110]
3392
3393          5022: I(FREE)
3394          SCOPE013:
3395          NEXT,        BUTD[SCOPE],                    |NO ERROR: "TEST014A" [+1, WORD]
3396          J/TEST014A | ERROR: "LOAD013A" [-14, WORDS]
(5022) DCS(0.00,0.0,0.0) BM[0000..00,00..00,00..000..000..0,0,0..0,0,0,0,0,0000...0,0000,0...11,000...111,100,101]
3397
3398          | - - - - -
3399
3400          |*** TEST 014 ***
3401          |TEST-014 USES AN IR DATA PATTERN OF: "0 000 000 001 010 101" (000125)
3402          | - - - - -
3403
3404          |* PART A *
3405          |TEST-011-A CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
3406          |AS ROM ADDRESS=(125) ON THE INSTR5 E88 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
3407          |OF (12), TARGETING TO (432) AFTER THE DECODE
3408          5745:
3409          TEST014A:

```

```

3369      PO,          LOAD=ENUA(ZTARGET432),          ILOAD EXPECTED ADDRESS AFTER "BUT"
3370      LOAD=ERROR(TEST014A),          IERROR DIRECTORY KEY
3371      DCS=CTR(C4.),          ICOMPARE ENUA;TNUA IN 4. MICROWORDS
3372      NEXT,          J/LOAD014A          IGO LOAD PATTERN
(5745) DCS(1.00,1.0,0,0) BM(1011.00.11.11.00.011.010.00.0.0.0.0.0000.00.0000.0.11.100.010)

3373
3374      5742:
3375      LOAD014A:
3376      P2=U,          IR_EMITT,          ILOAD IR WITH TEST PATTERN
3377      EMIT/000125,          I(000125)
3378      NEXT,          J/GOBUTO14A          IGO SETUP FOR "BUT"
3379      (5742) DCS(0.00,0.0,0,0) BM(0000.00.00.00.01.010.101.00.0.0.0.0.1010.00.0000.0.11.100.000.010.011)

3380
3381      5023: 1(FREE)
3382      GOBUTO14A:
3383      SETUP,          RETURN/TEST014B,          IRETURN TO START OF NEXT SUBTEST
3384      NEXT,          GOTO=PAGE(7),          IBUT TABLE IS ON PAGE 7
3385      J/BUTINSTRB          IGO DO INSTRB "BUT"
(5023) DCS(0.00,0.0,0,0) BM(0101.00.11.11.01.010.111.00.0.0.0.0.0.0000.00.0000.0.11.100.001.000.001)

3386
3387      I - - - - -
3388
3389
3390      I* PART B *
3391      ITEST-014-B CHECKS THAT BUT(DR6-7 L) READS THE "10" IN IR(2:1) H CORRECTLY
3392      IAND DOES NOT ASSERT THE SIGNAL
3393      5752:
3394      TEST014B:
3395      PO,          LOAD=ENUA(ZTARGET403),          ILOAD EXPECTED ADDRESS AFTER "BUT"
3396      LOAD=ERROR(TEST014B),          IERROR DIRECTORY KEY
3397      DCS=CTR(C3.),          ICOMPARE ENUA;TNUA IN 3. MICROWORDS
3398      BUMP=VERIFY,          ICDUNT
3399      NEXT,          J/GOBUTO14B          IGO SETUP FOR "BUT"
(5752) DCS(1.00,1.0,0,1) BM(1100.00.11.11.00.000.011.00.0.0.0.0.0000.00.0000.0.11.100.000.010.100)

3400
3401
3402      5024: 1(FREE)
3403      GOBUTO14B:
3404      SETUP,          RETURN/TEST014C,          IRETURN TO START OF NEXT SUBTEST
3405      NEXT,          GOTO=PAGE(7),          IBUT TABLE IS ON PAGE 7
3406      J/BUTDR6=7L          IGO DO DR 6-7 L "BUT"
(5024) DCS(0.00,0.0,0,0) BM(0101.00.11.11.10.010.111.00.0.0.0.0.0.0000.00.0000.0.11.100.001.100.111)

3407
3408      I - - - - -
3409
3410      I* PART C *
3411      ITEST-014-C CHECKS THAT BUT(INSTR1) READS THE IR CORRECTLY
3412      IAN CLASS=F=JMP, IR(5:3)H="010", AND TARGETS TO (632)
3413      5762:
3414      TEST014C:
3415      PO,          LOAD=ENUA(ZTARGET652),          ILOAD EXPECTED ADDRESS AFTER "BUT"

```

```

3414 LOAD=ERROR(TEST014C),                                |ERROR DIRECTORY KEY
3417 DCS=CTR(C3.),                                         |COMPARE ENUA:TNUA IN 3, MICROWORDS
3418 NEXT, J/GOBUT014C                                    |GO SETUP FOR "BUT"
(5762) DCS[1,00,1,0,0,0] BM[1100,00,11,11,10,101,010,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,010,101]

3419
3420
3421 5025: 1(FREE)
3422 GOBUT014C:
3423 SETUP, RETURN/TEST014D,                                |RETURN TO START OF NEXT SUBTEST
3424 NEXT, GOTO=PAGE(7),                                    |BUT TABLE IS ON PAGE 7
3425 J/BUTINSTR1,                                           |GO DO INSTR1 "BUT"
(5025) DCS[0,00,0,0,0,0] BM[0101,00,11,10,01,000,011,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,000,011,000,110]

3426
3427
3428
3429 | - - - - -
3430
3431 |* PART D *
3432 |TEST-014-D CHECKS THAT BUT[IR<11>%FLTP<3>0] READS THE "0" IN IR<11>H CORRECTLY,
3433 |AND THE FLTP DECODE ROM GETS ADDRESS (020), WHICH IS A STST INSTR;
3434 |DATA OUTPUT SHOULD BE (01)
3435 5710:
3436 TEST014D:
3437 PO, LOAD=ENUA(ZTARGET401),                                |LOAD EXPECTED ADDRESS AFTER "BUT"
3438 LOAD=ERROR(TEST014D),                                |ERROR DIRECTORY KEY
3439 DCS=CTR(C3.),                                         |COMPARE ENUA:TNUA IN 3, MICROWORDS
3440 NEXT, J/GOBUT014D                                    |GO SETUP FOR "BUT"
(5710) DCS[1,00,1,0,0,0] BM[1100,00,11,11,00,000,001,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,010,110]

3441
3442
3443 5026: 1(FREE)
3444 GOBUT014D:
3445 SETUP, RETURN/SCOPE014,                                |RETURN TO SCOPE LOOP TEST WORD
3446 NEXT, GOTO=PAGE(7),                                    |BUT TABLE IS ON PAGE 7
3447 J/BUTIR1FLTP<3>0,                                     |GO DO "BUT" ON IR<11>H%FLTP<3>0>H
(5026) DCS[0,00,0,0,0,0] BM[0101,00,00,00,10,111,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,000,011,000,010]

3448
3449
3450
3451 5027: 1(FREE)
3452 SCOPE014:
3453 NEXT, BUTD(SCOPE),                                       |NO ERROR: "TEST015A" [+1, WORD]
3454 J/TFST015A,                                           |ERROR: "LOAD014A" [-8, WORDS]
(5027) DCS[0,00,0,1,0,0] BM[0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,000,111,100,011]

3455
3456
3457
3458
3459
3460
3461 | - - - - -
3462

```

```

3463 1*** TEST 015 ***
3464 !TEST-015 USES AN IR DATA PATTERN OF: "0 000 000 001 101 010" (000152)
3465
3466 | - - - - -
3467
3468 !* PART A *
3469 !TEST-015-A CHECKS THAT BUT[INSTRS] READS THE IR CORRECTLY
3470 !AS ROM ADDRESS(152) ON THE INSTRS E88 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
3471 !OF (05), TARGETING TO (425) AFTER THE DECODE
3472 5743:
3473 TEST015A:
3474     PO,          LOAD-ENUA(2TARGET425),          !LOAD EXPECTED ADDRESS AFTER "BUT"
3475                LOAD-ERROR(TEST015A),          !ERROR DIRECTORY KEY
3476                DCS-CTR(C4,),          !COMPARE ENUA:THUA IN 4, MICROWORDS
3477     NEXT,        J/LOAD015A          !GO LOAD PATTERN
(5743) DCS[1,0,1,0,0,0] BM[1011,00,11,11,00,010,101,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,100,000]
3478
3479 5740:
3480 LOAD015A:
3481     PO,          BUMP-VERIFY,          !COUNT
3482     P2-U,        IP-EMIT,          !LOAD IR WITH TEST PATTERN
3483                EMIT/000152,          ! (000152)
3484     NEXT,        J/GOBUTO15A          !GO SETUP FOR "BUT"
3485 (5740) DCS[0,0,0,0,0,0,1] BM[0000,00,00,00,01,101,010,0,0,0,0,0,0,0,1010,0,0,0,0000,0,0,11,1000,0,000,011,000]
3486
3487 5030: 1(FREE)
3488 GOBUTO15A:
3489     SETUP,       RETURN/TEST015B,          !RETURN TO START OF NEXT SUBTEST
3490     NEXT,        GOTO-PAGE(7),          !BUT TABLE IS ON PAGE 7
3491     J/BUTINSTRS          !GO DO INSTRS "BUT"
(5030) DCS[0,0,0,0,0,0,0] BM[0101,00,11,11,11,010,111,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,100,0,011,000,001]
3492
3493 | - - - - -
3494
3495 !* PART B *
3496 !TEST-015-B CHECKS THAT BUT[DR6-7 L] READS THE "01" IN IR(211) H CORRECTLY
3497 !AND DOES NOT ASSEPT THE SIGNAL
3498 5772:
3499 TEST015B:
3500     PO,          LOAD-ENUA(2TARGET403),          !LOAD EXPECTED ADDRESS AFTER "BUT"
3501                LOAD-ERROR(TEST015B),          !ERROR DIRECTORY KEY
3502                DCS-CTR(C3,),          !COMPARE ENUA:THUA IN 3, MICROWORDS
3503     NEXT,        J/GOBUTO15B          !GO SETUP FOR "BUT"
(5772) DCS[1,0,0,1,0,0,0] BM[1100,00,11,11,00,000,011,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,1000,0,000,011,001]
3504
3505 5031: 1(FREE)
3506 GOBUTO15B:
3507     SETUP,       RETURN/TEST015C,          !RETURN TO START OF NEXT SUBTEST
3508     NEXT,        GOTO-PAGE(7),          !BUT TABLE IS ON PAGE 7
3509     J/RUTDR6-7L          !GO DO DR 6-7 L "BUT"
3510
3511

```

```

3512 (5031) DCS[0.00,0.0,0.0] BM[0101..00.11..10.10..010..111..0.0,0.0,0.0..0.0000...0..0000.0...11.100...011.100,111]
3513 ! - - - - -
3514
3515 1* PART C *
3516 1TEST-015-C CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3517 1AS CLASS-F=JMP, IR<5:3>H=101", AND TARGETS TO (655)
3518 5722:
3519 TEST015C:
3520 PO, LOAD-ENUA(ZTARGET655), 1LOAD EXPECTED ADDRESS AFTER "BUT"
3521 LOAD-ERROR(TEST015C), 1ERROR DIRECTORY KEY
3522 DCS-CTR(C3.), 1COMPARE ENUA:THUA IN 3, MICROWORDS
3523 NEXT, J/GOBUTO15C 1GO SETUP FOR "BUT"
3524 (5722) DCS[1.00,1.0,0.0] BM[1100..00.11..11.10..101..101..0.0,0.0,0.0..0.0000...0..0000.0...11.100...000.011.010]
3525
3526 5032: 1(FREF)
3527 GOBUTO15C:
3528 SETUP, RETURN/SCOPE015, 1RETURN TO SCOPE LOOP TEST WORD
3529 NEXT, GOTO-PAGE(7), 1BUT TABLE IS ON PAGE 7
3530 J/BUTINSTR1 1GO DO INSTR1 "BUT"
3531 (5032) DCS[0.00,0.0,0.0] BM[0101..00.00..00.11..011..111..0.0,0.0,0.0..0.0000...0..0000.0...11.100...011.000,110]
3532
3533 5033: 1(FREF)
3534 SCOPE015:
3535 PO, BUMP=VERIFY, 1COUNT
3536 NEXT, BUTD[SCOPE], 1NO ERROR: "TEST016A" [+1. WORD]
3537 J/TEST016A 1 ERROR: "LOAD016A" [-6. WORDS]
3538 (5033) DCS[0.00,0.1,0.1] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0..0.0000...0..0000.0...11.100...111.100,001]
3539
3540
3541
3542
3543 ! - - - - -
3544
3545 1** TEST 016 **
3546 1TEST-016 USES AN IR DATA PATTERN OF: "1 000 000 000 000 000" (1000000)
3547
3548 ! - - - - -
3549
3550 1* PART A *
3551 1TEST-016-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3552 1AS CLASS-F=BRANCH, IR<14:11>H="0000", IR<15:1010>H="0000",
3553 1AND TARGETS TO (757)
3554
3555 5741:
3556 TEST016A:
3557 PO, LOAD-ENUA(ZTARGET757), 1LOAD EXPECTED ADDRESS AFTER "BUT"
3558 LOAD-ERROR(TEST016A), 1ERROR DIRECTORY KEY
3559 DCS-CTR(C4.), 1COMPARE ENUA:THUA IN 4, MICROWORDS
3560 NEXT, J/LOAD016A 1GO LOAD PATTERN

```

```
(5741) DCS[1.00.1.0.0.0] BM[1011.00.11.11.11.101.111.0.0.0.0.0.0.0000.0.0.0000.0.11.000.111.011.110]
3560
3561
3562 5736:
3563 LOAD016A:
3564 P2=U, IR_EMIT, ILOAD IR WITH TEST PATTERN
3565 EMIT/100000, I(100000)
3566 NEXT, J/GOBUT016A IGO SETUP FOR "BUT"
(5736) DCS[0.00.0.0.0.0] BM[1000.00.00.00.00.000.000.0.0.0.0.0.1.1010.0.0.0000.0.11.000.000.011.100]
3567
3568
3569 5034: I(FREE)
3570 GORUT016A:
3571 SETUP, RETURN/SCOPE016, IRETURN TO SCOPE LOOP TEST WORD
3572 PO, BUMP-VERIFY, ICOUNT
3573 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
3574 J/BUTINSTR1 IGO DO INSTR1 "BUT"
(5034) DCS[0.00.0.0.0.1] BM[0101.00.00.00.11.101.111.0.0.0.0.0.0.0000.0.0.0000.0.11.100.011.000.110]
3575
3576
3577 5035: I(FREE)
3578 SCOPE016:
3579 NEXT, BUTD[SCOPE], INO ERROR: "TEST017A" [+1, WORD]
3580 J/TEST017A I ERROR: "LOAD016A" [-2, WORDS]
(5035) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.000.111.011.111]
3581
3582
3583
3584
3585
3586
3587
3588
3589 I*** TEST 017 ***
3590 ITEST-017 USES A DATA PATTERN OF: "0 000 000 101 010 010" (000522)
3591
3592
3593
3594 I* PART A *
3595 ITEST-017-A CHECKS THAT BUT[IR<916>] READS THE ALTERNATING PATTERN "0101"
3596 IIN IR<916>H CORRECTLY
3597 5737:
3598 TEST017A:
3599 PO, LOAD-ENUA(ZTARGET405), ILOAD EXPECTED ADDRESS AFTER "BUT"
3600 LOAD-ERROR(TEST017A), IERROR DIRECTORY KEY
3601 DCS-CTR(C4), ICOMPARE ENUA;TNUA IN 4, MICROWORDS
3602 NEXT, J/LOAD017A IGO LOAD PATTERN
(5737) DCS[1.00.1.0.0.0] BM[1011.00.11.11.00.000.101.0.0.0.0.0.0.0000.0.0.0000.0.11.000.111.011.010]
3603
3604
3605 5732:
3606 LOAD017A:
3607 P2=U, IR_EMIT, ILOAD IR WITH TEST PATTERN
```

```
3608 EMIT/000522, I(000522)
3609 NEXT, J/GOBUT017A IGO SETUP FOR "BUT"
(5732) DCS[0.00.0.0.0.0] BM[0000.00.00.01.01.010.010.0.0.0.0.0.1.1010.0.0.0000.0.11.000.000.011.110]
3610
3611
3612 5036: I(FREE)
3613 GORUT017A:
3614 SETUP, RETURN/TEST017B, IRETURN TO START OF NEXT SUBTEST
3615 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
3616 J/RUTIR9-6 IGO DO "BUT" ON IR<916>H
(5036) DCS[0.00.0.0.0.0] BM[0101.00.11.10.11.101.111.0.0.0.0.0.0.0000.0.0.0000.0.11.100.011.000.100]
3617
3618
3619
3620
3621 I* PART B *
3622 ITEST-017-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3623 IAS CLASS-F=BRANCH, IR<1411>H="0000", IR<1510100>H="0000",
3624 IAND TARGETS TO (757)
3625 5735:
3626 TEST017B:
3627 PO, LOAD-ENUA(ZTARGET757), ILOAD EXPECTED ADDRESS AFTER "BUT"
3628 LOAD-ERROR(TEST017B), IERROR DIRECTORY KEY
3629 DCS-CTR(C3), ICOMPARE ENUA;TNUA IN 3, MICROWORDS
3630 BUMP-VERIFY, ICOUNT
3631 NEXT, J/GOBUT017B IGO SETUP FOR "BUT"
(5735) DCS[1.00.1.0.0.1] BM[1100.00.11.11.11.101.111.0.0.0.0.0.0.0000.0.0.0000.0.11.000.000.011.111]
3632
3633
3634 5037: I(FREE)
3635 GORUT017B:
3636 SETUP, RETURN/SCOPE017, IRETURN TO SCOPE LOOP TEST WORD
3637 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
3638 J/BUTINSTR1 IGO DO INSTR1 "BUT"
(5037) DCS[0.00.0.0.0.0] BM[0101.00.00.01.00.000.111.0.0.0.0.0.0.0000.0.0.0000.0.11.100.011.000.110]
3639
3640
3641 5040: I(FREE)
3642 SCOPE017:
3643 NEXT, BUTD[SCOPE], INO ERROR: "TEST020A" [+1, WORD]
3644 J/TEST020A I ERROR: "LOAD017A" [-4, WORDS]
(5040) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.000.111.011.011]
3645
3646
3647
3648
3649
3650
3651
3652
3653 I*** TEST 020 ***
3654 ITEST-020 USES A DATA PATTERN OF: "0 000 001 010 101 111" (001257)
```

```
3655
3656 1 - - - - -
3657
3658 1* PART A *
3659 1TEST-020-A CHECKS THAT BUT[IR<916>] READS THE ALTERNATING PATTERN "1010"
3660 1IN IR<916>H CORRECTLY
3661 5733:
3662 TEST020A:
3663     PO,      LOAD-ENUA(ZTARGET412),      1LOAD EXPECTED ADDRESS AFTER "BUT"
3664             LOAD-ERROR(TEST020A),        1ERROR DIRECTORY KEY
3665             DCS-CTR(C4.),                1COMPARE ENUA:TNVA IN 4, MICROWORDS
3666     NEXT,    J/LOAD020A                  1GO LOAD PATTERN
(5733) DCS(1.00,1.0,0,0) BM(1011..00.11..11.00..001..010..0,0,0..0,0..0,0000...0,0000,0...11,000...111,010,100)
3667
3668
3669 5724:
3670 LOAD020A:
3671     P2-U,    IR-EMIT,                    1LOAD IR WITH TEST PATTERN
3672             EMIT/001257,                1(001257)
3673     NEXT,    J/GOBUTO20A                1GO SETUP FOR "BUT"
(5724) DCS(0.00,0.0,0,0) BM(0000..00.00..10,10..101..111..0,0,0..0,0..1,1010...0,0000,0...11,000...000,100,001)
3674
3675
3676 5041: 1(FREE)
3677 GOBUTO20A:
3678     SETUP,   RETURN/TEST020B,            1RETURN TO START OF NEXT SUBTEST
3679     NEXT,    GOTO-PAGE(7),              1BUT TABLE IS ON PAGE 7
3680     J/BUTIR9=4                          1GO DO "BUT" ON IR<916>H
(5041) DCS(0.00,0.0,0,0) BM(0101..00.11..10,11..001..111..0,0,0..0,0..0,0000...0,0000,0...11,100...011,000,100)
3681
3682
3683 1 - - - - -
3684
3685 1* PART B *
3686 1TEST-020-B CHECKS THAT BUT[MOV-DR7:IR<513>] READS THE -(-FLTPY+MOV+FLTPY+DR7) AND
3687 1ALTERNATING PATTERN "101" IN IR<513>H CORRECTLY
3688 5731:
3689 TEST020B:
3690     PO,      LOAD-ENUA(ZTARGET405),      1LOAD EXPECTED ADDRESS AFTER "BUT"
3691             LOAD-ERROR(TEST020B),        1ERROR DIRECTORY KEY
3692             DCS-CTR(C3.),                1COMPARE ENUA:TNVA IN 3, MICROWORDS
3693     NEXT,    J/GOBUTO20B                1GO SETUP FOR "BUT"
(5731) DCS(1.00,1.0,0,0) BM(1100..00.11..11.00..000..101..0,0,0..0,0..0,0000...0,0000,0...11,000...000,100,010)
3694
3695
3696 5042: 1(FREE)
3697 GOBUTO20B:
3698     SETUP,   RETURN/TEST020C,            1RETURN TO START OF NEXT SUBTEST
3699     PO,      BUMP-VERIFY,                1COUNT
3700     NEXT,    GOTO-PAGE(7),              1BUT TABLE IS ON PAGE 7
3701     J/BUTMOVDR7IRS=3                    1GO DO "BUT"
(5042) DCS(0.00,0.0,0,0) BM(0101..00.11..10,10..111..111..0,0,0..0,0..0,0000...0,0000,0...11,100...011,000,101)
```

```
3702
3703
3704 1 - - - - -
3705
3706 1* PART C *
3707 1TEST-020-C CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3708 1AS CLASS=F+BRANCH, IR<14111>H="0000", IR<15910108>H="0000",
3709 1AND TARGETS TO (757)
3710 5727:
3711 TEST020C:
3712     PO,      LOAD-ENUA(ZTARGET757),      1LOAD EXPECTED ADDRESS AFTER "BUT"
3713             LOAD-ERROR(TEST020C),        1ERROR DIRECTORY KEY
3714             DCS-CTR(C3.),                1COMPARE ENUA:TNVA IN 3, MICROWORDS
3715     NEXT,    J/GOBUTO20C                1GO SETUP FOR "BUT"
(5727) DCS(1.00,1.0,0,0) BM(1100..00.11..11.11..101..111..0,0,0..0,0..0,0000...0,0000,0...11,000...000,100,011)
3716
3717
3718 5043: 1(FREE)
3719 GOBUTO20C:
3720     SETUP,   RETURN/SCOPE020,            1RETURN TO SCOPE LOOP TEST WORD
3721     NEXT,    GOTO-PAGE(7),              1BUT TABLE IS ON PAGE 7
3722     J/RUTINSTR1                          1GO DO INSTR1 "BUT"
(5043) DCS(0.00,0.0,0,0) BM(0101..00.00..01,00..100..111..0,0,0..0,0..0,0000...0,0000,0...11,100...011,000,110)
3723
3724
3725 5044: 1(FREE)
3726 SCOPE020:
3727     NEXT,    RUTDISCOPE,                1NO ERROR: "TEST021A" [+1, WPD]
3728     J/TEST021A                          1 ERROR: "LOAD020A" [-6, WORDS]
(5044) DCS(0.00,0.1,0,0) BM(0000..00.00..00,00..000..000..0,0,0..0,0..0,0000...0,0000,0...11,000...111,010,101)
3729
3730
3731
3732
3733
3734
3735 1 - - - - -
3736
3737 1*** TEST 021 ***
3738 1TEST-021 USES A DATA PATTERN OF: "0 000 010 011 001 101" (002315)
3739
3740 1 - - - - -
3741
3742 1* PART A *
3743 1TEST-021-A CHECKS THAT BUT[DM0#SM0#BYTE],
3744 1DM=IR<513>H="001", DM0H=0, SM=IR<1119>H="010", SM0H=0, BYTE H=0
3745
3746 5725:
3747 TEST021A:
3748     PO,      LOAD-ENUA(ZTARGET400),      1LOAD EXPECTED ADDRESS AFTER "BUT"
3749             LOAD-ERROR(TEST021A),        1ERROR DIRECTORY KEY
3750             DCS-CTR(C4.),                1COMPARE ENUA:TNVA IN 4, MICROWORDS
3751     NEXT,    J/LOAD021A                  1GO LOAD PATTERN
(5725) DCS(1.00,1.0,0,0) BM(1011..00.11..11.00..000..000..0,0,0..0,0..0,0000...0,0000,0...11,000...111,010,000)
```

```
3751
3752
3753 5720:
3754 LOAD021A:
3755     PO,      BUMP-VERIFY,      ICOUNT
3756     P2-U,    IR_EMIT,          ILOAD IR WITH TEST PATTERN
3757     NEXT,    J/GOBUT021A,      I(002315)
3758     (5720) DCS[0,0,0,0,0,1] BM[0000,00,01,00,11,001,101,00,0,0,0,0,0,1,1010,00,0000,0,11,000,000,100,101]
3759
3760
3761 5045: 1(FREE)
3762 GOBUT021A:
3763     SETUP,   RETURN/TEST021B,  IRETURN TO START OF NEXT SUBTEST
3764     NEXT,    GOTO-PAGE(7),      IBUT TABLE IS ON PAGE 7
3765     (5045) DCS[0,0,0,0,0,0] BM[0101,00,11,10,10,011,111,00,0,0,0,0,0,0,0000,00,0000,0,11,100,011,001,011]
3766
3767
3768 1 - - - - -
3769
3770 1* PART B *
3771 ITEST-021-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3772 IAS CLASS-F=BRANCH, IR<14:11>H="0000", IR<15:10:08>H="0000",
3773 IAND TARGETS TO (757)
3774 5723:
3775 TEST021B:
3776     PO,      LOAD=ENUA(ZTARGET757), ILOAD EXPECTED ADDRESS AFTER "BUT"
3777     LOAD=ERROR(TEST021B),          IERROR DIRECTORY KEY
3778     DCS=CTR(C1),                  ICOMPARE ENUA:TWUA IN 3. MICROWORDS
3779     NEXT,    J/GOBUT021B,          IGO SETUP FOR "BUT"
3780     (5723) DCS[1,0,0,1,0,0,0] BM[1100,00,11,11,11,101,111,00,0,0,0,0,0,0,0000,00,0000,0,11,000,000,100,110]
3781
3782
3783 5046: 1(FREE)
3784 GOBUT021B:
3785     SETUP,   RETURN/TEST021C,  IRETURN TO START OF NEXT SUBTEST
3786     NEXT,    GOTO-PAGE(7),      IBUT TABLE IS ON PAGE 7
3787     (5046) DCS[0,0,0,0,0,0] BM[0101,00,11,11,11,100,111,00,0,0,0,0,0,0,0000,00,0000,0,11,100,011,000,110]
3788
3789
3790 1 - - - - -
3791
3792 1* PART C *
3793 ITEST-021-C CHECKS THAT BUT[IR<11>FL7PT<3:0>] READS THE "0" IN IR<11>H CORRECTLY,
3794 IAND THE FL7PT DFCODE ROW GETS ADDRESS (462), WHICH IS A ADDF/D MODE1-7 INSTR;
3795 IDATA OUTPUT SHOULD BE (11)
3796 5774:
3797 TEST021C:
3798     PO,      LOAD=ENUA(ZTARGET411), ILOAD EXPECTED ADDRESS AFTER "BUT"
```

```
3799     LOAD=ERROR(TEST021C),          IERROR DIRECTORY KEY
3800     DCS=CTR(C3),                  ICOMPARE ENUA:TWUA IN 3. MICROWORDS
3801     NEXT,    J/GOBUT021C,          IGO SETUP FOR "BUT"
3802     (5774) DCS[1,0,0,1,0,0,0] BM[1100,00,11,11,00,001,001,00,0,0,0,0,0,0,0000,00,0000,0,11,000,000,100,111]
3803
3804
3805 5047: 1(FREE)
3806 GOBUT021C:
3807     SETUP,   RETURN/SCOPE021,  IRETURN TO SCOPE LOOP TEST WORD
3808     NEXT,    GOTO-PAGE(7),      IBUT TABLE IS ON PAGE 7
3809     (5047) DCS[0,0,0,0,0,0,0] BM[0101,00,00,01,01,000,111,00,0,0,0,0,0,0,0000,00,0000,0,11,100,011,000,010]
3810
3811
3812 5050: 1(FREE)
3813 SCOPE021:
3814     NEXT,    BUT[SCOPE],          INO ERROR: "TEST022A" (+1, WORDS)
3815     J/TEST022A,                  I ERROR: "LOAD021A" (-6, WORDS)
3816     (5050) DCS[0,0,0,0,1,0,0] BM[0000,00,00,00,00,000,000,00,0,0,0,0,0,0,0000,00,0000,0,11,000,011,010,001]
3817
3818
3819
3820
3821 1 - - - - -
3822
3823 1*** TEST 022 ***
3824 ITEST-022 USES A DATA PATTERN OF: "0 000 011 000 110 011" (003063)
3825
3826 1 - - - - -
3827
3828 1* PART A *
3829 ITEST-022-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3830 IAS CLASS-F=BRANCH, IR<14:11>H="0000", IR<15:10:08>H="0000",
3831 IAND TARGETS TO (757)
3832 5721:
3833 TEST022A:
3834     PO,      LOAD=ENUA(ZTARGET757), ILOAD EXPECTED ADDRESS AFTER "BUT"
3835     LOAD=ERROR(TEST022A),          IERROR DIRECTORY KEY
3836     DCS=CTR(C4),                  ICOMPARE ENUA:TWUA IN 4. MICROWORDS
3837     NEXT,    J/LOAD022A,          IGO LOAD PATTERN
3838     (5721) DCS[1,0,0,1,0,0,0] BM[1011,00,11,11,11,101,111,00,0,0,0,0,0,0,0000,00,0000,0,11,000,011,001,110]
3839
3840
3841 5716:
3842 LOAD022A:
3843     P2-U,    IR_EMIT,          ILOAD IR WITH TEST PATTERN
3844     EMIT/003063,                  I(003063)
3845     NEXT,    J/GOBUT022A,          IGO SETUP FOR "BUT"
3846     (5716) DCS[0,0,0,0,0,0,0] BM[0000,00,01,10,00,110,011,00,0,0,0,0,0,0,1,1010,00,0000,0,11,000,000,101,001]
```

```
3847 5051: 1(FREE)
3848 GOBUT022A:
3849 SFTUP, RETURN/SCOPE022,      !RETURN TO SCOPE LOOP TEST WORD
3850 NEXT, GOTO=PAGE(7),          !BUT TABLE IS ON PAGE 7
3851 J/BUTINSTR1                  !GO DO INSTR1 "BUT"
(5051) DCS(0.00,0.0,0.0) BM(0101..00.00..01.01..010..111...0.0,0..0..0.0000...0..0000,0...11.100...011.000,110)
3852
3853
3854
3855 5052: 1(FREE)
3856 SCOPE022:
3857 NEXT, BUTD[SCOPE],          !NO ERROR: "TEST023A" [+1, WORD]
3858 J/TEST023A                  ! ERROR: "LOAD023A" [-2, WORDS]
(5052) DCS(0.00,0.0,0.0) BM(0000..00.00..00.00..000..000...0.0,0..0..0.0000...0..0000,0...11.000...111.001,111)
3859
3860
3861
3862
3863
3864
3865 |-----|
3866
3867 !*** TEST 023 ***
3868 !TEST-023 USES A DATA PATTERN OF: "1 000 110 000 000 100" (106004)
3869
3870 |-----|
3871
3872 !* PART A *
3873 !TEST-023-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3874 !AS CLASS=0=SDP=DM0; IR<14:06>H="000 110 000"; IR<15>H="1";
3875 !DM=IR<5:3>H="000", DM0H=1; AND TARGETS TO (560)
3876 5717:
3877 TEST023A:
3878 P0, LOAD=ENUA(ZTARGET560),      !LOAD EXPECTED ADDRESS AFTER "BUT"
3879 LOAD=ERROR(TEST023A),          !ERROR DIRECTORY KEY
3880 DCS=CTR(C4.),                 !COMPARE ENUA:ENUA IN 4, MICROWORDS
3881 NEXT, J/LOAD023A              !GO LOAD PATTERN
(5717) DCS(1.00,1.0,0.0) BM(1011..00.11..11.01..110..000...0.0,0..0..0.0000...0..0000,0...11.000...111.001,100)
3882
3883
3884 5714:
3885 LOAD023A:
3886 P2=U, IR_EMIT,              !LOAD IR WITH TEST PATTERN
3887 EMIT/106004,                 ! (106004)
3888 NEXT, J/GOBUT023A            !GO SETUP FOR "BUT"
(5714) DCS(0.00,0.0,0.0) BM(1000..00.11..00.00..000..100...0.0,0..0..0.0000...1.1010...0..0000,0...11.000...000.101,011)
3889
3890
3891 5053: 1(FREE)
3892 GOBUT023A:
3893 SFTUP, RETURN/SCOPE023,      !RETURN TO SCOPE LOOP TEST WORD
3894 NEXT, GOTO=PAGE(7),          !BUT TABLE IS ON PAGE 7
3895 J/BUTINSTR1                  !GO DO INSTR1 "BUT"
```

```
(5053) DCS(0.00,0.0,0.0) BM(0101..00.00..01.01..100..111...0.0,0..0..0.0000...0..0000,0...11.100...011.000,110)
3896
3897
3898 5054: 1(FREE)
3899 SCOPE023:
3900 P0, RIMP-VERIFY,             !COUNT
3901 NEXT, BUTD[SCOPE],          !NO ERROR: "TEST024A" [+1, WORD]
3902 J/TEST024A                  ! ERROR: "LOAD023A" [-2, WORDS]
(5054) DCS(0.00,0.1,0.1) BM(0000..00.00..00.00..000..000...0.0,0..0..0.0000...0..0000,0...11.000...111.001,101)
3903
3904
3905
3906
3907
3908
3909 |-----|
3910
3911 !*** TEST 024 ***
3912 !TEST-024 USES A DATA PATTERN OF: "0 000 101 111 000 110" (005706)
3913
3914 |-----|
3915
3916 !* PART A *
3917 !TEST-024-A CHECKS THAT BUT(DR6-7 L) READS THE "11" IN IR<2:1> H CORRECTLY
3918 !AND DOES ASSERT THE SIGNAL
3919 5715:
3920 TEST024A:
3921 P0, LOAD=ENUA(ZTARGET402),      !LOAD EXPECTED ADDRESS AFTER "BUT"
3922 LOAD=ERROR(TEST024A),          !ERROR DIRECTORY KEY
3923 DCS=CTR(C4.),                 !COMPARE ENUA:ENUA IN 4, MICROWORDS
3924 NEXT, J/LOAD024A              !
(5715) DCS(1.00,1.0,0.0) BM(1011..00.11..11.00..000..010...0.0,0..0..0.0000...0..0000,0...11.000...111.000,110)
3925
3926
3927 5706:
3928 LOAD024A:
3929 P0, RIMP-VERIFY,             !COUNT
3930 P2=U, IR_EMIT,              !LOAD IR WITH TEST PATTERN
3931 EMIT/005706,                 ! (005706)
3932 NEXT, J/GOBUT024A            !GO SETUP FOR "BUT"
(5706) DCS(0.00,0.0,0.1) BM(0000..00.10..11.11..000..110...0.0,0..0..0.0000...1.1010...0..0000,0...11.000...000.101,101)
3933
3934
3935 5055: 1(FREE)
3936 GOBUT024A:
3937 SFTUP, RETURN/TEST024B,      !RETURN TO START OF NEXT SUBTEST
3938 NEXT, GOTO=PAGE(7),          !BUT TABLE IS ON PAGE 7
3939 J/BUTDR6-7L                  !GO DO "BUT" ON DR 6/7 L
(5055) DCS(0.00,0.0,0.0) BM(0101..00.11..10.01..001..111...0.0,0..0..0.0000...0..0000,0...11.100...011.100,111)
3940
3941
3942 |-----|
```

```

3943 1* PART B *
3944 !TEST-024-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3945 !AS CLASS-D=SOP+DMO; I<14:06>H="000 101 111"; IR<15>H="0";
3946 !DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (557)
3947 57111
3948 TEST024B:
3949 PO, LOAD-ENUA(ZTARGETS57), !LOAD EXPECTED ADDRESS AFTER "BUT"
3950 LOAD-ERROR(TEST024B), !ERROR DIRECTORY KEY
3951 DCS-CTRC(3.), !COMPARE ENUA:TWUA IN 3, MICROWORDS
3952 J/GOBUTO24B !GO SETUP FOR "BUT"
3953 (5711) DCS[1,00,1,0,0,0] BM[1100,00,11,11,01,11,11,11,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,1000,0,000,101,110]
3954
3955 5056: 1(FREE)
3956 GORUTO24B:
3957 SETUP, RETURN/SCOPE024, !RETURN TO SCOPE LOOP TEST WORD
3958 PO, BUMP-VERIFY, !COUNT
3959 NEXT, GOTO-PAGE(7), !BUT TABLE IS ON PAGE 7
3960 J/BUTINSTR1 !GO DO INSTR1 "BUT"
3961 (5056) DCS[0,00,0,0,0,1] BM[0101,00,00,01,01,11,11,11,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,011,000,110]
3962
3963 5057: 1(FPEF)
3964 SCOPE024:
3965 NEXT, BUTD[SCOPE], !NO ERROR: "TEST025A" [+1, WORD]
3966 J/TEST025A ! ERROR: "LOAD024A" [-4, WORDS]
3967 (5057) DCS[0,00,0,1,0,0] BM[0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,1000,0,111,000,111]
3968
3969
3970
3971
3972
3973
3974
3975
3976 1** TEST 025 **
3977 !TEST-025 USES A DATA PATTERN OF: "0 000 110 011 000 011" (006303)
3978
3979
3980
3981
3982 1* PART A *
3983 !TEST-025-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
3984 !AS CLASS-D=SOP+DMO; IR<14:06>H="000 110 011"; IR<15>H="0";
3985 !DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (543)
3986 57071
3987 TEST025A:
3988 PO, LOAD-ENUA(ZTARGETS43), !LOAD EXPECTED ADDRESS AFTER "BUT"
3989 LOAD-ERROR(TEST025A), !ERROR DIRECTORY KEY
3990 DCS-CTRC(4.), !COMPARE ENUA:TWUA IN 4, MICROWORDS
3991 NEXT, J/LOAD025A !GO LOAD PATTERN
3992 (5707) DCS[1,00,1,0,0,0] BM[1011,00,11,11,01,100,0,011,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,1000,0,111,000,100]
3993

```

```

3992      57041:
3993      LOAD025A:
3994      P2=U,      IR_EMIT,      !LOAD IR WITH TEST PATTERN
3995      EMIT/006303,      !006303)
3996      J/G0BUT025A      !GO SETUP FOR "BUT"
3997      WFTX,      J/RJUTINSTR1
(5704) DCS[0,00,0,0,0,0] BM[0000,00,11,00,11,000,011,00,0,0,0,0,0,0,1,1010,00,0,000,0,0,11,000,000,110,000]

3999
4000      5060: 1(FREE)
4001      G0BUT025A:
4002      SETUP,      RETURN/SCOPE025,      !RETURN TO SCOPE LOOP TEST WORD
4003      NEXT,      GOTO-PAGE(7),      !BUT TABLE IS ON PAGE 7
4004      J/RJUTINSTR1      !GO DO INSTR1 "BUT"
(5060) DCS[0,00,0,0,0,0] BM[1010,00,00,01,10,001,111,00,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,00,011,000,110]

4005
4006      5061: 1(FREF)
4007      SCOP025:
4008      WFTX,      BUT0[SCOPE],      !NO ERROR: "TEST026A" [+1, WORD]
4009      J/TEST026A      ! ERROR: "LOAD025A" [-2, WORDS]
4010      DCS[0,00,0,1,0,0] BM[0000,00,00,00,00,000,000,00,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,111,000,101]
(5061) DCS[0,00,0,1,0,0] BM[0000,00,00,00,00,000,000,00,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,111,000,101]

4011
4012
4013
4014
4015
4016
4017      ! - - - - -
4018
4019      !*** TEST 026 ***
4020      !TFST-026 "USES A DATA PATTERN OF: "1 000 101 100 000 011" (105403)
4021
4022      ! - - - - -
4023
4024      !* PART A *
4025      !TFST-026-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4026      !AS CLASS=0=80P=DM0, IR<14|06>H="000 101 100"; IR<15>H="1";
4027      !DM=IR<5|3>H="000", DM0H=1; AND TARGETS TO (574)
4028      5705:
4029      TEST026A:
4030      P0,      LOAD=ENUA(ZTARGETS74),      !LOAD EXPECTED ADDRESS AFTER "BUT"
4031      LOAD=ERROR(TEST026A),      !ERROR DIRECTORY KEY
4032      DCS=CTR(C4,      !COMPARE ENUA;INUA IN 4, MICROWORDS
4033      BUMP=VERIFY,      !COUNT
4034      NEXT,      J/LOAD026A      !GO LOAD PATTERN
(5705) DCS[1,00,1,0,0,1] BM[1011,00,01,11,01,111,100,00,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,111,000,010]

4035
4036
4037      5702:
4038      LOAD026A:
4039      P2=U,      IR_EMIT,      !LOAD IR WITH TEST PATTERN
4040      EMIT/105403,      !105403)

```

```
4041 NEXT, J/GOBUT026A IGO SETUP FOR "BUT"
(5702) DCS(0,00,0,0,0,0) BM(1000,00,10,11,00,000,011,000,0,0,0,0,1,1010,00,0,0000,0,11,000,000,110,010)
4042
4043
4044 5062: 1(FREE)
4045 GOBUT026A:
4046 SETUP, RETURN/SCOPE026, IRETURN TO SCOPE LOOP TEST WORD
4047 PO, BUMP-VERIFY, ICOUNT
4048 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4049 J/BUTINSTR1 IGO DO INSTR1 "BUT"
(5062) DCS(0,00,0,0,0,1) BM(0101,00,00,01,10,011,111,000,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,110)
4050
4051
4052 5063: 1(FREE)
4053 SCOPE026:
4054 NEXT, BUTD[SCOPE], INO ERROR: "TEST027A" [+1, WORD]
4055 I ERROR: "LOAD027A" [-2, WORDS]
(5063) DCS(0,00,0,1,0,0) BM(0000,00,00,00,00,000,000,0,0,0,0,0,0,0000,0,0,0000,0,11,000,111,000,011)
4056
4057
4058
4059
4060
4061
4062 | - - - - -
4063
4064 I*** TEST 027 ***
4065 ITEST-027 USES A DATA PATTERN OF: "1 000 110 001 000 010" (106102)
4066
4067 | - - - - -
4068
4069 I* PART A *
4070 ITEST-027-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4071 IAS CLASS=D=SOP*DMO; IR<14:06>H="000 110 001"; IR<15>H="1";
4072 IDH=IR<5:3>H="000", DM0H=1; AND TARGETS TO (561)
4073 5703:
4074 TEST027A:
4075 PO, LOAD-ENUA(2TARGET561), ILOAD EXPECTED ADDRESS AFTER "BUT"
4076 LOAD-ERROR(TEST027A), IERROR DIRECTORY KEY
4077 DCS-CTR(C4,), ICOMPARE ENUA:TNUA IN 4, MICROWORDS
4078 NFXT, J/LOAD027A IGO LOAD PATTERN
(5703) DCS(1,00,1,0,0,0) BM(1011,00,11,11,01,110,001,000,0,0,0,0,0,0000,0,0,0000,0,11,000,111,000,000)
4079
4080
4081 5700:
4082 LOAD027A:
4083 P2-U, IR_EMIT, ILOAD IR WITH TEST PATTERN
4084 EMIT/106102, I(106102)
4085 NEXT, J/GOBUT027A IGO SETUP FOR "BUT"
(5700) DCS(0,00,0,0,0,0) BM(1000,00,11,00,01,000,010,000,0,0,0,0,0,1,1010,00,0,0000,0,11,000,000,110,100)
4086
4087
4088 5064: 1(FREE)
```

```
4089 GOBUT027A:
4090 SETUP, RETURN/SCOPE027, IRETURN TO SCOPE LOOP TEST WORD
4091 NFXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4092 J/BUTINSTR1 IGO DO INSTR1 "BUT"
(5064) DCS(0,00,0,0,0,0) BM(0101,00,00,01,10,101,111,000,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,110)
4093
4094 5065: 1(FREE)
4095 SCOPE027:
4096 PO, RUMP-VERIFY, ICOUNT
4097 NEXT, BUTD[SCOPE], INO ERROR: "TEST030A" [+1, WORD]
4098 I ERROR: "LOAD027A" [-2, WORDS]
(5065) DCS(0,00,0,1,0,1) BM(0000,00,00,00,00,000,000,0,0,0,0,0,0,0000,0,0,0000,0,11,000,111,000,001)
4099
4100
4101
4102
4103
4104
4105 | - - - - -
4106
4107 I*** TEST 030 ***
4108 ITEST-030 USES A DATA PATTERN OF: "0 000 101 010 000 100" (005204)
4109
4110 | - - - - -
4111
4112 I* PART A *
4113 ITEST-030-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4114 IAS CLASS=D=SOP*DMO; IR<14:06>H="000 101 010"; IR<15>H="0";
4115 IDH=IR<5:3>H="000", DM0H=1; AND TARGETS TO (552)
4116 5701:
4117 TEST030A:
4118 PO, LOAD-ENUA(2TARGET552), ILOAD EXPECTED ADDRESS AFTER "BUT"
4119 LOAD-ERROR(TEST030A), IERROR DIRECTORY KEY
4120 DCS-CTR(C4,), ICOMPARE ENUA:TNUA IN 4, MICROWORDS
4121 NFXT, J/LOAD030A IGO LOAD PATTERN
(5701) DCS(1,00,1,0,0,0) BM(1011,00,11,11,01,101,010,000,0,0,0,0,0,0000,0,0,0000,0,11,000,110,111,110)
4122
4123
4124 5676:
4125 LOAD030A:
4126 P2-U, IR_EMIT, ILOAD IR WITH TEST PATTERN
4127 EMIT/005204, I(005204)
4128 NEXT, J/GOBUT030A IGO SETUP FOR "BUT"
(5676) DCS(0,00,0,0,0,0) BM(0000,00,10,10,10,000,100,000,0,0,0,0,0,1,1010,00,0,0000,0,11,000,000,110,110)
4129
4130
4131 5066: 1(FREE)
4132 GOBUT030A:
4133 SETUP, RETURN/TEST030A, IRETURN TO START OF NEXT SUBTEST
4134 NFXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4135 J/BUTINSTR1 IGO DO INSTR1 "BUT"
(5066) DCS(0,00,0,0,0,0) BM(0101,00,11,10,01,010,111,000,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,110)
```

```
4136
4137
4138
4139
4140 | - - - - -
4141
4142 |* PART B *
4143 |TEST-030-B CHECKS THAT BUT[IR<11>:FLTPT<3:0>] READS THE "1" IN IR<11>H CORRECTLY,
4144 |AND THE FLTPT DECODE ROM GETS ADDRESS (240), WHICH IS A MULPLYMODE-0 INSTR.
4145 |DATA SHOULD BE (04)
4146 5712:
4147 TEST030B:
4148     PO,          LOAD-ENUA(ZTARGET424),          |LOAD EXPECTED ADDRESS AFTER "BUT"
4149                LOAD-ERROR(TEST030B),          |ERROR DIRECTORY KEY
4150                DCS-CTR(C3.),          |COMPARE ENUA:TNVA IN 3, MICROWORDS
4151                NEXT, JGOBUT030B          |
4152 (5712) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.010.100.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.000.110.111]
4153
4154 5067: 1(FREE)
4155 GORUTO30B:
4156     SETUP,      RETURN/TEST030C,          |RETURN TO START OF NEXT SUBTEST
4157     NEXT,       GOTO-PAGE(7),          |BUT TABLE IS ON PAGE 7
4158                J/BUTIR11FLTPT3=0          |GO DO "BUT" ON IR<11>H:FLTPT<3:0>H
4159 (5067) DCS[0.00.0.0.0.0] BM[0101.00.11.10.01.011.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011.000.010]
4160
4161 | - - - - -
4162
4163 |* PART C *
4164 |TEST-030-C CHECKS THAT BUTR[IR<11>B] READS THE "1" IN IR<11>H CORRECTLY
4165 5713:
4166 TEST030C:
4167     PO,          LOAD-ENUA(ZTARGET403),          |LOAD EXPECTED ADDRESS AFTER "BUT"
4168                LOAD-ERROR(TEST030C),          |ERROR DIRECTORY KEY
4169                DCS-CTR(C3.),          |COMPARE ENUA:TNVA IN 3, MICROWORDS
4170                NEXT, JGOBUT030C          |GO SETUP FOR "BUT"
4171 (5713) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.000.011.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.000.111.000]
4172
4173 5070: 1(FREE)
4174 GORUTO30C:
4175     SETUP,      RETURN/TEST030D,          |RETURN TO START OF NEXT SUBTEST
4176     NEXT,       GOTO-PAGE(7),          |BUT TABLE IS ON PAGE 7
4177                J/BUTIR11B          |GO DO "BUT" ON IR<11>H
4178 (5070) DCS[0.00.0.0.0.0] BM[0101.00.11.10.11.100.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011.100.010]
4179
4180
4181 | - - - - -
4182
4183
```

```
4184 |* PART D *
4185 |TEST-030-D CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
4186 |AS ROM ADDRESS=(452) ON THE INSTR5 E78 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
4187 |OF (05), TARGETING TO (405) AFTER THE DECODE
4188 5734:
4189 TEST030D:
4190     PO,          LOAD-ENUA(ZTARGET405),          |LOAD EXPECTED ADDRESS AFTER "BUT"
4191                LOAD-ERROR(TEST030D),          |ERROR DIRECTORY KEY
4192                DCS-CTR(C3.),          |COMPARE ENUA:TNVA IN 3, MICROWORDS
4193                NEXT, JGOBUT030D          |GO SETUP FOR "BUT"
4194 (5734) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.000.101.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.000.111.001]
4195
4196 5071: 1(FREE)
4197 GORUTO30D:
4198     SETUP,      RETURN/SCOPE030,          |RETURN TO SCOPE LOOP TEST WORD
4199     NEXT,       GOTO-PAGE(7),          |BUT TABLE IS ON PAGE 7
4200                J/BUTINSTR5          |GO DO "BUT" ON INSTR5<4:0>H
4201 (5071) DCS[0.00.0.0.0.0] BM[0101.00.00.01.11.010.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011.000.001]
4202
4203 5072: 1(FREE)
4204 SCOPE030:
4205     NEXT,       RUT[SCOPE],          |NO ERROR: "TEST031A" [+1, WORD]
4206                J/TEST031A          |ERROR: "LOAD030A" [-8, WORDS]
4207 (5072) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.110.111.111]
4208
4209
4210
4211
4212
4213 | - - - - -
4214
4215 |*** TEST 031 ***
4216 |TEST-031 USES A DATA PATTERN OF: "0 001 000 010 100 010" (040212)
4217
4218 | - - - - -
4219
4220 |* PART A *
4221 |TEST-031-A CHECKS THAT BUT[DM0:SM0:RYTE]
4222 |DM=IR<5:3>H="100", DM0H=0, SM=IR<11:9>H="000", SM0H=1, BYTE H=0
4223 5677:
4224 TEST031A:
4225     PO,          LOAD-ENUA(ZTARGET402),          |LOAD EXPECTED ADDRESS AFTER "BUT"
4226                LOAD-ERROR(TEST031A),          |ERROR DIRECTORY KEY
4227                DCS-CTR(C4.),          |COMPARE ENUA:TNVA IN 4, MICROWORDS
4228                NEXT, J/LOAD031A          |GO LOAD PATTERN
4229 (5677) DCS[1.00.1.0.0.0] BM[0111.00.11.11.00.000.010.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.110.111.010]
4230
4231 5672:
4232 LOAD031A:
```

```
4233      PO,      BUMP=VERIFY,      ICOUNT
4234      P2=U,      IR_EMIT,      ILOAD IR WITH TEST PATTERN
4235      NEXT,      EMIT/010242,      I(010242)
4236      J/GOBUT031A      IGO SETUP FOR "BUT"
(5672) DCS[0.00,0.0,0.1] BM[0001..00.00..00.10..100..010..0.0,0.0..0.0...1.1010...0..0000.0...11.000...000.111.011]
4237
4238
4239      5073: I(FREE)
4240      GOBUT031A:
4241      SETUP,      RETURN/TEST031B,      IRETURN TO START OF NEXT SUBTEST
4242      NEXT,      GOTO-PAGE(7),      IBUT TABLE IS ON PAGE 7
4243      J/BUTDMSMRYTE      IGO DO "BUT"
(5073) DCS[0.00,0.0,0.0] BM[0101..00.11..01.11..101..111...0.0,0.0..0.0...0.0000...0..0000.0...11.100...011.001.011]
4244
4245
4246      I - - - - -
4247
4248      I* PART B *
4249      ITEST-031-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4250      IAS CLASS=C=MOV*SMO*DMO; IR<14:9>H="001 000"; IR<15>H="0";
4251      IDM=IR<5:3>H="100"; AND TARGETS TO (604)
4252      5675:
4253      TEST031B:
4254      PO,      LOAD-ENUA(ZTARGET604),      ILOAD EXPECTED ADDRESS AFTER "BUT"
4255      LOAD-ERROR(TEST031B),      IERROR DIRECTORY KEY
4256      DCS-CTR(C3),      ICOMPARE ENUA:TNUA IN 3, MICROWORDS
4257      NFXT,      J/GOBUT031B      IGO SETUP FOR "BUT"
(5675) DCS[1.00,1.0,0.0] BM[1100..00.11..11.10..000..100...0.0,0.0..0.0...0.0000...0..0000.0...11.000...000.111.100]
4258
4259
4260      5074: I(FREE)
4261      GOBUT031B:
4262      SETUP,      RETURN/SCOPE031,      IRETURN TO SCOPE LOOP TEST WORD
4263      NEXT,      GOTO-PAGE(7),      IBUT TABLE IS ON PAGE 7
4264      J/BUTINSTR1      IGO DO INSTR1 "BUT"
(5074) DCS[0.00,0.0,0.0] BM[0101..00.00..01.11..101..111...0.0,0.0..0.0...0.0000...0..0000.0...11.100...011.000.110]
4265
4266
4267      5075: I(FREE)
4268      SCOPE031:
4269      NEXT,      BUTD[SCOPE],      INO ERROR; "TEST032A" [+1, WORDS]
4270      J/TEST032A      I ERROR; "LOAD031A" [-4, WORDS]
(5075) DCS[0.00,0.1,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0..0.0...0.0000...0..0000.0...11.000...110.111.011]
4271
4272
4273
4274
4275
4276
4277      I - - - - -
4278
4279      I*** TEST 032 ***
```

```
4280      ITEST-032 USES A DATA PATTERN OF: "1 001 000 001 010 101" (110125)
4281
4282      I - - - - -
4283
4284      I* PART A *
4285      ITEST-032-A CHECKS THAT BUT[MOV-DR7:IR<8:3>] READS THE (-FLTPT=MOV+FLTPT=DR7) AND
4286      IALTERNATING PATTERN "010" IN IR<5:3>H CORRECTLY
4287      5673:
4288      TEST032A:
4289      PO,      LOAD-ENUA(ZTARGET412),      ILOAD EXPECTED ADDRESS AFTER "BUT"
4290      LOAD-ERROR(TEST032A),      IERROR DIRECTORY KEY
4291      DCS-CTR(C4),      ICOMPARE ENUA:TNUA IN 4, MICROWORDS
4292      BUMP=VERIFY,      ICOUNT
4293      NFXT,      J/LOAD032A      IGO LOAD PATTERN
(5673) DCS[1.00,1.0,0.1] BM[1011..00.11..11.00..001..010...0.0,0.0..0.0...0.0000...0..0000.0...11.000...110.110.110]
4294
4295
4296      5666:
4297      LOAD032A:
4298      P2=U,      IR_EMIT,      ILOAD IR WITH TEST PATTERN
4299      EMIT/110125,      I(110125)
4300      NEXT,      J/GOBUT032A      IGO SETUP FOR "BUT"
(5666) DCS[0.00,0.0,0.0] BM[1001..00.00..00.01..010..101...0.0,0.0..0.0...1.1010...0..0000.0...11.000...000.111.110]
4301
4302
4303      5076: I(FREE)
4304      GOBUT032A:
4305      SETUP,      RETURN/TEST032B,      IRETURN TO START OF NEXT SUBTEST
4306      NEXT,      GOTO-PAGE(7),      IBUT TABLE IS ON PAGE 7
4307      J/BUTMOVDR7IRS=3      IGO DO "BUT"
(5076) DCS[0.00,0.0,0.0] BM[0101..00.11..01.11..001..111...0.0,0.0..0.0...0.0000...0..0000.0...11.100...011.000.101]
4308
4309
4310      I - - - - -
4311
4312      I* PART B *
4313      ITEST-032-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4314      IAS CLASS=C=MOV*SMO*DMO; IR<14:9>H="001 000"; IR<15>H="1";
4315      IDM=IR<5:3>H="010"; AND TARGETS TO (612)
4316      5671:
4317      TEST032B:
4318      PO,      LOAD-ENUA(ZTARGET612),      ILOAD EXPECTED ADDRESS AFTER "BUT"
4319      LOAD-ERROR(TEST032B),      IERROR DIRECTORY KEY
4320      DCS-CTR(C3),      ICOMPARE ENUA:TNUA IN 3, MICROWORDS
4321      NFXT,      J/GOBUT032B      IGO SETUP FOR "BUT"
(5671) DCS[1.00,1.0,0.0] BM[1100..00.11..11.10..001..010...0.0,0.0..0.0...0.0000...0..0000.0...11.000...000.111.111]
4322
4323
4324      5077: I(FREE)
4325      GOBUT032B:
4326      SETUP,      RETURN/SCOPE032,      IRETURN TO SCOPE LOOP TEST WORD
4327      PO,      BUMP=VERIFY,      ICOUNT
4328      NFXT,      GOTO-PAGE(7),      IBUT TABLE IS ON PAGE 7
```

```
4329      J/BUTINSTR1      ;GO DO INSTR1 "BUT"
(5077) DCS[0,00,0,0,0,1] BM[0101,00,00,10,00,000,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,001,000,110]
4330
4331
4332      5100: I(FREE)
4333      SCOPE033:
4334      NEXT,      BUTD(SCOPE),      ;NO ERROR: "TEST033A" [+1, WORDS]
4335      J/TEST033A      ; ERROR: "LOAD033A" [-4, WORDS]
(5100) DCS[0,00,0,1,0,0] BM[0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,110,110,111]
4336
4337
4338
4339
4340
4341
4342      I - - - - -
4343
4344      ;*** TEST 033 ***
4345      ;TEST-033 USES A DATA PATTERN OF: "1 000 110 010 111 100" (106274)
4346
4347      I - - - - -
4348
4349      ;* PART A *
4350      ;TEST-033-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4351      ;AS CLASS-B=SOP+DM0; IR<14108>H="000 110 0";
4352      ;DM=TR<5:3>H="111", DM0H=0; AND TARGETS TO (S17)
4353      5667:
4354      TEST033A:
4355      PO,      LOAD-ENUA(ZTARGETS17),      ;LOAD EXPECTED ADDRESS AFTER "BUT"
4356      LOAD-ERROR(TEST033A),      ;ERROR DIRECTORY KEY
4357      DCS-CTR(C4,),      ;COMPARE ENUA;TWUA IN 4, MICROWORDS
4358      BUMP-VERIFY,      ;COUNT
4359      NEXT,      J/LOAD033A      ;GO LOAD PATTERN
(5667) DCS[1,00,1,0,0,1] BM[1011,00,11,11,01,001,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,110,110,100]
4360
4361
4362      5664:
4363      LOAD033A:
4364      P2=H,      IR_EMIT,      ;LOAD IR WITH TEST PATTERN
4365      EMIT/106274,      ;(106274)
4366      NEXT,      J/GOBUT033A      ;GO SETUP FOR "BUT"
(5664) DCS[0,00,0,0,0,0] BM[1000,00,11,00,10,111,100,0,0,0,0,0,0,1,1010,0,0,0000,0,11,000,001,000,001]
4367
4368
4369      5101: I(FREE)
4370      GOBUT033A:
4371      SETUP,      RETURN/SCOPE033,      ;RETURN TO SCOPE LOOP TEST WORD
4372      PO,      BUMP-VERIFY,      ;COUNT
4373      NEXT,      GOTO-PAGE(7),      ;BUT TABLE IS ON PAGE 7
4374      J/BUTINSTR1      ;GO DO INSTR1 "BUT"
(5101) DCS[0,00,0,0,0,1] BM[0101,00,00,10,00,010,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,001,000,110]
4375
```

```
4376
4377      5102: I(FREE)
4378      SCOPE033:
4379      NEXT,      BUTD(SCOPE),      ;NO ERROR: "TEST034A" [+1, WORD]
4380      J/TEST034A      ; ERROR: "LOAD033A" [-2, WORDS]
(5102) DCS[0,00,0,1,0,0] BM[0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,110,110,101]
4381
4382
4383
4384
4385
4386      I - - - - -
4387
4388      ;*** TEST 034 ***
4389      ;TEST-034 USES A DATA PATTERN OF: "0 000 101 001 001 010" (005112)
4390
4391      I - - - - -
4392
4393      ;* PART A *
4394      ;TEST-034-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4395      ;AS CLASS-B=SOP+DM0; IR<14109>H="000 101";
4396      ;DM=TR<5:3>H="001", DM0H=0; AND TARGETS TO (S11)
4397      5665:
4398      TEST034A:
4399      PO,      LOAD-ENUA(ZTARGETS11),      ;LOAD EXPECTED ADDRESS AFTER "BUT"
4400      LOAD-ERROR(TEST034A),      ;ERROR DIRECTORY KEY
4401      DCS-CTR(C4,),      ;COMPARE ENUA;TWUA IN 4, MICROWORDS
4402      NEXT,      J/LOAD034A      ;GO LOAD PATTERN
(5665) DCS[1,00,1,0,0,0] BM[1011,00,11,11,01,001,001,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,110,110,010]
4403
4404
4405      5662:
4406      LOAD034A:
4407      PO,      BUMP-VERIFY,      ;COUNT
4408      P2=H,      IP_EMIT,      ;LOAD IR WITH TEST PATTERN
4409      EMIT/005112,      ;(005112)
4410      NEXT,      J/GOBUT034A      ;GO SETUP FOR "BUT"
(5662) DCS[0,00,0,0,0,1] BM[0000,00,10,10,01,001,010,0,0,0,0,0,0,1,1010,0,0,0000,0,11,000,001,000,011]
4411
4412
4413      5103: I(FREE)
4414      GOBUT034A:
4415      SETUP,      RETURN/SCOPE034,      ;RETURN TO SCOPE LOOP TEST WORD
4416      NEXT,      GOTO-PAGE(7),      ;BUT TABLE IS ON PAGE 7
4417      J/BUTINSTR1      ;GO DO INSTR1 "BUT"
(5103) DCS[0,00,0,0,0,0] BM[0101,00,00,10,00,100,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,001,000,110]
4418
4419
4420      5104: I(FREE)
4421      SCOPE034:
4422      NEXT,      BUTD(SCOPE),      ;NO ERROR: "TEST035A" [+1, WORD]
4423      J/TEST035A      ; ERROR: "LOAD034A" [-2, WORDS]
(5104) DCS[0,00,0,1,0,0] BM[0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,110,110,011]
```

```
4424
4425
4426
4427
4428
4429
4430
4431
4432
4433
4434
4435
4436
4437
4438
4439
4440
4441
4442
4443
4444
4445
4446
4447
4448
4449
4450
4451
4452
4453
4454
4455
4456
4457
4458
4459
4460
4461
4462
4463
4464
4465
4466
4467
4468
4469
4470
4471
4472
```

*** TEST 035 ***

TEST-035 USES A DATA PATTERN OF: "0 011 000 111 000 001" (030701)

1* PART A *

TEST-035-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY

1AS CLASS=A=DOP*SMO*DMO; IR<15:12>H="0011";

1SM=IR<11:9>H="000", SMOH=1; DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (443)

5663:

TEST035A:

PO, LOAD=ENUA(ZTARGET443), ;LOAD EXPECTED ADDRESS AFTER "BUT"

LOAD=ERROR(TEST035A), ;ERROR DIRECTORY KEY

DCS=CTR(C4.), ;COMPARE ENUA:THUA IN 4. MICROWORDS

NEXT, J/LOAD035A ;GO LOAD PATTERN

(5663) DCS[1.00,1.0,0.0] BM[1011.00.11.11.00.100.011...0.0,0.0,0.0...0.0000...0.0000,0...11.000...110.110.000]

5660:

LOAD035A:

P2=U, IR_EMIT, ;LOAD IR WITH TEST PATTERN

EMIT/030701, ;(030701)

NEXT, J/GOBUTO35A ;GO SETUP FOR "BUT"

(5660) DCS[0.00,0.0,0.0] BM[0011.00.00.01.11.000.001...0.0,0.0,0.0...1.1010...0.0000,0...11.000...001.000.101]

5105: 1(FREE)

GOBUTO35A:

SETUP, RETURN/SCOPE035, ;RETURN TO SCOPE LOOP TEST WORD

PO, BUMP-VERIFY, ;ICOUNT

NEXT, GOTO-PAGE(7), ;BUT TABLE IS ON PAGE 7

J/BUTINSTR1 ;GO DO INSTR1 "BUT"

(5105) DCS[0.00,0.0,0.1] BM[0101.00.00.10.00.110.111...0.0,0.0,0.0...0.0000...0.0000,0...11.100...011.000.110]

5106: 1(FREF)

SCOPE035:

NEXT, BUTD[SCOPE], ;NO ERROR: "TEST036A" (+1, WORD)

J/TEST036A ;ERROR: "LOAD035A" (-2, WORDS)

(5106) DCS[0.00,0.1,0.0] BM[0000.00.00.00.00.000.000...0.0,0.0,0.0...0.0000...0.0000,0...11.000...110.110.001]

```
4473
4474
4475
4476
4477
4478
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4499
4500
4501
4502
4503
4504
4505
4506
4507
4508
4509
4510
4511
4512
4513
4514
4515
4516
4517
4518
4519
4520
4521
```

*** TEST 036 ***

TEST-036 USES A DATA PATTERN OF: "1 101 000 101 000 110" (150506)

1* PART A *

TEST-036-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY

1AS CLASS=A=DOP*SMO*DMO; IR<15:12>H="1101";

1SM=IR<11:9>H="000", SMOH=1; DM=IR<5:3>H="000", DMOH=1; AND TARGETS TO (455)

5661:

TEST036A:

PO, LOAD=ENUA(ZTARGET455), ;LOAD EXPECTED ADDRESS AFTER "BUT"

LOAD=ERROR(TEST036A), ;ERROR DIRECTORY KEY

DCS=CTR(C4.), ;COMPARE ENUA:THUA IN 4. MICROWORDS

NEXT, J/LOAD036A ;GO LOAD PATTERN

(5661) DCS[1.00,1.0,0.0] BM[1011.00.11.11.00.101.101...0.0,0.0,0.0...0.0000...0.0000,0...11.000...110.101.110]

5656:

LOAD036A:

PO, BUMP-VERIFY, ;ICOUNT

P2=U, IR_EMIT, ;LOAD IR WITH TEST PATTERN

EMIT/150506, ;(150506)

NEXT, J/GOBUTO36A ;GO SETUP FOR "BUT"

(5656) DCS[0.00,0.0,0.1] BM[1101.00.00.01.01.000.110...0.0,0.0,0.0...1.1010...0.0000,0...11.000...001.000.111]

5107: 1(FREF)

GOBUTO36A:

SETUP, RETURN/SCOPE036, ;RETURN TO SCOPE LOOP TEST WORD

NEXT, GOTO-PAGE(7), ;BUT TABLE IS ON PAGE 7

J/ROUTINSTR1 ;GO DO INSTR1 "BUT"

(5107) DCS[0.00,0.0,0.0] BM[0101.00.00.10.01.000.111...0.0,0.0,0.0...0.0000...0.0000,0...11.100...011.000.110]

5110: 1(FREE)

SCOPE036:

NEXT, BUTD[SCOPE], ;NO ERROR: "TEST037A" (+1, WORD)

J/TEST037A ;ERROR: "LOAD036A" (-2, WORDS)

(5110) DCS[0.00,0.1,0.0] BM[0000.00.00.00.00.000.000...0.0,0.0,0.0...0.0000...0.0000,0...11.000...110.101.111]

*** TEST 037 ***

```
4522 ITEST-037 USES A DATA PATTERN OF: "0 110 000 010 000 101" (060205)
4523
4524 | - - - - -
4525
4526 |* PART A *
4527 ITEST-037-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4528 |AS CLASS=ADOP+SMO+DMO; IR<15:12>H="0110";
4529 |SM=IR<11:9>H="000", SMOH=1; DM=IR<5:3>H="000", DMO=1; AND TARGETS TO (446)
4530 5657:
4531 TEST037A:
4532 PO, LOAD-ENUA(ZTARGET446), ILOAD EXPECTED ADDRESS AFTER "BUT"
4533 LOAD-ERROR(TEST037A), IERROR DIRECTORY KEY
4534 DCS-CTR(C4.), ICOMPARE ENUA:ENUA IN 4, MICROWORDS
4535 NEXT, J/LOAD037A IGO LOAD PATTERN
(5657) DCS[1.00.1.0.0.0] BM[1011.00.11.11.00.100.110.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.110.101.100]
4536
4537 5654:
4538 LOAD037A:
4539 P2-U, IR_EMIT, ILOAD IR WITH TEST PATTERN
4540 EMIT/060205, I(060205)
4541 NEXT, J/GOBUT037A IGO SETUP FOR "BUT"
4542 (5654) DCS[0.00.0.0.0.0] BM[0110.00.00.00.10.000.101.0.0.0.0.0.0.1.1010.0.0.0.0.0.0.11.000.0.001.001.001]
4543
4544 5111: I(FREE)
4545 GOBUT037A:
4546 SETUP, RETURN/SCOPE037, IRETURN TO SCOPE LOOP TEST WORD
4547 PO, BUMP-VERIFY, ICOUNT
4548 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4549 J/BUTINSTR1 IGO DO INSTR1 "BUT"
4550 (5111) DCS[0.00.0.0.0.1] BM[0101.00.00.10.01.010.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011.000.110]
4551
4552 5112: I(FREE)
4553 SCOPE037:
4554 NEXT, BUTD[SCOPE], INO ERROR; "TEST040A" [+1, WORD]
4555 J/TEST040A I ERROR; "LOAD037A" [-2, WORDS]
4556 (5112) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.110.101.101]
4557
4558
4559
4560
4561
4562
4563 | - - - - -
4564
4565 |== TEST 040 ==
4566 ITEST-040 USES A DATA PATTERN OF: "0 101 000 111 111 111" (080777)
4567
4568 | - - - - -
4569
4570 |* PART A *
```

```
4571 ITEST-040-A CHECKS THAT BUT (IR<15:12>) READS THE
4572 |ALTERNATING PATTERN "0101" IN IR<15:12>H CORRECTLY
4573 5655:
4574 TEST040A:
4575 PO, LOAD-ENUA(ZTARGET405), ILOAD EXPECTED ADDRESS AFTER "BUT"
4576 LOAD-ERROR(TEST040A), IERROR DIRECTORY KEY
4577 DCS-CTR(C4.), ICOMPARE ENUA:ENUA IN 4, MICROWORDS
4578 NEXT, J/LOAD040A IGO LOAD PATTERN
(5655) DCS[1.00.1.0.0.0] BM[1011.00.11.11.00.000.101.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.110.101.000]
4579
4580 5650:
4581 LOAD040A:
4582 P2-U, IR_EMIT, ILOAD IR WITH TEST PATTERN
4583 EMIT/080777, I(080777)
4584 NEXT, J/GOBUT040A IGO SETUP FOR "BUT"
4585 (5650) DCS[0.00.0.0.0.0] BM[0101.00.00.01.11.111.111.0.0.0.0.0.0.1.1010.0.0.0.0.0.0.11.000.0.001.001.011]
4586
4587 5113: I(FREE)
4588 GOBUT040A:
4589 SETUP, RETURN/TEST040B, IRETURN TO START OF NEXT SUBTEST
4590 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4591 J/BUTIR15-12 IGO DO "BUT"
4592 (5113) DCS[0.00.0.0.0.0] BM[0101.00.11.01.01.011.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011.000.000]
4593
4594
4595 | - - - - -
4596
4597 |* PART B *
4598 ITEST-040-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4599 |AS CLASS=B+DOP+MOV+SMO+DMO; DM=IR<5:3>H="111";
4600 |DOP=IR<14:12>H="101"; TARGETS TO (517)
4601 5653:
4602 TEST040B:
4603 PO, LOAD-ENUA(ZTARGET517), ILOAD EXPECTED ADDRESS AFTER "BUT"
4604 LOAD-ERROR(TEST040B), IERROR DIRECTORY KEY
4605 DCS-CTR(C3.), ICOMPARE ENUA:ENUA IN 3, MICROWORDS
4606 RUMP-VERIFY, ICOUNT
4607 NEXT, J/GOBUT040B IGO SETUP FOR "BUT"
(5653) DCS[1.00.1.0.0.1] BM[1100.00.11.11.01.001.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.001.001.100]
4608
4609 5114: I(FREE)
4610 GOBUT040B:
4611 SETUP, RETURN/SCOPE040, IRETURN TO SCOPE LOOP TEST WORD
4612
4613 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4614 J/BUTINSTR1 IGO DO INSTR1 "BUT"
(5114) DCS[0.00.0.0.0.0] BM[0101.00.00.10.01.101.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011.000.110]
4615
4616 5115: I(FREE)
```

```
4618 SCOPE040:
4619 NEXT, BUTD(SCOPE),          !NO ERROR: "TEST041A" [+1, WORDS]
4620 J/TEST041A                   ! ERROR: "LOAD040A" [-4, WORDS]
(5115) DCS[0.00,0.1,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0..0.0000..0.0000,0...11,000...110,101,001]

4621
4622
4623
4624
4625
4626
4627 | - - - - -
4628
4629 |*** TEST 041 ***
4630 |TEST-041 USES A DATA PATTERN OF: "1 010 000 111 111 111" (120777)
4631
4632 | - - - - -
4633
4634 |* PART A *
4635 |TEST-041-A CHECKS THAT BUT[IR<15:12>] READS THE
4636 |ALTERNATING PATTERN "1010" IN IR<15:12>H CORRECTLY
4637 5651:
4638 TEST041A:
4639 PO,          LOAD-ENUA(ZTARGET412),          !LOAD EXPECTED ADDRESS AFTER "BUT"
4640              LOAD-ERROR(TEST041A),          !ERROR DIRECTORY KEY
4641              DCS-CTR(C4.),                  !COMPARE ENUA:TWUA IN 4, MICROWORDS
4642 NEXT, J/LOAD041A          !GO LOAD PATTERN
(5651) DCS[1.00,1.0,0.0] BM[1011..00.11..11,00..001..010...0.0,0.0..0.0..0.0000..0.0000,0...11,000...110,100,100]

4643
4644 5644:
4645 LOAD041A:
4646 PO,          BUMP-VERIFY,          !COUNT
4647 P2-U,        IR-EMIT,              !LOAD IR WITH TEST PATTERN
4648              EMIT/120777,          !120777
4649 NEXT, J/GOBUT041A          !GO SETUP FOR "BUT"
(5644) DCS[0.00,0.0,0.0,1] BM[1010..00.00..01,11..111..111...0.0,0.0..0.0..0.1010...0.0000,0...11,000...001,001,110]

4651
4652 5116: 1(FREE)
4653 GOBUT041A:
4654 SETUP, RETURN/TEST041B,          !RETURN TO START OF NEXT SURTEST
4655 NFXT, GOTO-PAGE(7),              !BUT TABLE IS ON PAGE 7
4656 J/RUTIR15-12          !GO DO "BUT"
(5116) DCS[0.00,0.0,0.0,0] BM[0101..00.11..01,00..111..111...0.0,0.0..0.0..0.0000..0.0000,0...11,100...011,000,000]

4658
4659 | - - - - -
4660
4661 |* PART B *
4662 |TEST041-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4663 |AS CLASS-B=DOP+MOV+SMO+DMO; DM=IR<5:3>H="111";
4664 |DOP=IR<14:12>H="010"; TARGETS TO (517)
4665 5647:
4666
```

```
4667 TEST041B:
4668 PO,          LOAD-ENUA(ZTARGETS17),          !LOAD EXPECTED ADDRESS AFTER "BUT"
4669              LOAD-ERROR(TEST041B),          !ERROR DIRECTORY KEY
4670              DCS-CTR(C3.),                  !COMPARE ENUA:TWUA IN 3, MICROWORDS
4671              BUMP-VERIFY,          !COUNT
4672 NFXT, J/GOBUT041B          !GO SETUP FOR "BUT"
(5647) DCS[1.00,1.0,0.0,1] BM[1100..00.11..11,01..001..111...0.0,0.0..0.0..0.0000..0.0000,0...11,000...001,001,111]

4673
4674 5117: 1(FREE)
4675 GOBUT041B:
4676 SETUP, RETURN/SCOPE041,          !RETURN TO SCOPE LOOP TEST WORD
4677 NFXT, GOTO-PAGE(7),              !BUT TABLE IS ON PAGE 7
4678 J/RUTINSTR1          !GO DO INSTR1 "BUT"
(5117) DCS[0.00,0.0,0.0,0] BM[0101..00.00..10,10..000..111...0.0,0.0..0.0..0.0000..0.0000,0...11,100...011,000,110]

4680
4681 5120: 1(FREE)
4682 SCOPE041:
4683 NEXT, BUTD(SCOPE),          !NO ERROR: "TEST042A" [+1, WORDS]
4684 J/TEST042A                   ! ERROR: "LOAD041A" [-4, WORDS]
(5120) DCS[0.00,0.1,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0..0.0000..0.0000,0...11,000...110,100,101]

4686
4687
4688
4689
4690
4691
4692 | - - - - -
4693
4694 |*** TEST 042 ***
4695 |TEST-042 USES A DATA PATTERN OF: "1 100 100 000 010 000" (144020)
4696
4697 | - - - - -
4698
4699 |* PART A *
4700 |TEST-042-A CHECKS THAT BUT[DMO+SMO+BYTE]
4701 |DM=IR<5:3>H="010", DMOH=0; SM=IR<11:9>H="100", SMOH=0; BYTE H=1
4702 5645:
4703 TEST042A:
4704 PO,          LOAD-ENUA(ZTARGET401),          !LOAD EXPECTED ADDRESS AFTER "BUT"
4705              LOAD-ERROR(TEST042A),          !ERROR DIRECTORY KEY
4706              DCS-CTR(C4.),                  !COMPARE ENUA:TWUA IN 4, MICROWORDS
4707 NEXT, J/LOAD042A          !GO LOAD PATTERN
(5645) DCS[1.00,1.0,0.0,0] BM[1011..00.11..11,00..000..001...0.0,0.0..0.0..0.0000..0.0000,0...11,000...110,100,000]

4708
4709 5640:
4710 LOAD042A:
4711 P2-U,        IP-FMIT,              !LOAD IR WITH TEST PATTERN
4712              FMIT/144020,          !144020
4713 NEXT, J/GOBUT042A          !GO SETUP FOR "BUT"
(5640) DCS[0.00,0.0,0.0,0] BM[1100..00.10..00,00..010..000...0.0,0.0..0.0..0.1010...0.0000,0...11,000...001,010,001]

4714
```

```

4715      51211: 1(FREE)
4716      GDBUT042A1
4717          SETUP, RETURN/TEST042B, ;RETURN TO START OF NEXT SUBTEST
4718          NEXT, GOTO=PAGE(7), ;BUT TABLE IS ON PAGE 7
4719          J/BUTDMSMBYTE ;GO DO "BUT"
4720      (5121) DCS[0.00,0.0,0.0] BM[0101,,00,11,,01,00,,011,,111,,0,0,0,,0,0...0.0000...0'.0000,0...11,100,...011,001,011]
4721
4722      | - - - - -
4723
4724      1* PART B *
4725      1TEST-042-M CHECKS THAT BUT(INSTR1) READS THE IR CORRECTLY
4726      1AM CLASS-G=DOP+SMO; DOP=IR<14:12>H="100";
4727      1SM=IR<11:9>H="100"; TARGETS TO (714)
4728      5643:
4729      TEST042B:
4730          PO, LOAD=ENUA(ZTARGET714), ;LOAD EXPECTED ADDRESS AFTER "BUT"
4731          DCS=ERROR(TEST042B), ;ERROR DIRECTORY KEY
4732          DCS=CTRC(), ;COMPARE ENUA:TNUA IN 3, MICROWORDS
4733          NEXT, J/GDBUT042B ;GO SETUP FOR "BUT"
4734      (5643) DCS[1.00,1.0,0.0] BM[1100,,00,11,,11,11,,001,,100,,0,0,0,,0,0...0.0000...0'.0000,0...11,000,...001,010,010]
4735
4736      5122: 1(FREE)
4737      GDBUT042B:
4738          SETUP, RETURN/SCOPE042, ;RETURN TO SCOPE LOOP TEST WORD
4739          PO, RUMP=VERIFY, ;COUNT
4740          NEXT, GOTO=PAGE(7), ;BUT TABLE IS ON PAGE 7
4741          J/BUTINSTR1 ;GO DO INSTR1 "BUT"
4742      (5122) DCS[0.00,0.0,0,1] BM[0101,,00,00,,10,10,,011,,111,,0,0,0,,0,0...0.0000...0'.0000,0...11,100,...011,000,110]
4743
4744      5123: 1(FREE)
4745      SCOPF042:
4746          NEXT, BUTD[SCOPE], ;NO ERROR: "TEST043A" [+1, WORDS]
4747          J/TEST043A ; ERROR: "LOAD042A" [-4, WORDS]
4748      (5123) DCS[0.0,0,1,0,0] BM[0000,,00,00,,00,00,,000,,000,,0,0,0,,0,0...0.0000...0'.0000,0...11,000,...110,100,001]
4749
4750
4751
4752
4753
4754
4755      | - - - - -
4756
4757      1*** TEST 043 ***
4758      1TEST-043 USES A DATA PATTERN OF: "1 110 001 111 000 111" (161707)
4759
4760
4761      | - - - - -
4762

```

```

4763 1* PART A *
4764 !TEST-043-A CHECKS THAT BUT[DM08:SM08:BYTE]
4765 !DM=IR<5;3>H=000", DM0H=1; SM=IR<11;9>H=001", SM0H=0; BYTE H=0
4766 56411
4767 TEST043A:
4768     PO,          LOAD=ENUA(ZTARGET040),          !LOAD EXPECTED ADDRESS AFTER "BUT"
4769                LOAD=ERROR(TEST043A),            !ERROR DIRECTORY KEY
4770                DCS=CTR(C4,),                    !COMPARE ENUA:TWUA IN 4, MICROWORDS
4771                NEXT, J/LOAD043A                  !GO LOAD PATTERN
(5641) DCS(1,00,1,0,0,0) RM(1011,00,11,11,00,000,100,0,0,0,0,0,0,0,0000,0,0,000,0,11,000,110,011,101)
4772
4773 5634:
4774 LOAD043A:
4775     PO,          RUMP=VERIFY,                    !COUNT
4776     P2-U,        IR=FWIT,                        !LOAD IR WITH TEST PATTERN
4777                EMIT/161707,                      !161707)
4778     NEXT,        J/GOBUTO43A                     !GO SETUP FOR "BUT"
4779 (5634) DCS(0,00,0,0,0,1) RM(1110,00,00,11,11,000,111,0,0,0,0,0,0,0,1010,0,0,0000,0,11,000,001,010,100)
4780
4781 5124: 1(FREF)
4782 GOBUTO43A:
4783     SETUP,       RETURN/TEST043B,                !RETURN TO START OF NEXT SUBTEST
4784     NEXT,        GOTO=PAGE(7),                   !BUT TABLE IS ON PAGE 7
4785     J/BUTDM8NBYTE
4786 (5124) DCS(0,00,0,0,0,0) RM(0101,00,11,00,11,111,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,001,011)
4787
4788 1 - - - - -
4789
4790 1* PART B *
4791 !TEST-043-B CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4792 !AS CLASS=G=DOP+SM0; DOP=IR<14;12>H=110";
4793 !SM=IR<11;9>H=001"; TARGETS TO (711)
4794 5637:
4795 TEST043B:
4796     PO,          LOAD=ENUA(ZTARGET711),          !LOAD EXPECTED ADDRESS AFTER "BUT"
4797                LOAD=ERROR(TEST043B),            !ERROR DIRECTORY KEY
4798                DCS=CTR(C3,),                    !COMPARE ENUA:TWUA IN 3, MICROWORDS
4799     NEXT,        J/GOBUTO43B                     !GO SETUP FOR "BUT"
(5637) DCS(1,00,1,0,0,0) RM(1100,00,11,11,11,001,001,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,001,010,101)
4800
4801 5125: 1(FREF)
4802 GOBUTO43B:
4803     SETUP,       RETURN/SCOPE043,                !RETURN TO SCOPE LOOP TEST WORD
4804     NEXT,        GOTO=PAGE(7),                   !BUT TABLE IS ON PAGE 7
4805     J/BUTINSTR1
4806 (5125) DCS(0,00,0,0,0,0) RM(0101,00,00,10,10,110,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,110)
4807
4808
4809

```

```

4810 5126: 1(FREE)
4811 SCOPE043;
4812 PO, BUMP-VERIFY, I/COUNT
4813 NEXT, BUTD[SCOPE], I/NO ERROR: "TEST044A" [+1, WORDS]
4814 J/TEST044A I/ERROR: "LOAD043A" [-2, WORDS]
(5126) DCS[0.00,0.1,0.1] BM[0000..00.00..00.00..000..000..0.0,0.0,0.0..0.0000..0.0000,0...11,000...110,011,011]
4815
4816
4817
4818
4819
4820
4821
4822 1 - - - - -
4823
4824 *** TEST 044 ***
4825 I/TEST-044 USES A DATA PATTERN OF: "0 111 110 010 101 000" (076250)
4826
4827 1 - - - - -
4828
4829 [* PART A *
4830 I/TEST-044-A CHECKS THAT BUT[INSTR1] READS THE IR CORRECTLY
4831 I/AS CLASS-A=XFC; IR<15:9>H="0 111 110"; AND TARGETS TO (447)
4832 5635;
4833 TEST044A;
4834 PO, LOAD-ENUA(ZTARGET447), I/LOAD EXPECTED ADDRESS AFTER "BUT"
4835 LOAD-ERROR(TEST044A), I/ERROR DIRECTORY KEY
4836 DCS-CTR(C4.), I/COMPARE ENUA:TNVA IN 4. MICROWORDS
4837 NEXT, J/LOAD044A I/GO LOAD PATTERN
(5635) DCS[1.00,1.0,0.0] BM[1011..00.11..11.00..100..111..0.0,0.0,0.0..0.0000..0.0000,0...11,000...110,011,010]
4838
4839
4840 5632;
4841 LOAD044A;
4842 P2-U, IR_EMIT, I/LOAD IR WITH TEST PATTERN
4843 EMIT/076250, I/(076250)
4844 NEXT, J/G08UT044A I/GO SETUP FOR "BUT"
(5632) DCS[0.00,0.0,0.0] BM[0111..00.11..00.10..101..000..0.0,0.0,0.0..1.1010..0.0000,0...11,000...001,010,111]
4845
4846
4847 5127: 1(FREE)
4848 G08UT044A;
4849 SETUP, RETURN/SCOPE044, I/RETURN TO SCOPE LOOP TEST WORD
4850 NEXT, GOTO-PAGE(7), I/BUT TABLE IS ON PAGE 7
4851 J/BUTINSTR1 I/GO DO INSTR; "BUT"
(5127) DCS[0.00,0.0,0.0] BM[0101..00.00..10.11..000..111..0.0,0.0,0.0..0.0000..0.0000,0...11,100...011,000,110]
4852
4853
4854 5130: 1(FREE)
4855 SCOPE044;
4856 PO, BUMP-VERIFY, I/COUNT
4857 NEXT, BUTD[SCOPE], I/NO ERROR: "TEST045A" [+1, WORD]
4858 J/TEST045A I/ERROR: "LOAD044A" [-2, WORDS]

```

```

(5130) DCS[0.00,0.1,0.1] BM[0000..00.00..00.00..000..000..0.0,0.0,0.0..0.0000..0.0000,0...11,000...110,011,011]
4859
4860
4861
4862
4863
4864
4865
4866 1 - - - - -
4867
4868 *** TEST 045 ***
4869 I/TEST-045 USES A DATA PATTERN OF: "0 000 100 011 000 000" (004300)
4870
4871 1 - - - - -
4872
4873 [* PART A *
4874 I/TEST-045-A CHECKS THAT BUT[DM#SM#BYTE] READS THE IR CORRECTLY AS:
4875 I/DM H="000", DM#H=1; SM H="100", SM#H=0; BYTE H=1 (IR#SWAB, SORT OF)
4876 I/CHECKS THAT SWAB INSTR ASSERTS BYTE H
4877 5633;
4878 TEST045A;
4879 PO, LOAD-ENUA(ZTARGET405), I/LOAD EXPECTED ADDRESS AFTER "BUT"
4880 LOAD-ERROR(TEST045A), I/ERROR DIRECTORY KEY
4881 DCS-CTR(C4.), I/COMPARE ENUA:TNVA IN 4. MICROWORDS
4882 NEXT, J/LOAD045A I/GO LOAD PATTERN
(5633) DCS[1.00,1.0,0.0] BM[1011..00.11..11.00..000..101..0.0,0.0,0.0..0.0000..0.0000,0...11,000...110,011,000]
4883
4884
4885 5630;
4886 LOAD045A;
4887 P2-U, IR_EMIT, I/LOAD IR WITH TEST PATTERN
4888 EMIT/004300, I/(004300)
4889 NEXT, J/G08UT045A I/GO SETUP FOR "BUT"
(5630) DCS[0.00,0.0,0.0] BM[0000..00.10..00.11..000..000..0.0,0.0,0.0..1.1010..0.0000,0...11,000...001,011,001]
4890
4891
4892 5131: 1(FREE)
4893 G08UT045A;
4894 SETUP, RETURN/SCOPE045, I/RETURN TO SCOPE LOOP TEST WORD
4895 NEXT, GOTO-PAGE(7), I/BUT TABLE IS ON PAGE 7
4896 J/BUTDM#SM#BYTE I/GO DO BUT ON DM#SM#BYTE
(5131) DCS[0.00,0.0,0.0] BM[0101..00.00..10.11..010..111..0.0,0.0,0.0..0.0000..0.0000,0...11,100...011,001,011]
4897
4898
4899 5132: 1(FREE)
4900 SCOPE045;
4901 NEXT, BUTD[SCOPE], I/NO ERROR: "TEST046A" [+1, WORD]
4902 J/TEST046A I/ERROR: "LOAD045A" [-2, WORDS]
(5132) DCS[0.00,0.1,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0,0.0..0.0000..0.0000,0...11,000...110,011,001]
4903
4904
4905
4906

```

```
4907
4908 1 - - - - -
4909
4910 *** TEST 046 ***
4911 1TEST-046 USES A DATA PATTERN OF: "0 000 000 010 000 000" (900200)
4912
4913 1 - - - - -
4914
4915 1* PART A *
4916 1TEST-046-A CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
4917 1AS ROM ADDRESS=(426) ON THE INSTR5 E00 ROW, AND RECEIVES THE VALUE
4918 1OF (06), TARGETING TO (426) AFTER THE DECODE
4919 5631:
4920 TEST046A:
4921 PO, LOAD-ENUA(ZTARGET426), ILOAD EXPECTED ADDRESS AFTER "BUT"
4922 LOAD-ERROR(TEST046A), IERROR DIRECTORY KEY
4923 DCS-CTR(C4.), ICOMPARE ENUA:ENUA IN 4, MICROWORDS
4924 J/LOAD046A IGO LOAD PATTERN
(5631) DCS[1.00.1.0.0.0] BM[1011..00.11..11.00..010..110...0.0.0..0.0...0.0000...0..0000.0...11.000...101.110.000]
4925
4926
4927 5560:
4928 LOAD046A:
4929 P2=0, IR_EMIT, ILOAD IR WITH TEST PATTERN
4930 EMIT/000200, I(000200)
4931 NEXT, J/GOBUT046A IGO SETUP FOR "BUT"
(5560) DCS[0.00.0.0.0.0] BM[0000..00.00..00.10..000..000...0.0.0..0.0...1.1010...0..0000.0...11.000...001.011.011]
4932
4933 5133: 1(FREE)
4934 GOBUT046A:
4935 SETUP, RETURN/TEST046B, IRETURN TO START OF NEXT SUBTEST
4936 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4937 J/BUTINSTRS IGO DO INSTR5 "BUT"
(5133) DCS[0.00.0.0.0.0] BM[0101..00.11..11.11..000..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.000.001]
4938
4939
4940
4941
4942
4943 1 - - - - -
4944
4945 1* PART B *
4946 1TEST-046-B CHECKS THAT BUT[IR<11>:FLTP<3:0>] READS THE "0" IN IR<11>N CORRECTLY,
4947 1AND THE FLTP DECODE ROM GETS ADDRESS (040), WHICH IS A 51ST INSTR,
4948 1DATA SHOULD BE (01)
4949 5770:
4950 TEST046B:
4951 PO, LOAD-ENUA(ZTARGET401), ILOAD EXPECTED ADDRESS AFTER "BUT"
4952 LOAD-ERROR(TEST046B), IERROR DIRECTORY KEY
4953 DCS-CTR(C3.), ICOMPARE ENUA:ENUA IN 3, MICROWORDS
4954 J/GOBUT046B I
(5770) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..000..001...0.0.0..0.0...0.0000...0..0000.0...11.000...001.011.100]
4955
```

```
4956
4957 5134: 1(FREE)
4958 GOBUT046B:
4959 SETUP, RETURN/SCOPE046, IRETURN TO SCOPE LOOP TEST WORD
4960 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
4961 J/BUTIR1:FLTP<3:0> IGO DO "BUT" ON IR<11>:FLTP<3:0>H
(5134) DCS[0.00.0.0.0.0] BM[0101..00.00..10.11..101..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.000.010]
4962
4963
4964
4965
4966 5135: 1(FREE)
4967 SCOPE046:
4968 NEXT, BUTD[SCOPE], INO ERROR: "TEST047A" [+1, WORD]
4969 J/TEST047A I ERROR: "LOAD046A" [-4, WORDS]
(5135) DCS[0.00.0.1.0.0] BM[0000..00.00..00.00..000..000...0.0.0..0.0...0.0000...0..0000.0...11.000...101.110.001]
4970
4971
4972
4973
4974
4975
4976 1 - - - - -
4977
4978 *** TEST 047 ***
4979 1TEST-047 USES A DATA PATTERN OF: "1 010 010 101 110 110" (122566)
4980
4981 1 - - - - -
4982
4983 1* PART A *
4984 1TEST-047-A CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
4985 1AS ROM ADDRESS=(325) ON THE INSTR5 E70 ROW, AND RECEIVES THE DIAGNOSTIC VALUE
4986 1OF (12), TARGETING TO (412) AFTER THE DECODE
4987 5561:
4988 TEST047A:
4989 PO, LOAD-ENUA(ZTARGET412), ILOAD EXPECTED ADDRESS AFTER "BUT"
4990 LOAD-ERROR(TEST047A), IERROR DIRECTORY KEY
4991 DCS-CTR(C4.), ICOMPARE ENUA:ENUA IN 4, MICROWORDS
4992 BUMP-VERIFY, ICOUNT
4993 NEXT, J/LOAD047A IGO LOAD PATTERN
(5561) DCS[1.00.1.0.0.1] BM[1011..00.11..11.00..001..010...0.0.0..0.0...0.0000...0..0000.0...11.000...101.101.000]
4994
4995
4996 5560:
4997 LOAD047A:
4998 P2=0, IR_EMIT, ILOAD IP WITH TEST PATTERN
4999 EMIT/122566, I(122566)
5000 NEXT, J/GOBUT047A IGO SETUP FOR "BUT"
(5560) DCS[0.00.0.0.0.0] BM[1010..00.01..01.01..110..110...0.0.0..0.0...1.1010...0..0000.0...11.000...001.011.110]
5001
5002
5003 5136: 1(FREE)
5004 GOBUT047A:
```

```
5005      SETUP, RETURN/TEST047B,      |RETURN TO START OF NEXT SUBTEST
5006      NEXT,   GOTO-PAGE(7),          |BUT TABLE IS ON PAGE 7
5007      J/BUTINSTRS                      |GO DO INSTRS "BUT"
(5136) DCS(0.00,0.0,0.0) BM(0101,00,11,01,11,000,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,001)
5008
5009
5010
5011
5012 | - - - - -
5013
5014 |* PART B *
5015 |TEST-047-B CHECKS THAT BUT[IR<11>:FLTPT<3:0>] READS THE "0" IN IR<11>H CORRECTLY,
5016 |AND THE FLTPT DECODE ROM GETS ADDRESS (534), WHICH IS A LOAD/NODE-6 INSTR,
5017 |DATA SHOULD BE (13)
5018 5670:
5019 TEST047B:
5020     PO,      LOAD-ENUA(ZTARGET413),      |LOAD EXPECTED ADDRESS AFTER "BUT"
5021             LOAD-ERROR(TEST047B),      |ERROR DIRECTORY KEY
5022             DCS-CTR(C3,),              |COMPARE ENUA:TNVA IN 3, MICROWORDS
5023     NEXT,    J/GOBUT047B
(5670) DCS(1.00,1.0,0.0) BM(1100,00,11,11,00,001,011,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,001,011,111)
5024
5025
5026 5137: 1(FREE)
5027 GOBUT047B:
5028     PO,      BUMP-VERIFY,              |COUNT
5029     SETUP,   RETURN/TEST047C,          |RETURN TO START OF NEXT SUBTEST
5030     NEXT,    GOTO-PAGE(7),              |BUT TABLE IS ON PAGE 7
5031     J/BUTIR11FLTPT3=0
(5137) DCS(0.00,0.0,0.0) BM(0101,00,11,10,11,000,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,010)
5032
5033 | - - - - -
5034
5035 |* PART C *
5036 |TEST-047-C CHECKS THAT BUT[MOV+DR7:IR<5:3>] READS THE -(FLTPT+MOV+FLTPT+DR7), IR<5:3>="110"
5037 5730:
5038 TEST047C:
5039     PO,      LOAD-ENUA(ZTARGET406),      |LOAD EXPECTED ADDRESS AFTER "BUT"
5040             LOAD-ERROR(TEST047C),      |ERROR DIRECTORY KEY
5041             DCS-CTR(C3,),              |COMPARE ENUA:TNVA IN 3, MICROWORDS
5042     NEXT,    J/GOBUT047C
(5730) DCS(1.00,1.0,0.0) BM(1100,00,11,11,00,000,110,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,001,100,000)
5044
5045
5046 5140: 1(FREE)
5047 GOBUT047C:
5048     SETUP,   RETURN/SCOPE047,          |RETURN TO SCOPE LOOP TEST WORD
5049     NEXT,    GOTO-PAGE(7),              |BUT TABLE IS ON PAGE 7
5050     J/BUTMOVDR7IRS=3
(5140) DCS(0.00,0.0,0.0) BM(0101,00,00,11,00,001,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,101)
5051
5052
5053
5054
5055
5056
5057
5058
5059
5060
5061
5062
5063
5064
5065
5066
5067 |*** TEST 050 ***
5068 |TEST-050-A USES A DATA PATTERN OF: "1 111 010 101 000 000" (172500)
5069 |TEST-050-B USES A DATA PATTERN OF: "1 111 101 010 000 000" (172000)
5070 | - - - - -
5071
5072 |* PART A *
5073 |TEST-050-A CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
5074 |AS ROM ADDRESS=(725) ON THE INSTR5 E78 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
5075 |OF (05), TARGETING TO (405) AFTER THE DECODE
5076 5551:
5077 TEST050A:
5078     PO,      LOAD-ENUA(ZTARGET405),      |LOAD EXPECTED ADDRESS AFTER "BUT"
5079             LOAD-ERROR(TEST050A),      |ERROR DIRECTORY KEY
5080             DCS-CTR(C4,),              |COMPARE ENUA:TNVA IN 4, MICROWORDS
5081             BUMP-VERIFY,              |COUNT
5082     NEXT,    J/LOAD050A
(5551) DCS(1.00,1.0,0.0) BM(1011,00,11,11,00,000,101,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,101,101,010)
5083
5084
5085 5552:
5086 LOAD050A:
5087     P2=U,    IR_EMIT,                  |LOAD IR WITH TEST PATTERN
5088             EMIT/172500,                | (172500)
5089     NEXT,    J/GOBUT050A
(5552) DCS(0.00,0.0,0.0) BM(1111,00,01,01,01,000,000,0,0,0,0,0,0,1,1010,0,0,0000,0,11,000,001,100,010)
5090
5091
5092 5142: 1(FREE)
5093 GOBUT050A:
5094     SETUP,   RETURN/TEST050B,          |RETURN TO START OF NEXT SUBTEST
5095     NEXT,    GOTO-PAGE(7),              |BUT TABLE IS ON PAGE 7
5096     J/BUTINSTRS
(5142) DCS(0.00,0.0,0.0) BM(0101,00,11,10,10,110,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,001)
5097
5098
5099
```

```
5051
5052
5053
5054
5055 5141: 1(FREE)
5056 SCOPE047:
5057     NEXT,    BUTD[SCOPE],              |NO ERROR: "TEST050A" [+1, WORD]
5058             J/TEST050A                  | ERROR: "LOAD047A" [+6, WORDS]
(5141) DCS(0.00,0.1,0.0) BM(0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,101,101,001)
5059
5060
5061
5062
5063
5064 | - - - - -
5065
5066 |*** TEST 050 ***
5067 |TEST-050-A USES A DATA PATTERN OF: "1 111 010 101 000 000" (172500)
5068 |TEST-050-B USES A DATA PATTERN OF: "1 111 101 010 000 000" (172000)
5069 | - - - - -
5070
5071 |* PART A *
5072 |TEST-050-A CHECKS THAT BUT[INSTR5] READS THE IR CORRECTLY
5073 |AS ROM ADDRESS=(725) ON THE INSTR5 E78 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
5074 |OF (05), TARGETING TO (405) AFTER THE DECODE
5075 5551:
5076 TEST050A:
5077     PO,      LOAD-ENUA(ZTARGET405),      |LOAD EXPECTED ADDRESS AFTER "BUT"
5078             LOAD-ERROR(TEST050A),      |ERROR DIRECTORY KEY
5079             DCS-CTR(C4,),              |COMPARE ENUA:TNVA IN 4, MICROWORDS
5080             BUMP-VERIFY,              |COUNT
5081     NEXT,    J/LOAD050A
(5551) DCS(1.00,1.0,0.0) BM(1011,00,11,11,00,000,101,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,101,101,010)
5083
5084
5085 5552:
5086 LOAD050A:
5087     P2=U,    IR_EMIT,                  |LOAD IR WITH TEST PATTERN
5088             EMIT/172500,                | (172500)
5089     NEXT,    J/GOBUT050A
(5552) DCS(0.00,0.0,0.0) BM(1111,00,01,01,01,000,000,0,0,0,0,0,0,1,1010,0,0,0000,0,11,000,001,100,010)
5090
5091
5092 5142: 1(FREE)
5093 GOBUT050A:
5094     SETUP,   RETURN/TEST050B,          |RETURN TO START OF NEXT SUBTEST
5095     NEXT,    GOTO-PAGE(7),              |BUT TABLE IS ON PAGE 7
5096     J/BUTINSTRS
(5142) DCS(0.00,0.0,0.0) BM(0101,00,11,10,10,110,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,011,000,001)
5097
5098
5099
```

```

5100
5101 | - - - - -
5102
5103 |* PART B *
5104 |TEST-050-B CHECKS THAT BUT(INSTR=5) READS THE IR CORRECTLY
5105 |AS ROM ADDR=(752) ON THE INSTRS E78 ROM, AND RECEIVES THE DIAGNOSTIC VALUE
5106 |OF (12), TARGETING TO (412) AFTER THE DECODE
5107 |7261
5108 |TEST050B:
5109 |    PO,          LOAD=ENUA(ZTARGET412),          |LOAD EXPECTED ADDRESS AFTER "BUT"
5110 |                LOAD=ERROR(TEST050B),          |ERROR DIRECTORY KEY
5111 |                DCS=CTR(24.),          |COMPARE ENUA:THUA IN 4. MICROWORDS
5112 |    NEXT,        J/LOAD050B
5113 | (5726) DCS(1,00,1,0,0,0) BM(1011,00,11,11,00,001,010,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,001,100,011)
5114
5115 |143: 1(FREE)
5116 |LOAD050B:
5117 |    P2=U,        IR=EMIT,          |LOAD IR WITH TEST PATTERN
5118 |                EMIT/175200,          | (175200)
5119 |    NEXT,        J/GOBUT050B          |GO SETUP FOR "BUT"
5120 | (5143) DCS(0,00,0,0,0,0) BM(1111,00,10,10,10,000,000,0,0,0,0,0,0,1,1010,0,0,0000,0,0,11,000,001,100,100)
5121
5122 |144: 1(FREE)
5123 |GOBUT050B:
5124 |    SETUP,       RETURN/SCOPE050,          |RETURN TO SCOPE LOOP TEST WORD
5125 |    NEXT,        COTO=PAGE(7),          |BUT TABLE IS ON PAGE 7
5126 |                J/BUTINSTRS          |DO INSTRS BUT
5127 | (5144) DCS(0,00,0,0,0,0) BM(0101,00,00,11,00,101,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,011,000,001)
5128
5129 |145: 1(FREE)
5130 |SCOPE050:
5131 |    PO,          BUSDIN_EMIT=[1],          |RESET PROC UCON
5132 |                EN=CLK=IR[15=00],          |
5133 |    NEXT,        BUTD[SCOPE],          |NO ERROR: "TEST101A" [+1, WORD]
5134 |                J/TEST101A          |ERROR: "LOAD050A" [-5, WORDS]
5135 | (5145) DCS(0,00,0,1,0,0) BM(0000,00,00,00,01,000,100,0,0,0,0,0,0,1,1001,0,0,0000,0,0,11,000,101,101,011)
5136
5137 |.PAGE=====
5138
5139 |.TOC * TEST101: D -> DBUF -> IR PATH
5140
5141
5142 |*****
5143 |*
5144 |* TESTS: 101 - 104                      UNWORDS: 057 + 067
5145 |*
5146 |* FUNCTIONS:
5147 |*
5148 |* THESE TESTS VERIFY THE "EMIT -> CSP -> SBUS -> D -> DBUF -> IR" DATAPATH,

```

```

5149 |*
5150 |* TEST 101 FIRST VERIFIES THE "D -> DBUF -> IR" DATA PATH, INSURING THAT THE
5151 |* DBUF LATCH CAN BE WRITTEN WITH ZEROS, AND ENABLED ONTO BUSDIN, TO BE PUT
5152 |* INTO THE IR AND VERIFIED (VIA INSTRS DECODE, AS A HALT INSTRUCTION).
5153 |*
5154 |* TESTS 102-104 THEN GO ON TO FURTHER TEST THE FULL DATAPATH FROM EMIT TO
5155 |* IR, VIA THE EXTENDED ROUTE. THESE TESTS THEN VERIFY THE CSP WRITE, ADDRESSING AND
5156 |* DATAPATHS LOGIC.
5157 |*
5158 |*****
5159
5160 |5531:
5161 |TEST101A:
5162 |    PO,          LOAD=ENUA(ZTARGET414),          |INSTRS E88 OUTPUT FOR IR=(000000)
5163 |                LOAD=ERROR(TEST101A),          |ERROR DIRECTORY KEY
5164 |                DCS=CTR(27.),          |COMPARE AT TARGET
5165 |    NEXT,        J/LOAD0101A
5166 | (5553) DCS(1,00,1,0,0,0) BM(1000,00,11,11,00,011,100,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,101,010,000)
5167
5168 |5520:
5169 |LOAD0101A:
5170 |    P2=T,        D=ZERO,          |PUT (000000) IN D
5171 |    NEXT,        J/GOTEST101A          |
5172 | (5520) DCS(0,00,0,0,0,0) BM(0011,00,00,00,00,000,000,0,0,1,0,0,0,0,0,0000,0,0,0000,0,0,11,000,001,100,111)
5173
5174 |147: 1(FREE)
5175 |GOTEST101A:
5176 |    SETUP,       RETURN/SCOPE101,          |GOT TO SUBR THAT:
5177 |    NEXT,        CALL(DINTOIR=5),          |D -> DBUF -> IR, THEN BUT(INSTR5)
5178 | (5147) DCS(0,00,0,0,0,0) BM(0101,00,00,11,01,000,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,010,111,011)
5179
5180 |150: 1(FREE)
5181 |SCOPE101:
5182 |    NEXT,        BUTD[SCOPE],          |NO ERROR: "TEST102A" [+1, WORDS]
5183 |                J/TEST102A          |ERROR: "LOADCSP101A" [-3, WORDS]
5184 | (5150) DCS(0,00,0,1,0,0) BM(0000,00,00,00,00,000,000,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,101,010,001)
5185
5186 | - - - - -
5187
5188 |.TOC * TEST102-104: TESTING CSP ADDRESS/READ/WRITE FUNCTIONS
5189
5190 |THE FOLLOWING SFT OF FOUR TESTS VERIFIES THAT THE CSP, AND THE CSP ADDRESS FIELD "CSPADDR"
5191 |HAS NO STUCK ZERO BITS, AND THAT THE EMIT -> CSP -> D -> DBUF -> IR DATAPATH
5192 |IS FULLY FUNCTIONAL.
5193 |
5194 |
5195 |
5196 |
5197 |
5198 |
5199 |
5200 |
5201 |
5202 |
5203 |
5204 |
5205 |
5206 |
5207 |
5208 |
5209 |
5210 |
5211 |
5212 |
5213 |
5214 |
5215 |
5216 |
5217 |
5218 |
5219 |
5220 |
5221 |
5222 |
5223 |
5224 |
5225 |
5226 |
5227 |
5228 |
5229 |
5230 |
5231 |
5232 |
5233 |
5234 |
5235 |
5236 |
5237 |
5238 |
5239 |
5240 |
5241 |
5242 |
5243 |
5244 |
5245 |
5246 |
5247 |
5248 |
5249 |
5250 |
5251 |
5252 |
5253 |
5254 |
5255 |
5256 |
5257 |
5258 |
5259 |
5260 |
5261 |
5262 |
5263 |
5264 |
5265 |
5266 |
5267 |
5268 |
5269 |
5270 |
5271 |
5272 |
5273 |
5274 |
5275 |
5276 |
5277 |
5278 |
5279 |
5280 |
5281 |
5282 |
5283 |
5284 |
5285 |
5286 |
5287 |
5288 |
5289 |
5290 |
5291 |
5292 |
5293 |
5294 |
5295 |
5296 |
5297 |
5298 |
5299 |
5300 |
5301 |
5302 |
5303 |
5304 |
5305 |
5306 |
5307 |
5308 |
5309 |
5310 |
5311 |
5312 |
5313 |
5314 |
5315 |
5316 |
5317 |
5318 |
5319 |
5320 |
5321 |
5322 |
5323 |
5324 |
5325 |
5326 |
5327 |
5328 |
5329 |
5330 |
5331 |
5332 |
5333 |
5334 |
5335 |
5336 |
5337 |
5338 |
5339 |
5340 |
5341 |
5342 |
5343 |
5344 |
5345 |
5346 |
5347 |
5348 |
5349 |
5350 |
5351 |
5352 |
5353 |
5354 |
5355 |
5356 |
5357 |
5358 |
5359 |
5360 |
5361 |
5362 |
5363 |
5364 |
5365 |
5366 |
5367 |
5368 |
5369 |
5370 |
5371 |
5372 |
5373 |
5374 |
5375 |
5376 |
5377 |
5378 |
5379 |
5380 |
5381 |
5382 |
5383 |
5384 |
5385 |
5386 |
5387 |
5388 |
5389 |
5390 |
5391 |
5392 |
5393 |
5394 |
5395 |
5396 |
5397 |
5398 |
5399 |
5400 |
5401 |
5402 |
5403 |
5404 |
5405 |
5406 |
5407 |
5408 |
5409 |
5410 |
5411 |
5412 |
5413 |
5414 |
5415 |
5416 |
5417 |
5418 |
5419 |
5420 |
5421 |
5422 |
5423 |
5424 |
5425 |
5426 |
5427 |
5428 |
5429 |
5430 |
5431 |
5432 |
5433 |
5434 |
5435 |
5436 |
5437 |
5438 |
5439 |
5440 |
5441 |
5442 |
5443 |
5444 |
5445 |
5446 |
5447 |
5448 |
5449 |
5450 |
5451 |
5452 |
5453 |
5454 |
5455 |
5456 |
5457 |
5458 |
5459 |
5460 |
5461 |
5462 |
5463 |
5464 |
5465 |
5466 |
5467 |
5468 |
5469 |
5470 |
5471 |
5472 |
5473 |
5474 |
5475 |
5476 |
5477 |
5478 |
5479 |
5480 |
5481 |
5482 |
5483 |
5484 |
5485 |
5486 |
5487 |
5488 |
5489 |
5490 |
5491 |
5492 |
5493 |
5494 |
5495 |
5496 |
5497 |
5498 |
5499 |
5500 |
5501 |
5502 |
5503 |
5504 |
5505 |
5506 |
5507 |
5508 |
5509 |
5510 |
5511 |
5512 |
5513 |
5514 |
5515 |
5516 |
5517 |
5518 |
5519 |
5520 |
5521 |
5522 |
5523 |
5524 |
5525 |
5526 |
5527 |
5528 |
5529 |
5530 |
5531 |
5532 |
5533 |
5534 |
5535 |
5536 |
5537 |
5538 |
5539 |
5540 |
5541 |
5542 |
5543 |
5544 |
5545 |
5546 |
5547 |
5548 |
5549 |
5550 |
5551 |
5552 |
5553 |
5554 |
5555 |
5556 |
5557 |
5558 |
5559 |
5560 |
5561 |
5562 |
5563 |
5564 |
5565 |
5566 |
5567 |
5568 |
5569 |
5570 |
5571 |
5572 |
5573 |
5574 |
5575 |
5576 |
5577 |
5578 |
5579 |
5580 |
5581 |
5582 |
5583 |
5584 |
5585 |
5586 |
5587 |
5588 |
5589 |
5590 |
5591 |
5592 |
5593 |
5594 |
5595 |
5596 |
5597 |
5598 |
5599 |
5600 |
5601 |
5602 |
5603 |
5604 |
5605 |
5606 |
5607 |
5608 |
5609 |
5610 |
5611 |
5612 |
5613 |
5614 |
5615 |
5616 |
5617 |
5618 |
5619 |
5620 |
5621 |
5622 |
5623 |
5624 |
5625 |
5626 |
5627 |
5628 |
5629 |
5630 |
5631 |
5632 |
5633 |
5634 |
5635 |
5636 |
5637 |
5638 |
5639 |
5640 |
5641 |
5642 |
5643 |
5644 |
5645 |
5646 |
5647 |
5648 |
5649 |
5650 |
5651 |
5652 |
5653 |
5654 |
5655 |
5656 |
5657 |
5658 |
5659 |
5660 |
5661 |
5662 |
5663 |
5664 |
5665 |
5666 |
5667 |
5668 |
5669 |
5670 |
5671 |
5672 |
5673 |
5674 |
5675 |
5676 |
5677 |
5678 |
5679 |
5680 |
5681 |
5682 |
5683 |
5684 |
5685 |
5686 |
5687 |
5688 |
5689 |
5690 |
5691 |
5692 |
5693 |
5694 |
5695 |
5696 |
5697 |
5698 |
5699 |
5700 |
5701 |
5702 |
5703 |
5704 |
5705 |
5706 |
5707 |
5708 |
5709 |
5710 |
5711 |
5712 |
5713 |
5714 |
5715 |
5716 |
5717 |
5718 |
5719 |
5720 |
5721 |
5722 |
5723 |
5724 |
5725 |
5726 |
5727 |
5728 |
5729 |
5730 |
5731 |
5732 |
5733 |
5734 |
5735 |
5736 |
5737 |
5738 |
5739 |
5740 |
5741 |
5742 |
5743 |
5744 |
5745 |
5746 |
5747 |
5748 |
5749 |
5750 |
5751 |
5752 |
5753 |
5754 |
5755 |
5756 |
5757 |
5758 |
5759 |
5760 |
5761 |
5762 |
5763 |
5764 |
5765 |
5766 |
5767 |
5768 |
5769 |
5770 |
5771 |
5772 |
5773 |
5774 |
5775 |
5776 |
5777 |
5778 |
5779 |
5780 |
5781 |
5782 |
5783 |
5784 |
5785 |
5786 |
5787 |
5788 |
5789 |
5790 |
5791 |
5792 |
5793 |
5794 |
5795 |
5796 |
5797 |
5798 |
5799 |
5800 |
5801 |
5802 |
5803 |
5804 |
5805 |
5806 |
5807 |
5808 |
5809 |
5810 |
5811 |
5812 |
5813 |
5814 |
5815 |
5816 |
5817 |
5818 |
5819 |
5820 |
5821 |
5822 |
5823 |
5824 |
5825 |
5826 |
5827 |
5828 |
5829 |
5830 |
5831 |
5832 |
5833 |
5834 |
5835 |
5836 |
5837 |
5838 |
5839 |
5840 |
5841 |
5842 |
5843 |
5844 |
5845 |
5846 |
5847 |
5848 |
5849 |
5850 |
5851 |
5852 |
5853 |
5854 |
5855 |
5856 |
5857 |
5858 |
5859 |
5860 |
5861 |
5862 |
5863 |
5864 |
5865 |
5866 |
5867 |
5868 |
5869 |
5870 |
5871 |
5872 |
5873 |
5874 |
5875 |
5876 |
5877 |
5878 |
5879 |
5880 |
5881 |
5882 |
5883 |
5884 |
5885 |
5886 |
5887 |
5888 |
5889 |
5890 |
5891 |
5892 |
5893 |
5894 |
5895 |
5896 |
5897 |
5898 |
5899 |
5900 |
5901 |
5902 |
5903 |
5904 |
5905 |
5906 |
5907 |
5908 |
5909 |
5910 |
5911 |
5912 |
5913 |
5914 |
5915 |
5916 |
5917 |
5918 |
5919 |
5920 |
5921 |
5922 |
5923 |
5924 |
5925 |
5926 |
5927 |
5928 |
5929 |
5930 |
5931 |
5932 |
5933 |
5934 |
5935 |
5936 |
5937 |
5938 |
5939 |
5940 |
5941 |
5942 |
5943 |
5944 |
5945 |
5946 |
5947 |
5948 |
5949 |
5950 |
5951 |
5952 |
5953 |
5954 |
5955 |
5956 |
5957 |
5958 |
5959 |
5960 |
5961 |
5962 |
5963 |
5964 |
5965 |
5966 |
5967 |
5968 |
5969 |
5970 |
5971 |
5972 |
5973 |
5974 |
5975 |
5976 |
5977 |
5978 |
5979 |
5980 |
5981 |
5982 |
5983 |
5984 |
5985 |
5986 |
5987 |
5988 |
5989 |
5990 |
5991 |
5992 |
5993 |
5994 |
5995 |
5996 |
5997 |
5998 |
5999 |
6000 |
6001 |
6002 |
6003 |
6004 |
6005 |
6006 |
6007 |
6008 |
6009 |
6010 |
6011 |
6012 |
6013 |
6014 |
6015 |
6016 |
6017 |
6018 |
6019 |
6020 |
6021 |
6022 |
6023 |
6024 |
6025 |
6026 |
6027 |
6028 |
6029 |
6030 |
6031 |
6032 |
6033 |
6034 |
6035 |
6036 |
6037 |
6038 |
6039 |
6040 |
6041 |
6042 |
6043 |
6044 |
6045 |
6046 |
6047 |
6048 |
6049 |
6050 |
6051 |
6052 |
6053 |
6054 |
6055 |
6056 |
6057 |
6058 |
6059 |
6060 |
6061 |
6062 |
6063 |
6064 |
6065 |
6066 |
6067 |
6068 |
6069 |
6070 |
6071 |
6072 |
6073 |
6074 |
6075 |
6076 |
6077 |
6078 |
6079 |
6080 |
6081 |
6082 |
6083 |
6084 |
6085 |
6086 |
6087 |
6088 |
6089 |
6090 |
6091 |
6092 |
6093 |
6094 |
6095 |
6096 |
6097 |
6098 |
6099 |
6100 |
6101 |
6102 |
6103 |
6104 |
6105 |
6106 |
6107 |
6108 |
6109 |
6110 |
6111 |
6112 |
6113 |
6114 |
6115 |
6116 |
6117 |
6118 |
6119 |
6120 |
6121 |
6122 |
6123 |
6124 |
6125 |
6126 |
6127 |
6128 |
6129 |
6130 |
6131 |
6132 |
6133 |
6134 |
6135 |
6136 |
6137 |
6138 |
6139 |
6140 |
6141 |
6142 |
6143 |
6144 |
6145 |
6146 |
6147 |
6148 |
6149 |
6150 |
6151 |
6152 |
6153 |
6154 |
6155 |
6156 |
6157 |
6158 |
6159 |
6160 |
6161 |
6162 |
6163 |
6164 |
6165 |
6166 |
6167 |
6168 |
6169 |
6170 |
6171 |
6172 |
6173 |
6174 |
6175 |
6176 |
6177 |
6178 |
6179 |
6180 |
6181 |
6182 |
6183 |
6184 |
6185 |
6186 |
6187 |
6188 |
6189 |
6190 |
6191 |
6192 |
6193 |
6194 |
6195 |
6196 |
6197 |
6198 |
6199 |
6200 |
6201 |
6202 |
6203 |
6204 |
6205 |
6206 |
6207 |
6208 |
6209 |
6210 |
6211 |
6212 |
6213 |
6214 |
6215 |
6216 |
6217 |
6218 |
6219 |
6220 |
6221 |
6222 |
6223 |
6224 |
6225 |
6226 |
6227 |
6228 |
6229 |
6230 |
6231 |
6232 |
6233 |
6234 |
6235 |
6236 |
6237 |
6238 |
6239 |
6240 |
6241 |
6242 |
6243 |
6244 |
6245 |
6246 |
6247 |
6248 |
6249 |
6250 |
6251 |
6252 |
6253 |
6254 |
6255 |
6256 |
6257 |
6258 |
6259 |
6260 |
6261 |
6262 |
6263 |
6264 |
6265 |
6266 |
6267 |
6268 |
6269 |
6270 |
6271 |
6272 |
6273 |
6274 |
6275 |
6276 |
6277 |
6278 |
6279 |
6280 |
6281 |
6282 |
6283 |
6284 |
6285 |
6286 |
6287 |
6288 |
6289 |
6290 |
6291 |
6292 |
6293 |
6294 |
6295 |
6296 |
6297 |
6298 |
6299 |
6300 |
6301 |
6302 |
6303 |
6304 |
6305 |
6306 |
6307 |
6308 |
6309 |
6310 |
6311 |
6312 |
6313 |
6314 |
6315 |
6316 |
6317 |
6318 |
6319 |
6320 |
6321 |
6322 |
6323 |
6324 |
6325 |
6326 |
6327 |
6328 |
6329 |
6330 |
6331 |
6332 |
6333 |
6334 |
6335 |
6336 |
6337 |
6338 |
6339 |
6340 |
6341 |
6342 |
6343 |
6344 |
6345 |
6346 |
6347 |
6348 |
6349 |
6350 |
6351 |
6352 |
6353 |
6354 |
6355 |
6356 |
6357 |
6358 |
6359 |
6360 |
6361 |
6362 |
6363 |
6364 |
6365 |
6366 |
6367 |
6368 |
6369 |
6370 |
6371 |
6372 |
6373 |
6374 |
6375 |
6376 |
6377 |
6378 |
6379 |
6380 |
6381 |
6382 |
6383 |
6384 |
6385 |
6386 |
6387 |
6388 |
6389 |
6390 |
6391 |
6392 |
6393 |
6394 |
6395 |
6396 |
6397 |
6398 |
6399 |
6400 |
6401 |
6402 |
6403 |
6404 |
6405 |
6406 |
6407 |
6408 |
6409 |
6410 |
6411 |
6412 |
6413 |
6414 |
6415 |
6416 |
6417 |
6418 |
6419 |
6420 |
6421 |
6422 |
6423 |
6424 |
6425 |
6426 |
6427 |
6428 |
6429 |
6430 |
6431 |
6432 |
6433 |
6434 |
6435 |
6436 |
6437 |
6438 |
6439 |
6440 |
6441 |
6442 |
6443 |
6444 |
6445 |
6446 |
6447 |
6448 |
6449 |
6450 |
6451 |
6452 |
6453 |
6454 |
6455 |
6456 |
6457 |
6458 |
6459 |
6460 |
6461 |
6462 |
6463 |
6464 |
6465 |
6466 |
6467 |
6468 |
6469 |
6470 |
6471 |
6472 |
6473 |
6474 |
6475 |
6476 |
6477 |
6478 |
6479 |
6480 |
6481 |
6482 |
6483 |
6484 |
6485 |
6486 |
6487 |
6488 |
6489 |
6490 |
6491 |
6492 |
6493 |
6494 |
6495 |
6496 |
6497 |
6498 |
6499 |
6500 |
6501 |
6502 |
6503 |
6504 |
6505 |
6506 |
6507 |
6508 |
6509 |
6510 |
6511 |
6512 |
6513 |
6514 |
6515 |
6516 |
6517 |
6518 |
6519 |
6520 |
6521 |
6522 |
6523 |
6524 |
6525 |
6526 |
6527 |
6528 |
6529 |
6530 |
6531 |
6532 |
6533 |
6534 |
6535 |
6536 |
6537 |
6538 |
6539 |
6540 |
6541 |
6542 |
6543 |
6544 |
6545 |
6546 |
6547 |
6548 |
6549 |
6550 |
6551 |
6552 |
6553 |
6554 |
6555 |
6556 |
6557 |
6558 |
6559 |
6560 |
6561 |
6562 |
6563 |
6564 |
6565 |
6566 |
6567 |
6568 |
6569 |
6570 |
6571 |
6572 |
6573 |
6574 |
6575 |
6576 |
6577 |
6578 |
6579 |
6580 |
6581 |
6582 |
6583 |
6584 |
6585 |
6586 |
6587 |
6588 |
6589 |
6590 |
6591 |
6592 |
6593 |
6594 |
6595 |
6596 |
6597 |
6598 |
6599 |
6600 |
6601 |
6602 |
6603 |
6604 |
6605 |
6606 |
6607 |
6608 |
6609 |
6610 |
6611 |
6612 |
6613 |
6614 |
6615 |
6616 |
6617 |
6618 |
6619 |
6620 |
6621 |
6622 |
6623 |
6624 |
6625 |
6626 |
6627 |
6628 |
6629 |
6630 |
6631 |
6632 |
6633 |
6634 |
6635 |
6636 |
6637 |
6638 |
6639 |
6640 |
6641 |
6642 |
6643 |
6644 |
6645 |
6646 |
6647 |
6648 |
6649 |
6650 |
6651 |
6652 |
6653 |
6654 |
6655 |
6656 |
6657 |
6658 |
6659 |
6660 |
6661 |
6662 |
6663 |
6664 |
6665 |
6666 |
6667 |
6668 |
6669 |
6670 |
6671 |
6672 |
6673 |
6674 |
6675 |
6676 |
6677 |
6678 |
6679 |
6680 |
6681 |
6682 |
6683 |
6684 |
6685 |
6686 |
6687 |
6688 |
6689 |
6690 |
6691 |
6692 |
6693 |
6694 |
6695 |
6696 |
6697 |
6698 |
6699 |
6700 |
6701 |
6702 |
6703 |
6704 |
6705 |
6706 |
6707 |
6708 |
6709 |
6710 |
6711 |
6712 |
6713 |
6714 |
6715 |
6716 |
6717 |
6718 |
6719 |
6720 |
6721 |
6722 |
6723 |
6724 |
6725 |
6726 |
6727 |
6728 |
6729 |
6730 |
6731 |
6732 |
6733 |
6734 |
6735 |
6736 |
6737 |
6738 |
6739 |
6740 |
6741 |
6742 |
6743 |
6744 |
6745 |
6746 |
6747 |
6748 |
6749 |
6750 |
6751 |
6752 |
6753 |
6754 |
6755 |
6756 |
6757 |
6758 |
6759 |
6760 |
6761 |
6762 |
6763 |
6764 |
6765 |
6766 |
6767 |
6768 |
6769 |
6770 |
6771 |
6772 |
6773 |
6774 |
6775 |
6776 |
6777 |
6778 |
6779 |
6780 |
6781 |
6782 |
6783 |
6784 |
6785 |
6786 |
6787 |
6788 |
6789 |
6790 |
6791 |
6792 |
6793 |
6794 |
6795 |
6796 |
6797 |
6798 |
6799 |
6800 |
6801 |
6802 |
6803 |
6804 |
6805 |
6806 |
6807 |
6
```

```

5198      |          -----
5199      |          1111          00      000000 E78/434
5200      |          1110          01      000182 E78/435
5201      |          1101          02      000125 E78/432
5202      |          1100          03      -----
5203      |          1011          04      125200 E88/412
5204      |          1010          05      -----
5205      |          1001          06      -----
5206      |          1000          07      125500 E88/408
5207      |          0111          10      125200 E88/408
5208      |          0110          11      -----
5209      |          0101          12      -----
5210      |          0100          13      125200 E88/412
5211      |              11          0011          14      -----
5212      |              10          0010          15      000125 E78/435
5213      |               01          0001          16      000182 E78/435
5214      |               00          0000          17      000000 E78/434
5215      |
5216
5217
5218      | - - - - -
5219
5220
5221      | TEST 102A VERIFIES THAT CSPD[02] WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5222      |      (000125), E78 TARGET (432). LOOKING FOR CSP ADDRESS BIT<01> STUCK ONE/ZERO,
5223      | OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR.
5224      | 5521:
5225      | TEST102A:
5226      |     PO,          LOAD-ENUA(ETARGET432),          INSTRS E78 OUTPUT
5227      |             LOAD=ERROR(TEST102A),          ERROR DIRECTORY KEY
5228      |             DCB=CTR(C12),          ICOMPARE AT TARGET
5229      |             NEXT, J/LOAD01=102A          |
5230      | (5521) DCB[1,0,0,1,0,0,0] BM[0011..00,11..11,00..011..010...0,0,0,0..0,0...0,0000...0..0000,0...11,000...101,011,000]
5231      | 5530:
5232      | LOAD01=102A:
5233      |     P3,          CSPD[01]=EMIT, EMIT/000182,          INSTRS DATA PATTERN:
5234      |     NEXT, J/LOAD02=102A          | (000182)=E78(428)
5235      | (5530) DCB[0,0,0,0,0,0,0] BM[0000..10,00..00,01..101..010...0,0,0,0..0,0...0,1110...1..0000,0...11,000...001,101,001]
5236      | 5151: 1(FREE)
5237      | LOAD02=102A:
5238      |     P3,          CSPD[02]=EMIT, EMIT/000125,          INSTRS DATA PATTERN:
5239      |     NEXT, J/LOAD04=102A          | (000125)=E78(432)
5240      | (5151) DCB[0,0,0,0,0,0,0] BM[0000..10,00..00,01..010..101...0,0,0,0..0,0...0,0101...1..0000,0...11,000...001,101,010]
5241      | 5152: 1(FREE)
5242      | LOAD04=102A:
5243      |     P3,          CSPD[04]=EMIT, EMIT/125200,          INSTRS DATA PATTERN:
5244      |     NEXT, J/LOAD10=102A          | (125200)=E88(412)
5245      | (5152) DCB[0,0,0,0,0,0,0] BM[1010..10,10..10,10..000..000...0,0,0,0..0,0...0,0101...1..0000,0...11,000...001,101,011]
5246      | 5153: 1(FREE)
5247      | LOAD10=102A:

```

```

5248      P3,      CSPD(10) _EMIT, EMIT/152500,      INSTRS DATA PATTERN;
5249      NEXT,    J/LOAD00-102A      ! (152500)=E88(405)
      (5153) DCS(0.00,0.0,0.0) BM(1101..10.01..01.01..000..000...0.0.0..0.0...0.0111...1..0000.0...11.000...001.101,1001)

5250      5154: 1(FREE)
5251      LOAD00-102A;
5252      P3,      CSPD(00) _EMIT, EMIT/000000,      INSTRS DATA PATTERN;
5253      NEXT,    J/LOADD102A      ! (000000)=E78(434)
5254      (5154) DCS(0.00,0.0,0.0) BM(0000..10.00..00.00..000..000...0.0.0..0.0...0.1111...1..0000.0...11.000...001.101,1011)
5255      5155: 1(FREE)
5256      LOADD102A;
5257      P2-T,    D_CSPD(D02), BSEL/B17,      IGET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5258      NEXT,    J/GOTEST102A      !
5259      (5155) DCS(0.00,0.0,0.0) BM(1010..10.00..00.00..000..000...0.1.0..0.0...0.1101...0..0000.0...11.000...001.101,1101)
5260      5156: 1(FREE)
5261      GOTEST102A;
5262      SETUP,   RETURN/TEST102B,      IGO TO SUBR WHICH:
5263      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5264      (5156) DCS(0.00,0.0,0.0) BM(0101..00.10..11.11..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5265      5157: 1(FREE)
5266      TEST102B;
5267      PO,      LOAD-ENUA(ZTARGET405),      INSTRS E88 OUTPUT
5268      LOAD-ERRR(TEST102B),      !ERROR DIRECTORY KEY
5269      DCS-CTR(C7.),      !COMPARE AT TARGET
5270      NEXT,    J/LOADD102B      !
5271      (5574) DCS(1.00,1.0,0.0) BM(1000..00.11..11.00..000..101...0.0.0..0.0...0.0000...0..0000.0...11.000...001.101,1111)
5272      5157: 1(FREE)
5273      LOADD102B;
5274      P2-T,    D_CSPD(D10), BSEL/B17,      IGET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5275      NEXT,    J/GOTEST102B      !
5276      (5157) DCS(0.00,0.0,0.0) BM(1010..10.00..00.00..000..000...0.1.0..0.0...0.0111...0..0000.0...11.000...001.110,0001)
5277      5160: 1(FREE)
5278      GOTEST102B;
5279      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5280      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5281      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5282      5160: 1(FREE)
5283      GOTEST102B;
5284      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5285      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5286      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5287      5160: 1(FREE)
5288      GOTEST102B;
5289      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5290      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5291      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5292      5160: 1(FREE)
5293      GOTEST102B;
5294      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5295      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5296      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5297      5160: 1(FREE)
5298      GOTEST102B;
5299      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5300      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5301      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5302      5160: 1(FREE)
5303      GOTEST102B;
5304      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5305      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5306      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5307      5160: 1(FREE)
5308      GOTEST102B;
5309      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5310      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5311      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5312      5160: 1(FREE)
5313      GOTEST102B;
5314      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5315      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5316      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5317      5160: 1(FREE)
5318      GOTEST102B;
5319      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5320      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5321      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5322      5160: 1(FREE)
5323      GOTEST102B;
5324      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5325      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5326      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5327      5160: 1(FREE)
5328      GOTEST102B;
5329      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5330      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5331      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5332      5160: 1(FREE)
5333      GOTEST102B;
5334      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5335      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5336      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5337      5160: 1(FREE)
5338      GOTEST102B;
5339      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5340      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5341      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5342      5160: 1(FREE)
5343      GOTEST102B;
5344      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5345      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5346      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5347      5160: 1(FREE)
5348      GOTEST102B;
5349      SETUP,   RETURN/TEST102C,      IGO TO SUBR WHICH:
5350      NEXT,    CALL(DINTOIR-5)      ! D -> DBUF -> IR, THEN BUT(INSTRS)
5351      (5160) DCS(0.00,0.0,0.0) BM(0101..00.10..11.10..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.111,0111)
5352      5160: 1(FREE)

```

```

5293
5294
5295
5296
5297 | - - - - -
5298
5299 1 TEST 102C VERIFIES THAT CSPD(04) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5300 1 (125200), E88 TARGET (412). LOOKING FOR CSP ADDRESS BIT<02> STUCK ONE/ZERO,
5301 1 OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR,
5302 55641
5303 TEST102C:
5304     PO,          LOAD=ENUA(ZTARGET412),          1INSTRS E88 OUTPUT
5305                LOAD=ERROR(TEST102C),          1ERROR DIRECTORY KEY
5306                DCS=CTR(C7.),          1COMPARE AT TARGET
5307                BUMP=VERIFY,          1COUNT
5308                NEXT, J/LOADD102C          1
5309 (5564) DCS[1,00,1,0,0,1] BM[1000,00,11,11,00,001,010,00,0,0,0,0,0,0,0,0000,00,0,0000,0,11,1000,0001,110,001]
5310
5311 5161: 1(FREE)
5312 LOADD102C:
5313     P2-T,      D_CSPD(D04), BSEL/B17,          1GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5314     NEXT,      J/GOTEST102C          1
5315 (5161) DCS[0,00,0,0,0,0] BM[1010,10,00,00,00,000,000,00,1,0,0,0,0,0,0,1011,00,0000,0,11,1000,0001,110,010]
5316
5317 5162: 1(FREE)
5318 GOTEST102C:
5319     SETUP,     RETURN/TEST102D,          1GO TO SUBR WHICH:
5320     NEXT,      CALLDINTDIR=5,          1 D -> DBUF -> IR, THEN BUT(INSTRS)
5321 (5162) DCS[0,00,0,0,0,0] BM[0101,00,10,11,01,100,111,00,0,0,0,0,0,0,0,0000,00,0,0000,0,11,100,0010,111,011]
5322
5323 | - - - - -
5324
5325 1 TEST 102D VERIFIES THAT CSPD(01) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5326 1 (000152), E78 TARGET (425). LOOKING FOR CSP ADDRESS BIT<00> STUCK ONE/ZERO,
5327 1 OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR,
5328 55541
5329 TEST102D:
5330     PO,          LOAD=ENUA(ZTARGET425),          1INSTRS E88 OUTPUT
5331                LOAD=ERROR(TEST102D),          1ERROR DIRECTORY KEY
5332                DCS=CTR(C7.),          1COMPARE AT TARGET
5333     NEXT,      J/LOADD102D          1
5334 (5554) DCS[1,00,1,0,0,0] BM[1000,00,11,11,00,010,101,00,0,0,0,0,0,0,0,0000,00,0,0000,0,11,1000,0001,110,011]
5335
5336 5163: 1(FREE)
5337 LOADD102D:
5338     P2-T,      D_CSPD(D01), BSEL/B17,          1GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5339     NEXT,      J/GOTEST102D          1
5340 (5163) DCS[0,00,0,0,0,0] BM[1010,10,00,00,00,000,000,00,1,0,0,0,0,0,0,1110,00,0000,0,11,1000,0001,110,100]
5341
5342

```

```

5341 5164: 1(FREE)
5342 G0TEST102D:
5343 SFTMP, RETURN/SCOPE102, IGO TO SUBR WHICH:
5344 NFXT, CALLIDINTOIR=5 I D -> DBUF -> IR, THEN BUT(INSTR5)
(5164) DCS[0.00,0.0,0.0] BM[0101..00.00..11.10..101..111...0.0,0.0,0.0...0.0000...0..0000,0...11.100...010,111,011]
5345
5346
5347
5348
5349 5165: 1(FREE)
5350 SCOPE102:
5351 NFXT, BUTD[SCOPE], INO ERROR: "TEST103A" (+1, WORDS)
5352 J/TEST103A I ERROR: "LOAD01-102A" {-16, WORDS}
(5165) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000,0...11.000...101,011,001]
5353
5354
5355
5356
5357
5358 !THE FOLLOWING SET OF FOUR TESTS VERIFIES THAT THE CSP, AND THE CSP ADDRESS FIELD *CSPADDR*
5359 !HAS NO STUCK ONE BITS, AND THAT THE EMIT -> CSP -> D -> DBUF -> IR DATAPATH
5360 !IS FULLY FUNCTIONAL.
5361
5362
5363
5364
5365
5366 !TEST 103A VERIFIES THAT CSPD(13) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5367 ! (125200), E88 TARGET (412). LOOKING FOR CSP ADDRESS BIT<02> STUCK ONE/ZERO,
5368 ! OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR,
5369 5531:
5370 TEST103A:
5371 PO, LOAD=ENUA(ZTARGET412), INSTRS E88 OUTPUT
5372 LOAD=ERRD(TEST103A), IERROR DIRECTORY KEY
5373 DCS=CTR(C12,), ICOMPARE AT TARGET
5374 NFXT, J/LOAD16-103A I
(5531) DCS[1.00,1.0,0.0] BM[0011..00.11..11.00..001..010...0.0,0.0,0.0...0.0000...0..0000,0...11.000...101,100,110]
5375
5376 5546:
5377 LOAD16-103A:
5378 PO, BUMP=VERIFY, ICOUNT
5379 P3, CSPD(16) EMIT, EMIT/000125, INSTRS DATA PATTERN:
5380 NFXT, J/LOAD15-103A I (000125)=E78(425)
(5546) DCS[0.00,0.0,0.0,1] BM[0000..10.00..00.01..101..010...0.0,0.0,0.0...0.0001...1..0000,0...11.000...001,110,110]
5381
5382 5166: 1(FREE)
5383
5384 LOAD15-103A:
5385 P3, CSPD(15) EMIT, EMIT/000125, INSTRS DATA PATTERN:
5386 NFXT, J/LOAD13-103A I (000125)=E78(432)
(5166) DCS[0.00,0.0,0.0] BM[0000..10.00..00.01..010..101...0.0,0.0,0.0...0.0010...1..0000,0...11.000...001,110,111]
5387
5388 5167: 1(FREE)
5389 LOAD13-103A:

```

```
5389 P3, CSPD(13)_EMIT, EMIT/125200, INSTRS DATA PATTERN:
5390 NEXT, J/LOAD07-103A (125200)=E88(412)
(5167) DCS(0.00,0.0,0.0) BM(1010..10.10..10.10..000..000...0.0,0.0,0.0...0.0000...1..0000,0...11.000...001.111,000)
5391 5170: 1(FREE)
5392 LOAD07-103A:
5393 P3, CSPD(07)_EMIT, EMIT/152500, INSTRS DATA PATTERN:
5394 NEXT, J/LOAD17-103A (152500)=E88(405)
(5170) DCS(0.00,0.0,0.0) BM(1101..10.01..01.01..000..000...0.0,0.0,0.0...0.0000...1..0000,0...11.000...001.111,001)
5395 5171: 1(FREE)
5396 LOAD17-103A:
5397 P3, CSPD(17)_EMIT, EMIT/000000, INSTRS DATA PATTERN:
5398 NEXT, J/LOAD07-103A (000000)=E78(434)
5399 (5171) DCS(0.00,0.0,0.0) BM(0000..10.00..00.00..000..000...0.0,0.0,0.0...0.0000...1..0000,0...11.000...001.111,010)
5400 IF THIS DATA USED, CSP ADDRESSING ERROR
5401 5172: 1(FREE)
5402 LOAD07-103A:
5403 P2=T, D_CSPD(D13), BSEL/B17, GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5404 NEXT, J/GOTEST103A
(5172) DCS(0.00,0.0,0.0) BM(1010..10.00..00.00..000..000...0.1,0.0,0.0...0.0000...0..0000,0...11.000...001.111,011)
5405 5173: 1(FREE)
5406 GOTEST103A:
5407 SETUP, RETURN/TEST103B, GO TO SUBR WHICH:
5408 NEXT, CALL(DINTOIR=5) D -> DBUF -> IR, THEN BUT(INSTRS)
(5173) DCS(0.00,0.0,0.0) BM(0101..00.10..11.11..010..111...0.0,0.0,0.0...0.0000...0..0000,0...11.100...010.111,011)
5409 5410 5411 5412 5413 5414 5415 5416 5417 5418
5419 1 - - - - -
5420 1 TEST 103B VERIFIES THAT CSPD(15) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5421 1 (000125), E78 TARGET (432). LOOKING FOR CSP ADDRESS BIT<01> STUCK ONE/ZERO,
5422 1 OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR,
5423 5572:
5424 TEST103B:
5425 P0, LOAD=ENUA(ZTARGET432), INSTRS E78 OUTPUT
5426 LOAD=ERROR(103B), ERROR DIRECTORY KEY
5427 DCS=CTR(C7.), COMPARE AT TARGET
(5572) DCS(1.00,1.0,0.0) BM(1000..00.11..11.00..011..010...0.0,0.0,0.0...0.0000...0..0000,0...11.000...001.111,100)
5428 5429 5430 5431 5432 5433 5434
5435 5174: 1(FREE)
5436 LOAD07-103B:
5437 P2=T, D_CSPD(D15), BSEL/B17, GET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5438 NEXT, J/GOTEST103B
(5174) DCS(0.00,0.0,0.0) BM(1010..10.00..00.00..000..000...0.1,0.0,0.0...0.0010...0..0000,0...11.000...001.111,101)
5439 5440 5441 5442 5443 5444 5445 5446 5447 5448 5449 5450 5451 5452 5453 5454 5455 5456 5457 5458 5459 5460 5461 5462 5463 5464 5465 5466 5467 5468 5469 5470 5471 5472 5473 5474 5475 5476 5477 5478 5479 5480 5481
```

```
5435 GOTEST103B:
5436 SETUP, RETURN/TEST103C, GO TO SUBR WHICH:
5437 NEXT, CALL(DINTOIR=5) D -> DBUF -> IR, THEN BUT(INSTRS)
(5175) DCS(0.00,0.0,0.0) BM(0101..00.10..11.11..110..111...0.0,0.0,0.0...0.0000...0..0000,0...11.100...010.111,011)
5438 5439 5440 5441 5442 5443 5444 5445 5446 5447 5448 5449 5450 5451 5452 5453 5454 5455 5456 5457 5458 5459 5460 5461 5462 5463 5464 5465 5466 5467 5468 5469 5470 5471 5472 5473 5474 5475 5476 5477 5478 5479 5480 5481
5482 1 - - - - -
5483 1 TEST 103C VERIFIES THAT CSPD(16) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5484 1 (000152), E78 TARGET (425). LOOKING FOR CSP ADDRESS BIT<00> STUCK ONE/ZERO,
5485 1 OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR,
5486 5576:
5487 TEST103C:
5488 P0, LOAD=ENUA(ZTARGET425), INSTRS E78 OUTPUT
5489 LOAD=ERROR(103C), ERROR DIRECTORY KEY
5490 DCS=CTR(C7.), COMPARE AT TARGET
5491 BUMP=VERIFY, COUNT
(5576) DCS(1.00,1.0,0.0) BM(1000..00.11..11.00..010..101...0.0,0.0,0.0...0.0000...0..0000,0...11.000...001.111,110)
5492 5493 5494 5495 5496 5497 5498 5499 5500 5501 5502 5503 5504 5505 5506 5507 5508 5509 5510 5511 5512 5513 5514 5515 5516 5517 5518 5519 5520 5521 5522 5523 5524 5525 5526 5527 5528 5529 5530 5531 5532 5533 5534 5535 5536 5537 5538 5539 5540 5541 5542 5543 5544 5545 5546 5547 5548 5549 5550 5551 5552 5553 5554 5555 5556 5557 5558 5559 5560 5561 5562 5563 5564 5565 5566 5567 5568 5569 5570 5571 5572 5573 5574 5575 5576 5577 5578 5579 5580 5581 5582 5583 5584 5585 5586 5587 5588 5589 5590 5591 5592 5593 5594 5595 5596 5597 5598 5599 5600 5601 5602 5603 5604 5605 5606 5607 5608 5609 5610 5611 5612 5613 5614 5615 5616 5617 5618 5619 5620 5621 5622 5623 5624 5625 5626 5627 5628 5629 5630 5631 5632 5633 5634 5635 5636 5637 5638 5639 5640 5641 5642 5643 5644 5645 5646 5647 5648 5649 5650 5651 5652 5653 5654 5655 5656 5657 5658 5659 5660 5661 5662 5663 5664 5665 5666 5667 5668 5669 5670 5671 5672 5673 5674 5675 5676 5677 5678 5679 5680 5681 5682 5683 5684 5685 5686 5687 5688 5689 5690 5691 5692 5693 5694 5695 5696 5697 5698 5699 5700 5701 5702 5703 5704 5705 5706 5707 5708 5709 5710 5711 5712 5713 5714 5715 5716 5717 5718 5719 5720 5721 5722 5723 5724 5725 5726 5727 5728 5729 5730 5731 5732 5733 5734 5735 5736 5737 5738 5739 5740 5741 5742 5743 5744 5745 5746 5747 5748 5749 5750 5751 5752 5753 5754 5755 5756 5757 5758 5759 5760 5761 5762 5763 5764 5765 5766 5767 5768 5769 5770 5771 5772 5773 5774 5775 5776 5777 5778 5779 5780 5781 5782 5783 5784 5785 5786 5787 5788 5789 5790 5791 5792 5793 5794 5795 5796 5797 5798 5799 5800 5801 5802 5803 5804 5805 5806 5807 5808 5809 5810 5811 5812 5813 5814 5815 5816 5817 5818 5819 5820 5821 5822 5823 5824 5825 5826 5827 5828 5829 5830 5831 5832 5833 5834 5835 5836 5837 5838 5839 5840 5841 5842 5843 5844 5845 5846 5847 5848 5849 5850 5851 5852 5853 5854 5855 5856 5857 5858 5859 5860 5861 5862 5863 5864 5865 5866 5867 5868 5869 5870 5871 5872 5873 5874 5875 5876 5877 5878 5879 5880 5881 5882 5883 5884 5885 5886 5887 5888 5889 5890 5891 5892 5893 5894 5895 5896 5897 5898 5899 5900 5901 5902 5903 5904 5905 5906 5907 5908 5909 5910 5911 5912 5913 5914 5915 5916 5917 5918 5919 5920 5921 5922 5923 5924 5925 5926 5927 5928 5929 5930 5931 5932 5933 5934 5935 5936 5937 5938 5939 5940 5941 5942 5943 5944 5945 5946 5947 5948 5949 5950 5951 5952 5953 5954 5955 5956 5957 5958 5959 5960 5961 5962 5963 5964 5965 5966 5967 5968 5969 5970 5971 5972 5973 5974 5975 5976 5977 5978 5979 5980 5981 5982 5983 5984 5985 5986 5987 5988 5989 5990 5991 5992 5993 5994 5995 5996 5997 5998 5999 6000 6001 6002 6003 6004 6005 6006 6007 6008 6009 6010 6011 6012 6013 6014 6015 6016 6017 6018 6019 6020 6021 6022 6023 6024 6025 6026 6027 6028 6029 6030 6031 6032 6033 6034 6035 6036 6037 6038 6039 6040 6041 6042 6043 6044 6045 6046 6047 6048 6049 6050 6051 6052 6053 6054 6055 6056 6057 6058 6059 6060 6061 6062 6063 6064 6065 6066 6067 6068 6069 6070 6071 6072 6073 6074 6075 6076 6077 6078 6079 6080 6081 6082 6083 6084 6085 6086 6087 6088 6089 6090 6091 6092 6093 6094 6095 6096 6097 6098 6099 6100 6101 6102 6103 6104 6105 6106 6107 6108 6109 6110 6111 6112 6113 6114 6115 6116 6117 6118 6119 6120 6121 6122 6123 6124 6125 6126 6127 6128 6129 6130 6131 6132 6133 6134 6135 6136 6137 6138 6139 6140 6141 6142 6143 6144 6145 6146 6147 6148 6149 6150 6151 6152 6153 6154 6155 6156 6157 6158 6159 6160 6161 6162 6163 6164 6165 6166 6167 6168 6169 6170 6171 6172 6173 6174 6175 6176 6177 6178 6179 6180 6181 6182 6183 6184 6185 6186 6187 6188 6189 6190 6191 6192 6193 6194 6195 6196 6197 6198 6199 6200 6201 6202 6203 6204 6205 6206 6207 6208 6209 6210 6211 6212 6213 6214 6215 6216 6217 6218 6219 6220 6221 6222 6223 6224 6225 6226 6227 6228 6229 6230 6231 6232 6233 6234 6235 6236 6237 6238 6239 6240 6241 6242 6243 6244 6245 6246 6247 6248 6249 6250 6251 6252 6253 6254 6255 6256 6257 6258 6259 6260 6261 6262 6263 6264 6265 6266 6267 6268 6269 6270 6271 6272 6273 6274 6275 6276 6277 6278 6279 6280 6281 6282 6283 6284 6285 6286 6287 6288 6289 6290 6291 6292 6293 6294 6295 6296 6297 6298 6299 6300 6301 6302 6303 6304 6305 6306 6307 6308 6309 6310 6311 6312 6313 6314 6315 6316 6317 6318 6319 6320 6321 6322 6323 6324 6325 6326 6327 6328 6329 6330 6331 6332 6333 6334 6335 6336 6337 6338 6339 6340 6341 6342 6343 6344 6345 6346 6347 6348 6349 6350 6351 6352 6353 6354 6355 6356 6357 6358 6359 6360 6361 6362 6363 6364 6365 6366 6367 6368 6369 6370 6371 6372 6373 6374 6375 6376 6377 6378 6379 6380 6381 6382 6383 6384 6385 6386 6387 6388 6389 6390 6391 6392 6393 6394 6395 6396 6397 6398 6399 6400 6401 6402 6403 6404 6405 6406 6407 6408 6409 6410 6411 6412 6413 6414 6415 6416 6417 6418 6419 6420 6421 6422 6423 6424 6425 6426 6427 6428 6429 6430 6431 6432 6433 6434 6435 6436 6437 6438 6439 6440 6441 6442 6443 6444 6445 6446 6447 6448 6449 6450 6451 6452 6453 6454 6455 6456 6457 6458 6459 6460 6461 6462 6463 6464 6465 6466 6467 6468 6469 6470 6471 6472 6473 6474 6475 6476 6477 6478 6479 6480 6481 6482 6483 6484 6485 6486 6487 6488 6489 6490 6491 6492 6493 6494 6495 6496 6497 6498 6499 6500 6501 6502 6503 6504 6505 6506 6507 6508 6509 6510 6511 6512 6513 6514 6515 6516 6517 6518 6519 6520 6521 6522 6523 6524 6525 6526 6527 6528 6529 6530 6531 6532 6533 6534 6535 6536 6537 6538 6539 6540 6541 6542 6543 6544 6545 6546 6547 6548 6549 6550 6551 6552 6553 6554 6555 6556 6557 6558 6559 6560 6561 6562 6563 6564 6565 6566 6567 6568 6569 6570 6571 6572 6573 6574 6575 6576 6577 6578 6579 6580 6581 6582 6583 6584 6585 6586 6587 6588 6589 6590 6591 6592 6593 6594 6595 6596 6597 6598 6599 6600 6601 6602 6603 6604 6605 6606 6607 6608 6609 6610 6611 6612 6613 6614 6615 6616 6617 6618 6619 6620 6621 6622 6623 6624 6625 6626 6627 6628 6629 6630 6631 6632 6633 6634 6635 6636 6637 6638 6639 6640 6641 6642 6643 6644 6645 6646 6647 6648 6649 6650 6651 6652 6653 6654 6655 6656 6657 6658 6659 6660 6661 6662 6663 6664 6665 6666 6667 6668 6669 6670 6671 6672 6673 6674 6675 6676 6677 6678 6679 6680 6681 6682 6683 6684 6685 6686 6687 6688 6689 6690 6691 6692 6693 6694 6695 6696 6697 6698 6699 6700 6701 6702 6703 6704 6705 6706 6707 6708 6709 6710 6711 6712 6713 6714 6715 6716 6717 6718 6719 6720 6721 6722 6723 6724 6725 6726 6727 6728 6729 6730 6731 6732 6733 6734 6735 6736 6737 6738 6739 6740 6741 6742 6743 6744 6745 6746 6747 6748 6749 6750 6751 6752 6753 6754 6755 6756 6757 6758 6759 6760 6761 6762 6763 6764 6765 6766 6767 6768 6769 6770 6771 6772 6773 6774 6775 6776 6777 6778 6779 6780 6781 6782 6783 6784 6785 6786 6787 6788 6789 6790 6791 6792 6793 6794 6795 6796 6797 6798 6799 6800 6801 6802 6803 6804 6805 6806 6807 6808 6809 6810 6811 6812 6813 6814 6815 6816 6817 6818 6819 6820 6821 6822 6823 6824 6825 6826 6827 6828 6829 6830 6831 6832 6833 6834 6835 6836 6837 6838 6839 6840 6841 6842 6843 6844 6845 6846 6847 6848 6849 6850 6851 6852 6853 6854 6855 6856 6857 6858 6859 6860 6861 6862 6863 6864 6865 6866 6867 6868 6869 6870 6871 6872 6873 6874 6875 6876 6877 6878 6879 6880 6881 6882 6883 6884 6885 6886 6887 6888 6889 6890 6891 6892 6893 6894 6895 6896 6897 6898 6899 6900 6901 6902 6903 6904 6905 6906 6907 6908 6909 6910 6911 6912 6913 6914 6915 6916 6917 6918 6919 6920 6921 6922 6923 6924 6925 6926 6927 6928 6929 6930 6931 6932 6933 6934 6935 6936 6937 6938 6939 6940 6941 6942 6943 6944 6945 6946 6947 6948 6949 6950 6951 6952 6953 6954 6955 6956 6957 6958 6959 6960 6961 6962 6963 6964 6965 6966 6967 6968 6969 6970 6971 6972 6973 6974 6975 6976 6977 6978 6979 6980 6981 6982 6983 6984 6985 6986 6987 6988 6989 6990 6991 6992 6993 6994 6995 6996 6997 6998 6999 7000 7001 7002 7003 7004 7005 7006 7007 7008 7009 7010 7011 7012 7013 7014 7015 7016 7017 7018 7019 7020 7021 7022 7023 7024 7025 7026 7027 7028 7029 7030 7031 7032 7033 7034 7035 7036 7037 7038 7039 7040 7041 7042 7043 7044 7045 7046 7047 7048 7049 7050 7051 7052 7053 7054 7055 7056 7057 7058 7059 7060 7061 7062 7063 7064 7065 7066 7067 7068 7069 7070 7071 7072 7073 7074 7075 7076 7077 7078 7079 7080 7081 7082 7083 7084 7085 7086 7087 7088 7089 7090 7091 7092 7093 7094 7095 7096 7097 7098 7099 7100 7101 7102 7103 7104 7105 7106 7107 7108 7109 7110 7111 7112 7113 7114 7115 7116 7117 7118 7119 7120 7121 7122 7123 7124 7125 7126 7127 7128 7129 7130 7131 7132 7133 7134 7135 7136 7137 7138 7139 7140 7141 7142 7143 7144 7145 7146 7147 7148 7149 7150 7151 7152 7153 7154 7155 7156 7157 7158 7159 7160 7161 7162 7163 7164 7165 7166 7167 7168 7169 7170 7171 7172 7173 7174 7175 7176 7177 7178 7179 7180 7181 7182 7183 7184 7185 7186 7187 7188 7189 7190 7191 7192 7193 7194 7195 7196 7197 7198 7199 7200 7201 7202 7203 7204 7205 7206 7207 7208 7209 7210 7211 7212 7213 7214 7215 7216 7217 7218 7219 7220 7221 7222 7223 7224 7225 7226 7227 7228 7229 7230 7231 7232 7233 7234 7235 7236 7237 7238 7239 7240 7241 7242 7243 7244 7245 7246 7247 7248 7249 7250 7251 7252 7253 7254 7255 7256 7257 7258 7259 7260 7261 7262 7263 7264 7265 7266 7267 7268 7269 7270 7271 7272 7273 7274 7275 7276 7277 7278 7279 7280 7281 7282 7283 7284 7285 7286 7287 7288 7289 7290 7291 7292 7293 7294 7295 7296 7297 7298 7299 7300 7301 7302 7303 7304 7305 7306 7307 7308 7309 7310 7311 7312 7313 7314 7315 7316 7317 7318 7319 7320 7321 7322 7323 7324 7325 7326 7327 7328 7329 7330 7331 7332 7333 7334 7335 7336 7337 7338 7339 7340 7341 7342 7343 7344 7345 7346 7347 7348 7349 7350 7351 7352 7353 7354 7355 7356 7357 7358 7359 7360 7361 7362 7363 7364 7365 7366 7367 7368 7369 7370 7371 7372 7373 7374 7375 7376 7377 7378 7379 7380 7381 7382 7383 7384 7385 7386 7387 7388 7389 7390 7391 7392 7393 7394 7395 7396 7397 7398 7399 7400 7401 7402 7403 7404 7405 7406 7407 7408 7409 7410 7411 7412 7413 7414 7415 7416 7417 7418 7419 7420 74
```

```

5482 5200: 1(FREE)
5483 LOADD103D:
5484 P2-T, D_CSPD(D07), BSEL/D17, IGET CSP LOC VIA CSPADDR, BASCON FIELD "00"
5485 NEXT, J/GOTEST103D
(5200) DCS[0.00,0.0,0.0] BM[1010..10.00..00.00..000..000...0.1.0..0..0.1000...0.0000.0...11.000...010.000.001]
5486
5487 5201: 1(FREE)
5488 GOTEST103D:
5489 SETUP, RETURN/SCOPE103, IGO TO SUBR WHICH:
5490 NEXT, CALL[DINTOIR=5] I D -> DBUF -> IR, THEN BUT(INSTR5)
(5201) DCS[0.00,0.0,0.0] BM[0101..00.01..00.00..010..111...0.0.0..0..0.0000...0.0000.0...11.100...010.111.011]
5491
5492
5493
5494
5495 5202: 1(FREE)
5496 SCOPE103:
5497 NEXT, BUTD[SCOPE], INO ERROR: "TEST104A" (+1, WORDS)
5498 J/TEST104A I ERROR: "LOAD10-103A" (-16, WORDS)
(5202) DCS[0.00,0.1,0.0] BM[0000..00.00..00.00..000..000...0.0.0..0..0.0000...0.0000.0...11.100...101.100.111]
5499
5500
5501 I - - - - -
5502
5503 I THE FOLLOWING SET OF TWO TESTS VERIFIES THAT THE CSP, AND THE CSP ADDRESS FIELD "BASCON"
5504 I HAS NO STUCK ZERO/ONE BITS, AND THAT THE EMIT -> CSP -> D -> DBUF -> IR DATAPATH
5505 I IS FULLY FUNCTIONAL.
5506
5507 I - - - - -
5508
5509
5510
5511 I TEST 104A VERIFIES THAT CSPB(16) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5512 I (000142), E78 TARGET (425), LOOKING FOR CSP ADDRESS BIT<00> STUCK ONE/ZERO,
5513 I OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR,
5514 I THIS TEST USES THE "BASCON" ADDRESS MODE FOR THE CSP,
5515 5547:
5516 TEST104A:
5517 PO, LOAD-ENUA(ZTARGET425), INSTRS E78 OUTPUT
5518 LOAD-ERROR(TEST104A), IERROR DIRECTORY KEY
5519 DCS-CTR(C7), ICOMPARE AT TARGET
5520 NEXT, J/LOADD104A
(5547) DCS[1.00,1.0,0.0] BM[1000..00.11..11.00..010..101...0.0.0..0..0.0000...0.0000.0...11.000...101.100.000]
5521
5522 5540:
5523 LOADD104A:
5524 P2-T, D_CSPB(B16), CSPADDR/D17, IGET CSP LOC VIA BASCON, CSPADDR FIELD "0000"
5525 NEXT, J/GOTEST104A
(5540) DCS[0.00,0.0,0.0] BM[1010..11.01..00.00..000..000...0.1.0..0..0.0000...0.0000.0...11.000...010.000.011]
5526
5527 5203: 1(FREE)
5528 GOTEST104A:
5529 SETUP, RETURN/TEST104B, IGO TO SUBR WHICH:

```

```

5530 NEXT, CALL[DINTOIR=5] I D -> DBUF -> IR, THEN BUT(INSTR5)
(5203) DCS[0.00,0.0,0.0] BM[0101..00.11..01.00..110..111...0.0.0..0..0.0000...0.0000.0...11.100...010.111.011]
5531
5532
5533
5534
5535
5536 I - - - - -
5537
5538 I TEST 104B VERIFIES THAT CSPB(15) WAS WRITTEN WITH THE UNIQUE INSTRS PATTERN:
5539 I (000125), F78 TARGET (432), LOOKING FOR CSP ADDRESS BIT<01> STUCK ONE/ZERO,
5540 I OR ERRORS IN DATAPATH FROM EMIT -> CSP -> ALU-B -> D -> DBUF -> IR,
5541 I THIS TEST USES THE "BASCON" ADDRESS MODE FOR THE CSP,
5542 5646:
5543 TEST104B:
5544 PO, LOAD-ENUA(ZTARGET432), INSTRS E78 OUTPUT
5545 LOAD-ERROR(TEST104B), IERROR DIRECTORY KEY
5546 DCS-CTR(C7), ICOMPARE AT TARGET
5547 NEXT, J/LOADD104B
(5646) DCS[1.00,1.0,0.0] BM[1000..00.11..11.00..011..010...0.0.0..0..0.0000...0.0000.0...11.000...010.000.100]
5548
5549 5204: 1(FREE)
5550 LOADD104B:
5551 P2-T, D_CSPB(B15), CSPADDR/D17, IGET CSP LOC VIA BASCON, CSPADDR FIELD "0000"
5552 NEXT, J/GOTEST104B
(5204) DCS[0.00,0.0,0.0] BM[1010..11.10..00.00..000..000...0.1.0..0..0.0000...0.0000.0...11.000...010.000.101]
5553
5554 5205: 1(FREE)
5555 GOTEST104B:
5556 SETUP, RETURN/SCOPE104, IGO TO SUBR WHICH:
5557 NEXT, CALL[DINTOIR=5] I D -> DBUF -> IR, THEN BUT(INSTR5)
(5205) DCS[0.00,0.0,0.0] BM[0101..00.01..00.00..110..111...0.0.0..0..0.0000...0.0000.0...11.100...010.111.011]
5558
5559
5560
5561
5562 5206: 1(FREE)
5563 SCOPF104:
5564 PO, BUSDIN_EMIT-[1], IRESET PROC UCON
5565 EN-CLK-IR[15-00], I
5566 NEXT, BUTD[SCOPE], INO ERROR: "TEST105A" (+1, WORDS)
5567 J/TEST105A I ERROR: "LOAD104A" (-5, WORDS)
(5206) DCS[0.00,0.1,0.0] BM[0000..00.00..00.01..000..100...0.0.0..0..0.11001...0.0000.0...11.000...101.100.001]
5568
5569
5570
5571 I PAGE*****
5572
5573 .TNC * TEST105: SR CAN LOAD/STORE AS A REGISTER
5574
5575 I*****
5576 I*

```

```

5577 1* TESTS: 105
5578 1*
5579 1* FUNCTIONS:
5580 1*
5581 1* THE FOLLOWING TESTS VERIFY THE VALIDITY OF THE SR AS A TEMPORARY REGISTER,
5582 1* (IE, IT CAN BE LOADED/READ) IN ALL BIT POSITIONS, AND THAT THE ALU-A SIDE CAN
5583 1* PASS DATA INTO D.
5584 1*
5585 1*****
5586
5587
5588 1TEST 105A VERIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTRS PATTERN:
5589 1 (000125), E78 TARGET (432), AND TO VERIFY THE:
5590 1 DATAPATH FROM SR -> ALU-A -> D -> DBUF -> IR.
5591 5541:
5592 TEST105A:
5593 PO, LOAD-ENUA(ZTARGET432), INSTRS E78 OUTPUT
5594 LOAD-ERROR(TEST105A), ERROR DIRECTORY KEY
5595 DCS-CTR(C3), COMPARE AT TARGET
5596 BUMP-VERIFY, COUNT
5597 J/LOADSR105A
(5541) DCS[1.00.1.0.0.1] BM[0111.00.11.11.00.011.010.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.101.100.010]
5598
5599 5542:
5600 LOADSR105A:
5601 P2, RES_CSPD(D02), BITS<13:11>=00/0, WHICH IS; SR/LOAD, GUARD/DISABLED
5602 P3-T, SR_CSPD(D02), BSEL/B17, DATA IS (000125) = INSTRS E78 (432)
5603 NEXT, J/GOTEST105A
(5542) DCS[0.00.0.0.0.0] BM[1010.10.00.00.00.000.000.0.1.0.1.0.0.0.1101.0.1000.1.11.000.0.010.000.111]
5604
5605 5207: 1(FREE)
5606 GOTEST105A:
5607 PO, BUMP-VERIFY, COUNT
5608 SETUP, RETURN/TEST105A1, GO TO SUBR WHICH:
5609 NEXT, CALL(SRINTOIR-5) 1 SR -> D -> DBUF -> IR, THEN BUT(INSTRS)
(5207) DCS[0.00.0.0.0.1] BM[0101.00.11.00.01.000.0.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.111.010]
5610
5611
5612
5613
5614
5615 1 - - - - -
5616
5617 1TEST 105A1 VERIFIES THAT THE BUT(SR3=0) SEES THE "0101" IN THE SR.
5618 5610:
5619 TEST105A1:
5620 PO, LOAD-ENUA(ZTARGET405), BIT<3:0> = "0101"
5621 LOAD-ERROR(TEST105A1), ERROR DIRECTORY KEY
5622 DCS-CTR(C3), COMPARE AT TARGET
5623 NEXT, J/GOBUT105A1
(5610) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.000.0.101.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.010.001.000]
5624
5625 5210: 1(FREE)

```

```

5626 GORUT105A1:
5627 PO, BUMP-VERIFY, COUNT
5628 SETUP, RETURN/TEST105B, RETURN TO START OF NEXT SUBTEST
5629 NEXT, GOTN-PAGE(7), BUT TABLE
5630 J/BUTSR3=0 1SR<3:0> IN BIT<3:0>
(5710) DCS[0.00.0.0.0.1] BM[0101.00.11.01.01.010.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.111.110]
5631
5632
5633
5634
5635
5636 1 - - - - -
5637
5638 1TEST 105B VPPIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTRS PATTERN:
5639 1 (000152), E78 TARGET (425), AND TO VERIFY THE:
5640 1 DATAPATH FROM SR -> ALU-A -> D -> DBUF -> IR.
5641 5652:
5642 TEST105B:
5643 PO, LOAD-ENUA(ZTARGET425), INSTRS E78 OUTPUT
5644 LOAD-ERROR(TEST105B), ERROR DIRECTORY KEY
5645 DCS-CTR(C3), COMPARE AT TARGET
5646 NEXT, J/LOADSR105B
(5652) DCS[1.00.1.0.0.0] BM[0111.00.11.11.00.010.101.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.010.001.001]
5647
5648 5211: 1(FREE)
5649 LOADSR105B:
5650 P2-T, SR_CSPD(D01), BSEL/B17, DATA IS (000152) = INSTRS E78 (425)
5651 NEXT, J/GOTEST105B
(5211) DCS[0.00.0.0.0.0] BM[1010.10.00.00.00.000.000.0.0.0.1.0.0.0.1110.0.0.0000.0.11.000.0.010.001.010]
5652
5653 5212: 1(FREE)
5654 GOTEST105B:
5655 SETUP, RETURN/TEST105B1, GO TO SUBR WHICH:
5656 NEXT, CALL(SRINTOIR-5) 1 SR -> D -> DBUF -> IR, THEN BUT(INSTRS)
(5212) DCS[0.00.0.0.0.0] BM[0101.00.11.00.00.000.0.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.111.010]
5657
5658
5659
5660
5661
5662 1 - - - - -
5663
5664 1TEST 105B1 VERIFIES THAT THE BUT(SR3=0) SEES THE "1010" IN THE SR.
5665 5600:
5666 TEST105B1:
5667 PO, LOAD-ENUA(ZTARGET412), BIT<3:0> = "1010"
5668 LOAD-ERROR(TEST105B1), ERROR DIRECTORY KEY
5669 DCS-CTR(C3), COMPARE AT TARGET
5670 BUMP-VERIFY, COUNT
5671 NEXT, J/GOBUT105B1
(5600) DCS[1.00.1.0.0.1] BM[1100.00.11.11.00.001.010.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.010.001.011]
5672

```

```
5673 5213: 1(FREE)
5674 GORUT105B1;
5675     SETUP, RETURN/TEST105C,      IRETURN TO START OF NEXT SUBTEST
5676     NEXT,  GOTO-PAGE(7),          I BUT TABLE
5677     J/BUTSR3=0                     I SR<15:12> IN BITS<10>
(5213) DCS[0.00,0.0,0.0] BM[0101..00.11..01.00..010..111...0.0,0,0,0...0.0000...0.0000,0...11.100...010,111,110]

5678
5679
5680
5681
5682
5683
5684 1 - - - - -
5685
5686 ITEST 105C VERIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTRS PATTERN:
5687 1 (125200), E88 TARGET (412), AND TO VERIFY THE:
5688 1 DATAPATH FROM SR -> ALU-A -> D -> DBUF -> IR,
5689 5642:
5690 TEST105C:
5691     PO,      LOAD-ENUA(ZTARGET412),      INSTRS E88 OUTPUT
5692     LOAD-ERROR(TEST105C),                I ERROR DIRECTORY KEY
5693     DCS-CTR(C3.),                        I COMPARE AT TARGET
5694     NEXT,   J/LOADSR105C
(5642) DCS[1.00,1.0,0.0] BM[0111..00.11..11.00..001..010...0.0,0,0,0...0.0000...0.0000,0...11.000...010,001,100]

5695
5696 5214: 1(FREE)
5697 LOADSR105C:
5698     P2-T,   SR_CSPD(D04), BSEL/B17,      I DATA IS (125200) = INSTRS F88 (412)
5699     NEXT,   J/GOTEST105C
(5214) DCS[0.00,0.0,0.0] BM[1010..10.00..00.00..000..000...0.0,0,0,0...0.1011...0.0000,0...11.000...010,001,101]

5700
5701 5215: 1(FREE)
5702 GOTEST105C:
5703     SETUP, RETURN/TEST105D,      I GO TO SUBR WHICH:
5704     NEXT,   CALL(SRINTOIR-5),     I SR -> D -> DBUF -> IR, THEN BUT(INSTR5)
(5215) DCS[0.00,0.0,0.0] BM[0101..00.11..00.11..110..111...0.0,0,0,0...0.0000...0.0000,0...11.100...010,111,010]

5705
5706
5707
5708
5709
5710 1 - - - - -
5711
5712 ITEST 105D VERIFIES THAT THE SR CAN BE LOADED/READ WITH THE INSTRS PATTERN:
5713 1 (152500), E88 TARGET (405), AND TO VERIFY THE:
5714 1 DATAPATH FROM SR -> ALU-A -> D -> DBUF -> IR,
5715 5636:
5716 TEST105D:
5717     PO,      LOAD-ENUA(ZTARGET405),      INSTRS E88 OUTPUT
5718     LOAD-ERROR(TEST105D),                I ERROR DIRECTORY KEY
5719     DCS-CTR(C3.),                        I COMPARE AT TARGET
5720     NEXT,   J/LOADSR105D
I
```

```
(5636) DCS[1.00,1.0,0.0] BM[0111..00.11..11.00..000..101...0.0,0,0,0...0.0000...0.0000,0...11.000...010,001,110]

5721
5722 5216: 1(FREE)
5723 LOADSR105D:
5724     PO,      BUMP-VERIFY,          I COUNT
5725     P2-T,   SR_CSPD(D10), BSEL/B17,      I DATA IS (152500) = INSTRS E88 (405)
5726     NEXT,   J/GOTEST105D
(5216) DCS[0.00,0.0,0.1] BM[1010..10.00..00.00..000..000...0.0,0,0,0...0.0111...0.0000,0...11.000...010,001,111]

5727
5728 5217: 1(FREE)
5729 GOTEST105D:
5730     SETUP, RETURN/TEST105E,      I GO TO SUBR WHICH:
5731     NEXT,   CALL(SRINTOIR-5),     I SR -> D -> DBUF -> IR, THEN BUT(INSTR5)
(5217) DCS[0.00,0.0,0.0] BM[0101..00.11..00.10..000..111...0.0,0,0,0...0.0000...0.0000,0...11.100...010,111,010]

5732
5733
5734
5735
5736
5737 1 - - - - -
5738
5739 ITEST 105E VERIFIES THE DATAPATH FROM:
5740 1 EMIT -> CSP -> ALU-B -> SR -> ALU-A -> D -> DBUF -> IR,
5741 1 IS A VALID "1101" IN BITS<15:12>. SR VALUE = (152500) FROM PREVIOUS TEST.
5742 5620:
5743 TEST105E:
5744     PO,      LOAD-ENUA(ZTARGET415),      I BIT<15:12> = "1101"
5745     LOAD-ERROR(TEST105E),                I ERROR DIRECTORY KEY
5746     DCS-CTR(C3.),                        I COMPARE AT TARGET
5747     NEXT,   J/GOTEST105E
(5620) DCS[1.00,1.0,0.0] BM[1100..00.11..11.00..001..101...0.0,0,0,0...0.0000...0.0000,0...11.000...010,010,000]

5748
5749 5220: 1(FREE)
5750 GOTEST105E:
5751     SETUP, RETURN/SCOPE105,        I RETURN TO SCOPE LOOP TEST WORD
5752     NEXT,   GOTO-PAGE(7),          I BUT TABLE
5753     J/BUTIR15-12                    I BIT<15:12> TEST
(5220) DCS[0.00,0.0,0.0] BM[0101..00.01..00.10..001..111...0.0,0,0,0...0.0000...0.0000,0...11.100...011,000,000]

5754
5755
5756
5757 5221: 1(FREE)
5758 SCOPE105:
5759     PO,      BUSDIN_EMIT={I},        I RESET PROC UCON
5760     EN=CLK-IR[15-00],                I
5761     NEXT,   BUTD(SCOPE),             I NO ERROR: "SETUPCSP17A" (+1, WORDS)
5762     J/SETUPCSP17A                    I ERROR: "LOADSR105A" (-17, WORDS)
(5221) DCS[0.00,0.1,0.0] BM[0000..00.00..00.01..000..100...0.0,0,0,0...1.1001...0.0000,0...11.000...101,100,011]

5763
5764
5765
5766
```

```

5767
5768
5769
5770 1. PAGE=====
5771
5772 .T0C * TEST114-121: ALU LOGIC TESTS / DIC TESTS
5773
5774
5775
5776 TESTS: 114 - 121                                UWORDS: 250 + 300
5777
5778 FUNCTIONS:
5779
5780 THESE TESTS TEST THE ALU LOGIC FUNCTIONS.
5781
5782
5783
5784
5785
5786
5787
5788 SUMMARY OF ALU LOGIC / DIC TESTS:
5789
5790 TEST      ALU      OPERANDS      D(C)
5791 NUMR      FUNCTION  A/B=D      FUNCTION
5792 -----
5793 114A      ZERO      1/1=0      CIN=PS(C)=0
5794
5795 115A      NOT-A      1/1=0      CIN=1
5796 115B      NOT-A      0/1=1      PS(C)=0
5797 115C      NOT-A      0/0=1
5798 115D      NOT-A      1/0=0
5799
5800 116A      NOT-A-AND-B 0/1=1      ALU15=1, D(C)=1
5801 116B      ZERO      1/0=0
5802 116C      NOT-A-AND-B 0/0=0      ALU15=0, D(C)=0
5803 116D      ZERO      0/1=0
5804
5805 117A      A-AND-NOT-B 1/0=1      CIN=D(C)=0
5806 117B      A-AND-NOT-B 1/1=0      (ALU15=1)
5807 117C      A-AND-NOT-B 0/0=0      CIN=D(C)=1
5808
5809 120A      A-AND-B     0/1=0
5810 120B      A-AND-B     1/1=1      CIN=0
5811
5812 121A      A-XOR-B     0/0=0      ALU00=1
5813 121B      A-XOR-B     1/0=1      ALU00=0
5814 121C      A-XOR-B     0/1=1      ALU07=1
5815 121D      A-XOR-B     1/1=0      ALU07=0
5816
5817 122A      A-IOR-B     0/0=0      CIN=1
5818 122A3     BUT(D<14-00>=ZERO&D15) W/ D=(0000000)
5819 122A4     BUT(D<14-00>=ZERO&D15) W/ D=(125252)
5820

```

```

5821 1
5822
5823 1. FOR THE ALU LOGIC TESTS FOLLOWING, THE REQUIRED CONSTANTS
5824 1. IN THE C8P ARE:
5825 5543:
5826 SETUPCSP17A:
5827 P3, C8PD[15]_EMIT, EMIT/000077, 1. MASK FOR BITS<05:00>
5828 NEXT, GOTO-PAGE(7), 1. XFR
5829 J/SETUPCSP16A
(5543) DCS(0.00,0.0,0.0,0) BM(0000..10.00..00.00..111..111...0.0,0.0..0.0..0.0010...1..0000,0...11.100...000,000,010)
5830
5831 7002: 1(FREE)
5832 SETUPCSP16A:
5833 P3, C8PD[16]_EMIT, EMIT/170000, 1. BITS<15:12> SET
5834 NEXT, J/SETUPCSP15A
(7002) DCS(0.00,0.0,0.0,0) BM(1111..10.00..00.00..000..000...0.0,0.0..0.0..0.0001...1..0000,0...11.100...000,001,000)
5835
5836 7010: 1(FREE)
5837 SETUPCSP15A:
5838 P3, C8PD[17]_EMIT, EMIT/007700, 1. MASK FOR BITS<11:06>
5839 NEXT, J/SETUPCSP14A
(7010) DCS(0.00,0.0,0.0,0) BM(0000..10.11..11.11..000..000...0.0,0.0..0.0..0.0000...1..0000,0...11.100...000,001,001)
5840
5841 7011: 1(FREE)
5842 SETUPCSP14A:
5843 P3, C8PD[14]_EMIT, EMIT/000100, 1. BIT<06> SET
5844 NEXT, J/SETUPCSP12A
(7011) DCS(0.00,0.0,0.0,0) BM(0000..10.00..00.01..000..000...0.0,0.0..0.0..0.0011...1..0000,0...11.100...000,001,010)
5845
5846 7012: 1(FREE)
5847 SETUPCSP12A:
5848 P3, C8PD[11]_EMIT, EMIT/125252, 1. PATTERN: "1010 1010 1010 1010"
5849 NEXT, J/SETUPCSP05A
(7012) DCS(0.00,0.0,0.0,0) BM(1010..10.10..10.10..101..010...0.0,0.0..0.0..0.0110...1..0000,0...11.100...000,001,011)
5850
5851 7013: 1(FREE)
5852 SETUPCSP05A:
5853 P3, C8PD[10]_EMIT, EMIT/052525, 1. PATTERN: "0101 0101 0101 0101"
5854 NEXT, J/SETUPCSP07A
(7013) DCS(0.00,0.0,0.0,0) BM(0101..10.01..01.01..010..101...0.0,0.0..0.0..0.0111...1..0000,0...11.100...000,001,100)
5855
5856 7014: 1(FREE)
5857 SETUPCSP07A:
5858 P3, C8PD[12]_EMIT, EMIT/177777, 1. PATTERN: "1111 1111 1111 1111 "
5859 NEXT, J/SETUPCSP00A
(7014) DCS(0.00,0.0,0.0,0) BM(1111..10.11..11.11..111..111...0.0,0.0..0.0..0.0101...1..0000,0...11.100...000,001,101)
5860
5861 7015: 1(FREE)
5862 SETUPCSP00A:
5863 P3, C8PD[13]_EMIT, EMIT/000000, 1. PATTERN: "0000 0000 0000 0000"
5864 NEXT, GOTO-PAGE(0), 1. SAME AS (4)
5865 J/TEST114A

```

```

(7015) DCS[0.00.0.0.0.0] BM[0000..10.00..00.00..000.000..0.0.0.0..0.0100..1..0000.0..11.100..110.000.111]
5866
5867
5868
5869
5870
5871
5872 IVERIFY THAT WITH: ALU=(ZERO), A=(177777), B=(177777), THEN D=(000000), AND D[C]=CIN=PS[C]=(0)
5873 4607:
5874 TEST114A:
5875     PO,      LOAD=ENUA(ZTARGET434),      INSTRS FOR ID=(000000)
5876             LOAD=ERROR(TEST114A),      ERROR DIRECTORY KEY
5877             DCS=CTR(C6.),              COMPARE AT TARGET
5878     NEXT,    J/GETONES114A
(4607) DCS[1.00.1.0.0.0] BM[0111..00.11..11.00..011..100..0.0.0.0..0.0000..0..0000.0..11.000..110.000.000]
5879
5880 4600:
5881 GETONES114A:
5882     P2-T,    D_CSPD(C177777), D[C]=ALU15,      IALL ONES STORED HERE
5883     P3,      A=BSPHI(C177777)_D,              IWRITE INTO ASP,BSP
5884     NEXT,    J/ALU114A
(4600) DCS[0.00.0.0.0.0] BM[1010..10.11..00.01..101..100..0.1.0..0.0..0.0101..0..1111.0..11.000..000.000.011]
5885
5886 4003: I(FREE)
5887 ALU114A:
5888     P2-T,    D_ZERO, D[C]=CINMUX,              IALU=(ZERO), D[C]=CIN=PS[C]=(0)
5889             BUS=A_ASPHI(C177777),          IA=(177777)
5890             BUS=B_CSPD(C177777),          IB=(177777)
5891             NEXT, J/GETZEROS114A          ID=(000000)
(4003) DCS[0.00.0.0.0.0] BM[0011..10.00..11.01..101..000..0.1.0..0.0..0.0101..0..0000.0..11.000..000.000.100]
5892
5893 4004: I(FREE)
5894 GETZEROS114A:
5895     P3,      A=BSPHI(C000000)_D,              IALL ZEROS STORED HERE IN ASP, BSP
5896     NEXT,    J/GOBUT114A
(4004) DCS[0.00.0.0.0.0] BM[0000..00.11..00.01..100..000..0.0.0.0.0..0.0000..0..1111.0..11.000..000.000.101]
5897
5898 4005: I(FREE)
5899 GOBUT114A:
5900     SETUP,   RETURN/TEST114A2,              IEXEC SUBR WHICH:
5901             NEXT, CALL[DZERO]                I(1) D -> IR
5902             (4005) DCS[0.00.0.0.0.0] BM[0100..00.11..00.00..011..111..0.0.0.0.0..0.0000..0..0000.0..11.100..010.100.010]
5903
5904 I CHECK THAT D[C] GOT A (0) FROM "CINMUX=PS[C]", ABOVE
5905 4603:
5906 TEST114A2:
5907     PO,      LOAD=ENUA(ZTARGET402),          IBIT <00> = D[C] = (0)
5908             LOAD=ERROR(TEST114A2),          ERROR DIRECTORY KEY
5909             DCS=CTR(C3.),                  ICOMPARE AT TARGET
5910     NEXT,    J/GOBUT114A2
(4603) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..000..010..0.0.0.0.0..0.0000..0..0000.0..11.000..000.000.110]
5911

```

```

5912 4006: I(FREE)
5913 GOBUT114A2:
5914     SETUP,   RETURN/SCOPE114A,              IRETURN TO SCOPE LOOP TEST WORD
5915     NEXT,    GOTO=PAGE(7),                  IBUT TABLE
5916             J/BUTD[C]A                      ID(C)N IN BIT<00>
(4006) DCS[0.00.0.0.0.0] BM[0100..00.00..00.00..111..111..0.0.0.0.0..0.0000..0..0000.0..11.100..011.100.100]
5917
5918
5919 4007: I(FREE)
5920 SCOPE114A:
5921     P2-T,    D_CSPD(C052525), SAVE=D[C],        ISTORE A CONSTANT
5922     P3,      A=BSPHI(C052525)_D,              I
5923     NEXT,    BUTD[SCOPE],                    INO ERROR: "TEST115A1" (+1, WORDS)
5924             J/TEST115A1                      I ERROR: "GETONES114A" (-6, WORDS)
(4007) DCS[0.00.0.1.0.0] BM[1010..10.11..00.01..111..111..0.1.0..0.0..0.0111..0..1111.0..11.000..110.000.001]
5925
5926
5927
5928
5929
5930
5931
5932 I THIS NEXT SET OF 12, TESTS EXERCISES THE "NOT-A" ALU FUNCTION
5933
5934
5935
5936 I TESTS 115A 1-3 VERIFIES THAT WITH:
5937 IALU=(NOT-A), A=(052525), B=(177777), THEN D=(125252), AND D[C]=CINMUX=(1)
5938 4601:
5939 TEST115A1:
5940     PO,      LOAD=ENUA(ZTARGET412),          IBIT<15:12> = "1010"
5941             LOAD=ERROR(TEST115A1),          ERROR DIRECTORY KEY
5942             DCS=CTR(C7.),                  ICOMPARE AT TARGET
5943     NEXT,    J/ALU115A1
(4601) DCS[1.00.1.0.0.0] BM[1000..00.11..11.00..001..010..0.0.0.0.0..0.0000..0..0000.0..11.000..110.001.000]
5944
5945 4610:
5946 ALU115A1:
5947     PO,      BUMP=VERIFY,                    ICOUNT
5948     P2-T,    D_NOT=A, D[C]=CINMUX,          IALU=(NOT-A) D[C]=CINMUX=(1)
5949             BUS=A_ASPHI(C052525),          IA=(052525)
5950             BUS=B_CSPD(C177777),          IB=(177777)
5951     NEXT,    J/GETALTN115A1                 ID=(125252)
(4610) DCS[0.00.0.0.0.1] BM[0000..10.00..11.01..111..000..0.0.1.0..0.0..0.0101..0..0000.0..11.000..000.001.000]
5952
5953 4010: I(FREE)
5954 GETALTN115A1:
5955     P3,      A=BSPHI(C125252)_D,              ISTORE CONSTANT (125252), HOPEFULLY
5956     NEXT,    J/GOBUT115A1
(4010) DCS[0.00.0.0.0.0] BM[0000..00.11..00.01..110..000..0.0.0.0.0..0.0000..0..1111.0..11.000..000.001.001]
5957
5958 4011: I(FREE)
5959 GOBUT115A1:

```

```

5960          SETUP,      RETURN/TEST115A4,          IEXEC SUBR WHICH:
5961          4604:        I(1) D<15:12> -> IP<15:12>
5962          NEXT,      CALL[D15-12]                  I(9) BUT{IR15-12} INTO ZTARGET---
(4601) DCS[0.00.0.0.0.0] BMT[0100..00.11..00.00..100..111...0.0.0.0.0...0.0000...0..0000.0...11.100...010.011.100]

5963
5964          ICHECK THAT D[C] GOT A {1} FROM "CINMUX={1}", ABOVE
5965          4604:
5966          TEST115A4:
5967          PO,
5968          LOAD=ENUA(ZTARGET403),                     IBIT 400 = D[C] = {1}
5969          LOAD=ERROR(TEST115A4),                     IERROR DIRECTORY KEY
5970          DCS=CTR(C3.),                             ICOMPARE AT TARGET
5971          BUMP=VERIFY,                               ICOUNT
5972          JGOBUT115A4
(4604) DCS[1.00.1.0.0.1] BMT[1100..00.11..11.00..000..011...0.0.0.0.0...0.0.0000...0..0000.0...11.100...000.001.010]

5973
5974          4012: I(FREE)
5975          GOBUT115A4:
5976          SETUP,      RETURN/TEST115A2,             IRETURN TO START OF NEXT SUBTEST
5977          NEXT,      GOTO=PAGE(7),                 IBUT TABLE
5978          J/BUTD[C]A                                IDC{N IN BIT<00>
(4012) DCS[0.00.0.0.0.0] BMT[0100..00.11..00.01..110..111...0.0.0.0.0...0.0.0000...0..0000.0...11.100...011.100.100]

5979
5980          ICHECK BIT<11:06> = "1010 10"
5981          4616:
5982          TEST115A2:
5983          PO,
5984          LOAD=ENUA(ZTARGET412),                     IINSTRS=E70 OUTPUT FOR BIT <11:06>="101 010"
5985          LOAD=ERROR(TEST115A2),                     IERROR DIRECTORY KEY
5986          DCS=CTR(C6.),                             ICOMPARE AT TARGET
5987          BUMP=VERIFY,                               ICOUNT
5988          NEXT,      JGOBUT115A2
(4616) DCS[1.00.1.0.0.1] BMT[1001..00.11..11.00..001..010...0.0.0.0.0...0.0.0000...0..0000.0...11.100...000.001.011]

5989
5990          4013: I(FREE)
5991          GOBUT115A2:
5992          SETUP,      RETURN/TEST115A3,             IEXEC SUBR WHICH:
5993          NEXT,      CALL[D11-06]                  I(1) D<11:06> -> IP<11:06>
5994          (4013) DCS[0.00.0.0.0.0] BMT[0100..00.11..00.01..101..111...0.0.0.0.0...0.0.0000...0..0000.0...11.100...010.011.110]

5995
5996          ICHECK BIT<05:00> = "10 1010"
5997          4615:
5998          TEST115A3:
5999
6000          PO,
6001          LOAD=ENUA(ZTARGET425),                     IINSTRS=E88 OUTPUT FOR BIT <05:00> = "101 010"
6002          LOAD=ERROR(TEST115A3),                     IERROR DIRECTORY KEY
6003          DCS=CTR(C6.),                             ICOMPARE AT TARGET
6004          NEXT,      JGOBUT115A3
6005

```

```

0606      DCS[1.00,1.0,0,0] BM[1001..00.11..00.01..100..111...0,0,0,0...0,0000...0..0000,0...11,000...000,001,100]
0607      4014: 1(FREE)
0608      GORUT115A3;
0609          SETUP, RETURN/TEST115B1,                IEXEC SUBR WHICH:
0610                                I(1) D<05100> -> IR<05100>
0611      NEXT, CALL[D05=00]                            I(2) BUT(INSTRS) INTO ZTARGET---
(4014) DCS[0.00,0.0,0,0] BM[0100..00.11..00.01..100..111...0,0,0,0...0,0000...0..0000,0...11,100...010,100,000]

0612
0613
0614
0615
0616      | - - - - -
0617
0618      !TESTS 115B 1-3 VERIFIES THAT WITH:
0619      !ALU=(NOT-A), A=(125252), B=(177777), THEN D=(052525), AND D[C]_PS[C]=(0)
0620      4614:
0621      TEST115B1:
0622          PO, LOAD-ENUA(ZTARGET405),                !BIT <15:12> = "0101"
0623              LOAD-ERROR(TEST115B1),                !ERROR DIRECTORY KEY
0624              DCS-CTR(C6.),                          !COMPARE AT TARGET
0625              J/ALU115B1                             !
(4614) DCS[1.00,1.0,0,0] BM[1001..00.11..11.00..000..101...0,0,0,0...0,0000...0..0000,0...11,000...000,001,101]

0626      4015: 1(FREE)
0627      ALU115B1:
0628          PO, BUMP-VERIFY,                            !COUNT
0629              D_NOT=A, D[C]_PS[C],                    !ALU=(NOT-A), D[C]=PS[C]=(0)
0630              BUS-A_ASPHI(C125252),                  !A=(125252)
0631              BUS-B_CSPD(C177777),                   !B=(177777)
0632              NEXT, J/GORUT115B1                     !D=(052525)
0633      (4015) DCS[0.00,0.0,0,1] BM[0000..10.00..11.01..110..001...0,1,0,0...0,0101...0..0000,0...11,000...000,001,110]

0634      4016: 1(FREE)
0635      GORUT115B1:
0636          SETUP, RETURN/TEST115B4,                IEXEC SUBR WHICH:
0637                                I(1) D<15:12> -> IR<15:12>
0638                                I(2) BUT(IR15-12) INTO ZTARGET---
(4016) DCS[0.00,0.0,0,0] BM[0100..00.11..00.00..010..111...0,0,0,0...0,0000...0..0000,0...11,100...010,011,100]

0640
0641
0642      !CHECK THAT D[C] GOT A (0) FROM "PS[C]", ABOVE
0643      4602:
0644      TEST115B4:
0645          PO, LOAD-ENUA(ZTARGET402),                !BIT <00> = D[C] = (0)
0646              LOAD-ERROR(TEST115B4),                !ERROR DIRECTORY KEY
0647              DCS-CTR(C3.),                          !COMPARE AT TARGET
0648              BUMP-VERIFY,                            !COUNT
0649              J/GORUT115B4                             !
(4602) DCS[1.00,1.0,0,1] BM[1100..00.11..11.00..000..010...0,0,0,0...0,0000...0..0000,0...11,000...000,001,111]

0650      4017: 1(FREE)

```

```
6052 GORUT115B4:
6053     SETUP, RETURN/TEST115B2,      ;RETURN TO START OF NEXT SUBTEST
6054     NEXT,  GOTO-PAGE(7),          ;BUT TABLE
6055     J/BUTD(C)A                     ;IDCM IN BIT<00>
(4017) DCS[0.00.0.0.0.0] BM[0100.00.11.00.01.011.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011.100.100]
6056
6057
6058 ;CHECK BIT<11:06> = "0101 01"
6059 4613:
6060 TEST115B2:
6061     PO,      LOAD-ENVA(ZTARGET405), ;INSTRS-E78 OUTPUT FOR BIT <11:06> = "010 101"
6062     LOAD-ERROR(TEST115B2),          ;ERROR DIRECTORY KEY
6063     DCS-CTR(C6.),                   ;COMPARE AT TARGET
6064     BUMP-VERIFY,                    ;COUNT
6065     NEXT,    J/GOBUT115B2
(4613) DCS[1.00.1.0.0.1] BM[1001.00.11.11.00.000.101.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.000.010.000]
6066
6067 4020: I(FREE)
6068 GORUT115B2:
6069     SETUP, RETURN/TEST115B3,      ;EXEC SUBR WHICH:
6070     NEXT,    CALL[D11-06]          ;(1) D<11:06> -> IR<11:06>
6071 (4020) DCS[0.00.0.0.0.0] BM[0100.00.11.00.01.010.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.010.011.110]
6072
6073
6074 ;CHECK BIT<05:00> = "01 0101"
6075 4612:
6076 TEST115B3:
6077
6078     PO,      LOAD-ENVA(ZTARGET432), ;INSTRS-E88 OUTPUT FOR BIT<05:00> = "010 101"
6079     LOAD-ERROR(TEST115B3),          ;ERROR DIRECTORY KEY
6080     DCS-CTR(C6.),                   ;COMPARE AT TARGET
6081     NEXT,    J/GOBUT115B3
(4612) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.011.010.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.000.010.001]
6083
6084 4021: I(FREE)
6085 GORUT115B3:
6086     SETUP, RETURN/SCOPE115B,      ;EXEC SUBR WHICH:
6087     NEXT,    CALL[D05-00]          ;(1) D<05:00> -> IR<05:00>
6088 (4021) DCS[0.00.0.0.0.0] BM[0100.00.00.00.10.010.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.010.100.000]
6089
6090
6091 4022: I(FREE)
6092 SCOPE115B:
6093     NEXT,    BUTD[SCOPE],          ;NO ERROR: "TEST115C1" (+1, WORDS)
6094     J/TEST115C1                   ;ERROR: "ALU115A1" (-17, WORDS)
(4022) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.110.001.001]
6095
6096
```

```
6097
6098
6099 I - - - - -
6100
6101 ;TESTS 115C 1-3 VERIFIES THAT WITH:
6102 ;ALU<NOT-A>, A=(052525), B=(000000), THEN D=(125252)
6103 4611:
6104 TEST115C1:
6105     PO,      LOAD-ENVA(ZTARGET412), ;BIT <15:12> = "1010"
6106     LOAD-ERROR(TEST115C1),          ;ERROR DIRECTORY KEY
6107     DCS-CTR(C6.),                   ;COMPARE AT TARGET
6108     NEXT,    J/ALU115C1
(4611) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.001.010.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.110.010.010]
6109
6110 4622:
6111 ALU115C1:
6112     PO,      BUMP-VERIFY,           ;COUNT
6113     P2-T,    D_NOT-A, SAVE-D[C],    ;[ALU<NOT-A>]
6114     BUS-A-ASPHI(C052525),          ;[A=(052525)]
6115     BUS-B-CPDP(C000000),           ;[B=(000000)]
6116     NEXT,    J/GOBUT115C1          ;[D=(125252)]
(4622) DCS[0.00.0.0.0.1] BM[0000.10.00.11.01.111.111.0.0.0.0.0.0.0.0100.0.0.0.0.0.0.11.000.0.000.010.011]
6117
6118 4023: I(FREE)
6119 GORUT115C1:
6120     SETUP, RETURN/TEST115C2,      ;EXEC SUBR WHICH:
6121     NEXT,    CALL[D15-12]          ;(1) D<15:12> -> IR<15:12>
6122 (4023) DCS[0.00.0.0.0.0] BM[0100.00.11.00.01.111.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.010.011.100]
6123
6124 ;CHECK BIT<11:06> = "1010 10"
6125 4617:
6126 TEST115C2:
6127     PO,      LOAD-ENVA(ZTARGET412), ;INSTRS-E78 OUTPUT FOR BIT <11:06> = "101 010"
6128     LOAD-ERROR(TEST115C2),          ;ERROR DIRECTORY KEY
6129     DCS-CTR(C6.),                   ;COMPARE AT TARGET
6130     BUMP-VERIFY,                    ;COUNT
6131     NEXT,    J/GOBUT115C2
(4617) DCS[1.00.1.0.0.1] BM[1001.00.11.11.00.001.010.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.000.0.000.010.100]
6132
6133 4024: I(FREE)
6134 GORUT115C2:
6135     SETUP, RETURN/TEST115C3,      ;EXEC SUBR WHICH:
6136     NEXT,    CALL[D11-06]          ;(1) D<11:06> -> IR<11:06>
6137 (4024) DCS[0.00.0.0.0.0] BM[0100.00.11.00.10.111.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.010.011.110]
6138
6139
6140
6141 ;CHECK BIT<05:00> = "10 1010"
6142 4627:
6143 TEST115C3:
```

```
6144
6145      PO,      LOAD-ENUA(ZTARGET425),      INSTRS-E88 OUTPUT FOR BIT<05:00> = "101 010"
6146      LOAD-ERROR(TEST115C3),      ERROR DIRECTORY KEY
6147      DCS-CTR(C6.),      COMPARE AT TARGET
6148      NEXT,      J/GOBUT115C3
(4627) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.010.101.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.000.010.101]
6149
6150      4025: 1(FREE)
6151      GOBUT115C3:
6152      SETUP,      RETURN/TEST115D1,      EXEC SUBR WHICH:
6153      NEXT,      CALL[D05-00]      (1) D<05:00> -> IR<05:00>
6154      (4075) DCS[0.00.0.0.0.0] BM[0100.00.11.00.10.110.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.100.000]      (2) BUT(INSTR5) INTO ZTARGET---
6155
6156
6157
6158
6159      I - - - - -
6160
6161      ITESTS 115D 1-3 VERIFIES THAT WITH:
6162      IALU=(NOT-A, A=(125252), B=(000000), THEN D=(052525)
6163      4626:
6164      TEST115D1:
6165      PO,      LOAD-ENUA(ZTARGET405),      BIT<15:12> = "0101"
6166      LOAD-ERROR(TEST115D1),      ERROR DIRECTORY KEY
6167      DCS-CTR(C6.),      COMPARE AT TARGET
6168      NEXT,      J/ALU115D1
(4626) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.000.101.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.000.010.110]
6169
6170      4026: 1(FREE)
6171      ALU115D1:
6172      PO,      BUMP-VERIFY,      COUNT
6173      P2-T,      D_NOT-A, SAVE-D[C],      IALU=(NOT-A), D[C]_0
6174      BUS-A-ASPHI(C125252),      IAm(125252)
6175      BUS-B-CSPD(C000000),      IB=(000000)
6176      NEXT,      J/GOBUT115D1      ID=(052525)
(4026) DCS[0.00.0.0.0.1] BM[0000.10.00.11.01.110.111.0.0.1.0.0.0.0.0100.0.0.0000.0.11.000.0.000.010.111]
6177
6178      4027: 1(FREE)
6179      GOBUT115D1:
6180      SETUP,      RETURN/TEST115D2,      EXEC SUBR WHICH:
6181      NEXT,      CALL[D15-12]      (1) D<15:12> -> IR<15:12>
6182      (4027) DCS[0.00.0.0.0.0] BM[0100.00.11.00.10.101.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.011.100]      (2) BUT(IR15-12) INTO ZTARGET---
6183
6184      ICHECK BIT<11:06> = "0101 01"
6185      4625:
6186      TEST115D2:
6187      PO,      LOAD-ENUA(ZTARGET405),      INSTRS-E78 OUTPUT FOR BIT<11:06> = "010 101"
6188      LOAD-ERROR(TEST115D2),      ERROR DIRECTORY KEY
6189      DCS-CTR(C6.),      COMPARE AT TARGET
6190      BUMP-VERIFY,      COUNT
```

```
6191      NEXT,      J/GOBUT115D2
(4625) DCS[1.00.1.0.0.1] BM[1001.00.11.11.00.000.101.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.000.011.000]
6192
6193      4030: 1(FREE)
6194      GOBUT115D3:
6195      SETUP,      RETURN/TEST115D3,      EXEC SUBR WHICH:
6196      NEXT,      CALL[D11-06]      (1) D<11:06> -> IR<11:06>
6197      (4030) DCS[0.00.0.0.0.0] BM[0100.00.11.00.10.100.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.011.110]      (2) BUT(INSTR5) INTO ZTARGET---
6198
6199
6200
6201      ICHECK BIT<05:00> = "01 0101"
6202      4624:
6203      TEST115D3:
6204
6205      PO,      LOAD-ENUA(ZTARGET432),      INSTRS-E88 OUTPUT FOR BIT<05:00> = "010 101"
6206      LOAD-ERROR(TEST115D3),      ERROR DIRECTORY KEY
6207      DCS-CTR(C6.),      COMPARE AT TARGET
6208      NEXT,      J/GOBUT115D3
(4624) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.011.010.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.000.011.001]
6209
6210      4031: 1(FREE)
6211      GOBUT115D3:
6212      SETUP,      RETURN/SCOPE115D,      EXEC SUBR WHICH:
6213      NEXT,      CALL[D05-00]      (1) D<05:00> -> IR<05:00>
6214      (4031) DCS[0.00.0.0.0.0] BM[0100.00.00.00.11.010.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.100.000]      (2) BUT(INSTR5) INTO ZTARGET---
6215
6216      4032: 1(FREE)
6217      SCOPE115D:
6218      PO,      BUMP-VERIFY,      COUNT
6219      NEXT,      BUT[SCOPE],      IALU=(NOT-A-AND-B), A=(000000), B=(125252), THEN D=(125252)
6220      J/TEST116A1      I ERROR: "TEST116A1" (+1, WORDS)
(4032) DCS[0.00.0.1.0.1] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.110.010.011]
6221
6222
6223
6224
6225      I - - - - -
6226
6227      ITHIS NEXT SET OF 12 TESTS EXERCISES THE ALU FUNCTIONS
6228      I"ZERO" AND "NOT-A-AND-B", AND THE D(C) INPUTS "ALU15" AND "D(C)"
6229
6230      I - - - - -
6231
6232      ITESTS 116A 1-5 VERIFIES THAT WITH:
6233      IALU=(NOT-A-AND-B), A=(000000), B=(125252), THEN D=(125252)
6234      4623:
6235      TEST116A1:
6236      PO,      LOAD-ENUA(ZTARGET412),      BIT<15:12> = "1010"
6237      LOAD-ERROR(TEST116A1),      ERROR DIRECTORY KEY
```

```

6238      DCS=CTR(C6.),          ICOMPARE AT TARGET
6239      NEXT, J/ALU116A1
(4623) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.001.010...0.0.0.0.0...0.0000...0.0000.0...11.000...110.011.100]
6240
6241      4634: 1(FREE)
6242      ALU116A1:
6243      P0,      BUMP-VERIFY,          ICOUNT
6244      P2-T,     D_NOT=A-AND=B, D[C]_ALU15,      !ALU=(NOT-A-AND-B), D[C]=(1)
6245      BUS-A_ASPH(C000000),          !A=(000000)
6246      BUS-B_CSPD(C123252),          !B=(123252)
6247      NEXT, J/GOBUT116A1          !D=(123252)
(4634) DCS[0.00.0.0.0.1] BM[0010.10.00.11.01.100.100...0.1.0.0.0...0.0110...0.0000.0...11.000...000.011.011]
6248
6249      4033: 1(FREE)
6250      GOBUT116A1:
6251      SETUP,   RETURN/TEST116A2,      IEXEC SUBR WHICH:
6252      NEXT,    CALL[D15=12]           ! (1) D<15112> => IR<15112>
6253      (4033) DCS[0.00.0.0.0.0] BM[0100.00.11.00.10.001.111...0.0.0.0.0...0.0000...0.0000.0...11.100...010.011.100]
        ! (2) BUT (IR15=12) INTO ZTARGET---
6254      !CHECK THAT D[C] GOT A (1) FROM "ALU15," ABOVE
6255      4621:
6256      TEST116A2:
6257      P0,      LOAD=ENUA(ZTARGET403),      !BIT <00> = D[C] = (1)
6258      LOAD=ERROR(TEST116A2),          !ERROR DIRECTORY KEY
6259      DCS=CTR(C3.),                  !COMPARE AT TARGET
6260      BUMP-VERIFY,                  !ICOUNT
6261      NEXT, J/GOBUT116A2
(4621) DCS[1.00.1.0.0.1] BM[1100.00.11.11.00.000.011...0.0.0.0.0...0.0000...0.0000.0...11.000...000.011.100]
6262      4034: 1(FREE)
6263      GOBUT116A2:
6264      SETUP,   RETURN/TEST116A3,      !RETURN TO START OF NEXT SUBTEST
6265      NEXT,    GOTO=PAGE(7),          !BUT TABLE
6266      J/BUTD[C]A                     !D[C] IN BIT <00>
6267      (4034) DCS[0.00.0.0.0.0] BM[0100.00.11.00.10.000.011...0.0.0.0.0...0.0000...0.0000.0...11.100...011.100.100]
6268
6269      !CHECK BIT<11106> = "1010 10"
6270      4620:
6271      TEST116A3:
6272      P0,      LOAD=ENUA(ZTARGET412),      !INSTR5-E78 OUTPUT FOR BIT <11106> = "101 010"
6273      LOAD=ERROR(TEST116A3),          !ERROR DIRECTORY KEY
6274      DCS=CTR(C6.),                  !COMPARE AT TARGET
6275      BUMP-VERIFY,                  !ICOUNT
6276      NEXT, J/GOBUT116A3
(4620) DCS[1.00.1.0.0.1] BM[1001.00.11.11.00.001.010...0.0.0.0.0...0.0000...0.0000.0...11.000...000.011.101]
6277
6278      4035: 1(FREE)
6279      GOBUT116A3:
6280      SETUP,   RETURN/TEST116A4,      IEXEC SUBR WHICH:
6281      NEXT,    CALL[D11=06]           ! (1) D<11106> => IR<11106>
6282      ! (2) BUT (INSTR5) INTO ZTARGET---

```

```

(4035) DCS[0.00.0.0.0.0] BM[0100.00.11.00.11.111.111...0.0.0.0.0...0.0000...0.0000.0...11.100...010.011.110]
6283
6284
6285      !CHECK BIT<05100> = "10 1010"
6286      4637:
6287      TEST116A4:
6288
6289      P0,      LOAD=ENUA(ZTARGET425),      !INSTR5-E88 OUTPUT FOR BIT <05100> = "101 010"
6290      LOAD=ERROR(TEST116A4),          !ERROR DIRECTORY KEY
6291      DCS=CTR(C6.),                  !COMPARE AT TARGET
6292      NEXT, J/GOBUT116A4
(4637) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.010.101...0.0.0.0.0...0.0000...0.0000.0...11.000...000.011.110]
6293
6294      4036: 1(FREE)
6295      GOBUT116A4:
6296      SETUP,   RETURN/TEST116A5,      IEXEC SUBR WHICH:
6297      NEXT,    CALL[D05=00]           ! (1) D<05100> => IR<05100>
6298      (4036) DCS[0.00.0.0.0.0] BM[0100.00.11.00.11.110.111...0.0.0.0.0...0.0000...0.0000.0...11.100...010.100.000]
        ! (2) BUT (INSTR5) INTO ZTARGET---
6299
6300
6301
6302      !CHECK THAT D[C] WAS PROPOGATED UNCHANGED AS A (1), VIA D[C]_D[C]
6303      4636:
6304      TEST116A5:
6305      P0,      LOAD=ENUA(ZTARGET403),      !BIT<01> = D[C] = (1)
6306      LOAD=ERROR(TEST116A5),          !ERROR DIRECTORY KEY
6307      DCS=CTR(C3.),                  !COMPARE AT TARGET
6308      NEXT, J/GOBUT116A5
(4636) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.000.011...0.0.0.0.0...0.0000...0.0000.0...11.000...000.011.111]
6309
6310      4037: 1(FREE)
6311      GOBUT116A5:
6312      SETUP,   RETURN/TEST116B,      !RETURN TO START OF NEXT SUBTEST
6313      NEXT,    GOTO=PAGE(7),          !BUT TABLE
6314      J/BUTD[C]B                     !D[C] IN BIT <01>
6315      (4037) DCS[0.00.0.0.0.0] BM[0100.00.11.00.00.110.111...0.0.0.0.0...0.0000...0.0000.0...11.100...011.101.000]
6316
6317
6318
6319
6320
6321      I = - - - - -
6322
6323      !TEST 116B VERIFIES THAT WITH:
6324      !ALU=(ZEP0), A=(123252), B=(052525), THEN D=(000000)
6325      4606:
6326      TEST116B:
6327      P0,      LOAD=ENUA(ZTARGET434),      !INSTR5 FOR IR=(000000)
6328      LOAD=ERROR(TEST116B),          !ERROR DIRECTORY KEY
6329      DCS=CTR(C6.),                  !COMPARE AT TARGET

```

```

6330      NEXT,      J/ALU116B
(4606) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.011.100.0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.100.000]
6331
6332      4040: 1(FREE)
6333      ALU116B:
6334      P2-T,      D_ZERO, SAVE-D[C],      ;ALU=(ZERO), D[C]=D[C]-(1)
6335      BUS-A_ASPHI(C125252),      ;A=(125252)
6336      BUS-B_CSPD(C052525),      ;B=(052525)
6337      NEXT,      J/GOBUT116B      ;D=(000000)
(4040) DCS[0.00.0.0.0.0] BM[0011.10.00.11.01.110.111.0.0.1.0.0.0.0.0111...0.0000.0...11.000...000.100.001]
6338
6339      4041: 1(FREE)
6340      GOBUT116B:
6341      SETUP,      RETURN/SCOPE116C,      ;EXEC SUBR WHICH:
6342      NEXT,      CALL[DZERO]      ;(1) D=> IR
6343      (4041) DCS[0.00.0.0.0.0] BM[0100.00.00.01.00.010.111.0.0.0.0.0.0.0.0000...0.0000.0...11.000...010.100.010]
6344
6345      4042: 1(FREE)
6346      SCOPE116C:
6347      NEXT,      BUTD[SCOPE],      ;NO ERROR: "TEST116C" (+1, WORDS)
6348      J/TEST116C      ; ERROR: "ALU116A1" (-13, WORDS)
6349      (4042) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000...0.0000.0...11.000...110.011.101]
6350
6351      I - - - - -
6352
6353      1TEST 116C 1-5 VERIFIES THAT WITH:
6354      ;ALU=(NOT-A-AND-B), A=(000000), B=(052525), THEN D=(052525)
6355      4635:
6356      TEST116C1:
6357      PO,      LOAD-ENUA(ETARGET405),      ;BIT<15:12> = "0101"
6358      LOAD-ERROR(TEST116C1),      ;ERROR DIRECTORY KEY
6359      DCS-CTRC(6.),      ;COMPARE AT TARGET
6360      NEXT,      J/ALU116C1      ;
6361      (4635) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.000.101.0.0.0.0.0.0.0.0000...0.0000.0...11.000...110.100.110]
6362
6363      4646:
6364      ALU116C1:
6365      PO,      BUMP-VERIFY,      ;COUNT
6366      P2-T,      D_NOT-A-AND-B, D[C]_ALU15,      ;ALU=(NOT-A-AND-B), D[C]=(0)
6367      BUS-A_ASPHI(C000000),      ;A=(000000)
6368      BUS-B_CSPD(C052525),      ;B=(052525)
6369      NEXT,      J/GOBUT116C1      ;D=(052525)
6370      (4646) DCS[0.00.0.0.0.1] BM[0010.10.00.11.01.100.100.0.0.1.0.0.0.0.0111...0.0000.0...11.000...000.100.011]
6371
6372      4043: 1(PREF)
6373      GOBUT116C1:
6374      SETUP,      RETURN/TEST116C2,      ;EXEC SUBR WHICH:
6375      NEXT,      CALL[D<15:12> -> IR<15:12>]
6376

```

```

6377      NEXT,      CALL[D15-12]      I(2) BUT(IR15-12) INTO ZTARGET---
(4043) DCS[0.00,0.0,0.0] BM[0100..00.11..00.11..011..111...0.0,0.0..0.0...0.0000...0..0000.0...11.100...010,011,100]
6378
6379      !CHECK THAT D[C] GOT A (0) FROM "ALU15", ABOVE
6380      4631:
6381      TEST116C2:
6382      PO,          LOAD-ENUA(ZTARGET402),      !BIT <00> = D[C] = (0)
6383      LOAD-ERROR(TEST116C2),      !ERROR DIRECTORY KEY
6384      DCS-CTR(C3,),      !COMPARE AT TARGET
6385      RUMP-VERIFY,      !COUNT
6386      NEXT,      J/GOBUT116C2      !
(4633) DCS[1.00,1.0,0,1] BM[1100..00.11..11.00..000..010...0.0,0.0..0.0...0.0000...0..0000.0...11.100...000,100,100]
6387
6388      4044: ! (FREE)
6389      GOBUT116C2:
6390      SETUP,      RETURN/TEST116C3,      !RETURN TO START OF NEXT SURTEST
6391      NEXT,      GOTO-PAGE(7),      !BUT TABLE
6392      J/BUTD[C]A      !DCIM IN BIT<00>
(4044) DCS[0.00,0.0,0,0] BM[0100..00.11..00.11..010..111...0.0,0.0..0.0...0.0000...0..0000.0...11.100...011,100,100]
6393
6394      !CHECK BIT<11:06> = "0101 01"
6395      4632:
6396      TEST116C3:
6397      PO,          LOAD-ENUA(ZTARGET405),      !INSTR5-E78 OUTPUT FOR BIT<11:06>="010 101"
6398      LOAD-ERROR(TEST116C3),      !ERROR DIRECTORY KEY
6399      DCS-CTR(C6,),      !COMPARE AT TARGET
6400      RUMP-VERIFY,      !COUNT
6401      NEXT,      J/GOBUT116C3      !
(4632) DCS[1.00,1.0,0,1] BM[1001..00.11..11.00..000..101...0.0,0.0..0.0...0.0000...0..0000.0...11.100...000,100,101]
6402
6403      4045: ! (FREE)
6404      GOBUT116C3:
6405      SETUP,      RETURN/TEST116C4,      !EXEC SUBR WHICH:
6406      NEXT,      CALL[D11-06]      ! (1) D<11:06>-> IR<11:06>
6407      (4045) DCS[0.00,0.0,0,0] BM[0100..00.11..00.11..001..111...0.0,0.0..0.0...0.0000...0..0000.0...11.100...010,011,110]
6408
6409
6410
6411      !CHECK BIT <05:00> = "01 0101"
6412      4631:
6413      TEST116C4:
6414
6415      PO,          LOAD-ENUA(ZTARGET432),      !INSTR5-E88 OUTPUT FOR BIT <05:00> = "010 101"
6416      LOAD-ERROR(TEST116C4),      !ERROR DIRECTORY KEY
6417      NEXT,      J/GOBUT116C4      !
(4631) DCS[1.00,0.0,0,0] BM[0000..00.11..11.00..011..010...0.0,0.0..0.0...0.0000...0..0000.0...11.100...000,100,110]
6418
6419      4046: ! (FREE)
6420      GOBUT116C4:
6421      SETUP,      RETURN/TEST116C5,      !EXEC SUBR WHICH:

```

```

6422      NEXT, CALL[D05=00]      I(1) D<05100> => IR<05100>
6423      (4046) DCS[0.00,0.0,0,0,0] BM[0100,00,11,00,11,000,011,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,010,100,000]
6424
6425
6426
6427      I(CHECK THAT D[C] WAS PROPOGATED UNCHANGED AS A (0), VIA D[C]_D[C])
6428      4630: I(FREE)
6429      TEST116CS:
6430      PO,      LOAD=ENUA(ZTARGET401),      I(BIT<01> = D[C] = (0)
6431      DCS=CTR(C3,),      LOAD=ERROR(TEST116CS),      I(ERROR DIRECTORY KEY
6432      NEXT,      J/GOBUT116CS      I(COMPARE AT TARGET
6433      (4630) DCS[1.00,1.0,0,0,0] BM[1100,00,11,00,11,00,000,001,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,000,100,111]
6434
6435      4047: I(FREE)
6436      GOBUT116CS:
6437
6438      SETUP,      RETURN/TEST116D,      I(RETURN TO START OF NEXT SUBTEST
6439      NEXT,      GOTO=PAGE(7),      I(BUT TABLE
6440      J/BUTD[C]S      I(DIC)N IN BIT <01>
6441      (4047) DCS[0.00,0.0,0,0,0] BM[0100,00,11,01,00,101,0,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,011,101,000]
6442
6443
6444
6445
6446
6447      I - - - - -
6448
6449      I( TEST 116D VERIFIES THAT WITH:
6450      I( ALU=(ZERO), A=(052525), B=(125252), THEN D=(000000)
6451      4645:
6452      TEST116D:
6453      PO,      LOAD=ENUA(ZTARGET434),      I(INSTRS FOR IR=(0000000)
6454      DCS=CTR(C6,),      LOAD=ERROR(TEST116D),      I(ERROR DIRECTORY KEY
6455      NEXT,      J/ALU116D      I(COMPARE AT TARGET
6456      (4645) DCS[1.00,1.0,0,0,0] BM[1001,00,11,11,00,011,0,100,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,000,101,000]
6457
6458      4050: I(FREE)
6459      ALU116D:
6460      P2-T,      D_ZERO, SAVE=D[C],      I( ALU=(ZERO), D[C]=D[C]=(0)
6461      BUS=A_ASPHI(C052525),      I(A=(052525)
6462      BUS=B_CSPD(C125252),      I(B=(125252)
6463      NEXT,      J/GOBUT116D      I(D=(000000)
6464      (4050) DCS[0.00,0.0,0,0,0] BM[0011,10,00,11,01,111,0,111,0,0,0,0,0,0,0,0110,0,0,0000,0,0,11,100,0,000,101,001]
6465
6466      4051: I(FREE)
6467      GOBUT116D:
6468      SETUP,      RETURN/SCOPE116D,      I( EXEC SUBR WHICH:
6469      I(1) D => IR

```

```

6469      NEXT, CALL(DZERO)                                (2) BUT(INSTRS) INTO ZTARGET---
(4051) DCS[0.00.0.0.0.0] BM[0100.00.00.01.01.010.111.0.0.0.0.0.0.0.0000.0.0.0.0000.0.11.100.0.010.100.010]
6470
6471
6472
6473      4052: 1(FREF)
6474      SCNPF116D;
6475      NEXT,      BUTD[SCOPE],                                INO ERROR: "TEST117A1" (+1, WORDS)
6476      J/TEST117A1      J/ERROR: "ALU116C1" (-13, WORDS)
(4052) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.0.0.0.0.0.0.0.0.0000.0.0.0.0000.0.11.000.110.100.111]
6477
6478
6479
6480
6481
6482      | - - - - -
6483
6484      |THIS NEXT SET OF 9. TESTS EXERCIZES THE ALU FUNCTION
6485      |"A-AND-NOT-B", AND THE CARRYOUT OF "CINMUX=D[C]" INTO D[C]
6486
6487
6488
6489
6490      | - - - - -
6491
6492      |TESTS 117A 1-4 VERIFIES THAT WITH:
6493      |ALU=(A-AND-NOT-B), A={177777}, B={125252}, THEN D={052525}
6494      4647:
6495      TEST117A1:
6496      PO,      LOAD-ENVA(ZTARGET405),                                |BIT<15;12> = "0101"
6497      LOAD-ERROR(TEST117A1),                                |ERROR DIRECTORY KEY
6498      DCS-CTR(C6.),                                |COMPARE AT TARGET
6499      NEXT, J/ALU117A1      |
(4647) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.000.101.0.0.0.0.0.0.0.0000.0.0.0.0000.0.11.000.110.110.110]
6500
6501      4666:
6502      ALU117A1:
6503      PO,      BUMP-VERIPY,                                |COUNT
6504      P2-T, D_A-AND-NOT-B, D[C]_CINMUX,                                |ALU=(A-AND-NOT-B), D[C]=CIN=D[C]={0}
6505      BUS-A_ASPI(C177777),                                |A={177777}
6506      BUS-B_CSPD(C125252),                                |B={125252}
6507      NEXT, J/GOBUT117A1      |D={052525}
(4666) DCS[0.00.0.0.0.1] BM[0111.10.00.11.01.101.000.0.1.0.0.0.0.0.0110.0.0.0.0000.0.11.000.0.000.101.011]
6508
6509      4053: 1(FREF)
6510      GOBUT117A1:
6511
6512      SETUP. RETURN/TEST117A2,                                |EXEC SUBR WHICH:
6513      NEXT, CALL(D15-12)                                |1) D<15;12> -> IR<15;12>
(4053) DCS[0.00.0.0.0.0] BM[0100.00.11.01.00.100.111.0.0.0.0.0.0.0.0000.0.0.0.0000.0.11.100.0.010.011.100]
6514
6515      |CHECK BIT<11;06> = "0101 01"

```

```
6516 4644:
6517 TEST117A2:
6518 PO, LOAD-ENUA(ZTARGET405), INSTR5-E78 OUTPUT FOR BIT<11:06>="010 101"
6519 LOAD-ERROR(TEST117A2), ERROR DIRECTORY KEY
6520 DCS-CTR(C6.), ICOMPARE AT TARGET
6521 BUMP-VERIFY, ICOUNT
6522 J/GOBUT117A2
(4644) DCS[1,00,1,0,0,0] BM[1001,00,11,11,00,000,010,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,000,101,100]
6523
6524 4054: I(FREE)
6525 GOBUT117A2:
6526 SETUP, RETURN/TEST117A3, IEXEC SUBR WHICH:
6527 NEXT, CALL[D11=06] I(1) D<11:06> -> IR<11:06>
6528 (4054) DCS[0,00,0,0,0,0] BM[0100,00,11,01,00,011,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,010,011,110]
6529
6530
6531 ICHECK BIT<05:00> = "01 0101"
6532 4643:
6533 TEST117A3:
6534
6535 PO, LOAD-ENUA(ZTARGET432), INSTR5-E88 OUTPUT FOR BIT <05:00>="010 101"
6536 LOAD-ERROR(TEST117A3), ERROR DIRECTORY KEY
6537 DCS-CTR(C6.), ICOMPARE AT TARGET
6538 BUMP-VERIFY, ICOUNT
6539 J/GOBUT117A3
(4643) DCS[1,00,1,0,0,0] BM[1001,00,11,11,00,011,010,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,000,101,101]
6540
6541 4055: I(FREE)
6542 GOBUT117A3:
6543 SETUP, RETURN/TEST117A4, IEXEC SUBR WHICH:
6544 NEXT, CALL[D05=00] I(1) D<05:00> -> IR<05:00>
6545 (4055) DCS[0,00,0,0,0,0] BM[0100,00,11,01,00,010,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,010,100,000]
6546
6547 ICHECK THAT D(C) GOT A (0) FROM CINMUX, ABOVE
6548 4642:
6549 TEST117A4:
6550 PO, LOAD-ENUA(ZTARGET413), IBIT<02> = D(C) = (0)
6551 LOAD-ERROR(TEST117A4), ERROR DIRECTORY KEY
6552 DCS-CTR(C3.), ICOMPARE AT TARGET
6553 BUMP-VERIFY, ICOUNT
6554 J/GOBUT117A4
(4642) DCS[1,00,1,0,0,0] BM[1100,00,11,11,00,001,011,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,000,101,110]
6555
6556 4056: I(FREE)
6557 GOBUT117A4:
6558 SETUP, RETURN/TEST117B1, IRETURN TO START OF NEXT SUBTEST
6559 NEXT, GOTO-PAGE(7), IOUT TABLE
6560 J/BUTD(C)C I(D(C)) IN BIT<02>
(4056) DCS[0,00,0,0,0,0] BM[0100,00,11,01,00,001,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,011,001,000]
```

```
6561
6562
6563
6564
6565 I - - - - -
6566
6567 !TESTS 117B 1-3 VERIFIES THAT WITH:
6568 !ALU=(A-AND-NOT-B), A=(177777), B=(052525), THEN D=(125252)
6569 4641:
6570 TEST117B1:
6571 PO, LOAD-ENUA(ZTARGET412), IBIT<15:12> = "1010"
6572 LOAD-ERROR(TEST117B1), ERROR DIRECTORY KEY
6573 DCS-CTR(C6.), ICOMPARE AT TARGET
6574 NEXT, J/ALU117B1
(4641) DCS[1,00,1,0,0,0] BM[1001,00,11,11,00,001,010,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,000,101,111]
6575
6576 4057: I(FREE)
6577 ALU117B1:
6578 PO, BUMP-VERIFY, ICOUNT
6579 P2-T, N_A-AND-NOT-B, D(C)_ALU15, !ALU=(A-AND-NOT-B), D(C)=ALU15=(1)
6580 BUS-A_ASPhi(C177777), !A=(177777)
6581 BUS-B_CSPD(C052525), !B=(052525)
6582 NEXT, J/GOBUT117B1 !D=(125252)
(4057) DCS[0,00,0,0,0,0] BM[0111,10,00,11,01,101,100,0,0,1,0,0,0,0,0111,0,0,0000,0,0,11,000,0,000,110,000]
6583
6584 4060: I(FREE)
6585 GOBUT117B1:
6586 SETUP, RETURN/TEST117B2, IEXEC SUBR WHICH:
6587 NEXT, CALL[D15=12] I(1) D<15:12> -> IR<15:12>
6588 (4060) DCS[0,00,0,0,0,0] BM[0100,00,11,01,00,000,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,010,011,100]
6589
6590 ICHECK BIT <11:06> = "1010 10"
6591 4640:
6592 TEST117B2:
6593 PO, LOAD-ENUA(ZTARGET412), INSTR5-E78 OUTPUT FOR BIT <11:06>="101 010"
6594 LOAD-ERROR(TEST117B2), ERROR DIRECTORY KEY
6595 DCS-CTR(C6.), ICOMPARE AT TARGET
6596 BUMP-VERIFY, ICOUNT
6597 J/GOBUT117B2
(4640) DCS[1,00,1,0,0,0] BM[1001,00,11,11,00,001,010,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,000,110,001]
6598
6599 4061: I(FREE)
6600 GOBUT117B2:
6601 SETUP, RETURN/TEST117B3, IEXEC SUBR WHICH:
6602 NEXT, CALL[D11=06] I(1) D<11:06> -> IR<11:06>
6603 (4061) DCS[0,00,0,0,0,0] BM[0100,00,11,01,01,111,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,010,011,110]
6604
6605
6606
6607 ICHECK BIT<05:00> = "10 1010"
```

```

6608 4657:
6609 TEST117B3:
6610
6611 PO, LOAD-ENUA(ZTARGET425), INSTRS=200 OUTPUT FOR BIT <05:00> = "101 010"
6612 LOAD-ERROR(TEST117B3), ERROR DIRECTORY KEY
6613 DCS=CTR(C6.), COMPARE AT TARGET
6614 NEXT, J/GOBUT117B3
(4657) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.010.101.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.000.110.010]
6615
6616 4062: 1(FREE)
6617 GOBUT117B3:
6618 SETUP, RETURN/TEST117C1, EXEC SUBR WHICH:
6619 NEXT, CALL[D05=00] (1) D<05:00> -> IR<05:00>
6620 (4062) DCS[0.00.0.0.0.0] BM[0100.00.11.01.01.110.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.100.000] (2) BUT (INSTRS) INTO ZTARGET---
6621
6622
6623
6624
6625
6626
6627
6628 | - - - - -
6629
6630 1TFSTS 117C 1-2 VERIFIES THAT WITH:
6631 |ALU=(A-AND-NOT-B), A=(000000), B=(000000), THEN D=(000000)
6632 4656:
6633 TEST117C1:
6634 PO, LOAD-ENUA(ZTARGET434), INSTRS FOR IR=(000000)
6635 LOAD-ERROR(TEST117C1), ERROR DIRECTORY KEY
6636 DCS=CTR(C6.), COMPARE AT TARGET
6637 NEXT, J/ALUI17C1
(4656) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.011.100.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.000.110.011]
6638
6639 4063: 1(FREE)
6640 ALUI17C1:
6641 P2-T, D_A=AND-NOT-B, D[C]_CINMUX, |ALU=(A-AND-NOT-B), D[C]=CINMUX=D[C]=(1)
6642 BUS=A_ASPHI(C000000), |A=(000000)
6643 BUS=B_CSPD(C000000), |B=(000000)
6644 NEXT, J/GOBUT117C1
(4063) DCS[0.00.0.0.0.0] BM[0111.10.00.11.01.100.000.0.0.0.0.0.0.0.0100.0.0.0000.0.11.000.0.000.110.100]
6645
6646 4064: 1(FREE)
6647 GOBUT117C1:
6648 SETUP, RETURN/TEST117C2, EXEC SUBR WHICH:
6649 NEXT, CALL[DZERO] (1) D -> IR
6650 (4064) DCS[0.00.0.0.0.0] BM[0100.00.11.01.11.111.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.010.100.010] (2) BUT (INSTRS) INTO ZTARGET---
6651
6652
6653 |CHECK THAT D[C] GOT A (1) FROM CINMUX, ABOVE
6654 4677:

```

```

6655 TEST117C2:
6656 PO, LOAD-ENUA(ZTARGET417), |BIT<03> = D[C] = (1)
6657 LOAD-ERROR(TEST117C2), ERROR DIRECTORY KEY
6658 DCS=CTR(C6.), COMPARE AT TARGET
6659 NEXT, J/GOBUT117C2
(4677) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.001.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.000.110.101]
6660
6661 4065: 1(FREE)
6662 GOBUT117C2:
6663 SETUP, RETURN/SCOPE117C, |RETURN TO SCOPE LOOP TEST WORD
6664 NEXT, GOTC-PAGE(7), |BUT TABLE
6665 J/BUTD[C] |BIT<02> = D[C]H
(4065) DCS[0.00.0.0.0.0] BM[0100.00.00.01.10.110.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.0.011.001.000]
6666
6667
6668
6669 4066: 1(FREE)
6670 SCOPE117C:
6671 NEXT, RUTD[SCOPE], |NO ERROR: "TEST120A1" (+1, WORDS)
6672 J/TEST120A1 |ERROR: "ALUI17A1" (-20, WORDS)
(4066) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.110.110.111]
6673
6674 | - - - - -
6675
6676 |THIS NEXT SET OF 7, TESTS EXERCIZES THE ALU FUNCTION
6677 |"A-AND-B", AND THE CARRYOUT OF "CINMUX = (0) INTO D[C]
6678
6679 | - - - - -
6680
6681 1TFSTS 120A 1-3 VERIFIES THAT WITH:
6682 |ALU=(A-AND-P), A=(125252), B=(177777), THEN D=(125252)
6683 4667:
6684 TEST120A1:
6685 PO, LOAD-ENUA(ZTARGET412), |BIT<15:12> = "1010"
6686 LOAD-ERROR(TEST120A1), ERROR DIRECTORY KEY
6687 DCS=CTR(C6.), COMPARE AT TARGET
6688 NEXT, J/ALUI20A1
(4667) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.001.010.0.0.0.0.0.0.0.0000.0.0.0000.0.11.000.0.110.110.100]
6689
6690 4664:
6691 ALUI20A1:
6692 PO, RUMP-VERIFY, |COUNT
6693 P2-T, D_A=AND-B, D[C]_ALU15, |ALU=(A-AND-B), D[C]=(1)
6694 BUS=A_ASPHI(C125252), |A=(125252)
6695 BUS=B_CSPD(C177777), |B=(177777)
6696
6697 NEXT, J/GOBUT120A1 |D=(125252)
(4664) DCS[0.00.0.0.0.1] BM[1011.10.00.11.01.110.100.0.0.1.0.0.0.0.0101.0.0.0000.0.11.000.0.000.110.111]
6698
6699 4067: 1(FREE)
6700 GOBUT120A1:
6701 SETUP, RETURN/TEST120A2, EXEC SUBR WHICH:
6702 | (1) D<15:12> -> IR<15:12>

```

```
6703      NEXT, CALL(D15-12)          ;(2) BUT(IR15-12) INTO ZTARGET---
(4067) DCS(0.00,0.0,0.0) BM(0100.00,11.01,01.101.111.00,0.0,0.0,0.0,0.0000...0.0000,0...11,100...010,011,100)
6704
6705      |CHECK BIT <11:06> = "1010 10"
6706      4655:
6707      TEST120A2:
6708      PO,          LOAD=ENUA(ZTARGET412),          ;INSTRS-E78 OUTPUT FOR BIT <11:06> = "101 010"
6709      LOAD=ERROR(TEST120A2),          ;ERROR DIRECTORY KEY
6710      DCS=CTR(C6,);          ;COMPARE AT TARGET
6711      BUMP=VERIFY,          ;COUNT
6712      JGOBUT120A2
(4655) DCS(1.00,1.0,0.0) BM(1001.00,11.11,00.001.010...0.0,0.0,0.0,0.0000...0.0000,0...11,000...000,111,000)
6713
6714      4070: I(FREE)
6715      GOBUT120A2:
6716      SETUP,      RETURN/TEST120A3,          ;EXEC SUBR WHICH:
6717      NEXT, CALL(D11-06)          ;(1) D<11:06> -> IR<11:06>
6718      (4070) DCS(0.00,0.0,0.0) BM(0100.00,11.01,01.100.111.00,0.0,0.0,0.0,0.0000...0.0000,0...11,100...010,011,110)
6719
6720
6721      |CHECK BIT<05:00> = "10 1010"
6722      4654:
6723      TEST120A3:
6724
6725      PO,          LOAD=ENUA(ZTARGET425),          ;INSTRS-E88 OUTPUT FOR BIT<05:00>="101 010"
6726      LOAD=ERROR(TEST120A3),          ;ERROR DIRECTORY KEY
6727      DCS=CTR(C6,);          ;COMPARE AT TARGET
6728      BUMP=VERIFY,          ;COUNT
6729      JGOBUT120A3
(4654) DCS(1.00,1.0,0.0) BM(1001.00,11.11,00.010.101...0.0,0.0,0.0,0.0000...0.0000,0...11,000...000,111,001)
6730
6731      4071: I(FREE)
6732      GOBUT120A3:
6733      SETUP,      RETURN/TEST120B1,          ;EXEC SUBR WHICH:
6734      NEXT, CALL(D05-00)          ;(1) D<05:00> -> IR<05:00>
6735      (4071) DCS(0.00,0.0,0.0) BM(0100.00,11.01,01.011.111.00,0.0,0.0,0.0,0.0000...0.0000,0...11,100...010,100,000)
6736
6737
6738
6739
6740
6741
6742      |TESTS 120B 1-4 VERIFIES THAT WITH:
6743      |ALU=(A-AND-B), A=(052525), B=(177777), THEN D=(052525)
6744      4653:
6745      TEST120B1:
6746      PO,          LOAD=ENUA(ZTARGET405),          ;BIT<15:12> = "0101"
6747      LOAD=ERROR(TEST120B1),          ;ERROR DIRECTORY KEY
6748      DCS=CTR(C6,);          ;COMPARE AT TARGET
```

```
6749      NEXT, J/ALU120B1
(4653) DCS(1.00,1.0,0.0) BM(1001.00,11.11,00.000.101...0.0,0.0,0.0,0.0000...0.0000,0...11,000...000,111,010)
6750
6751      4072: I(FREE)
6752      ALU120B1:
6753
6754      P2-T,      D=A-AND=B, D[C]=CINMUX,          ;ALU=(A-AND-B), D[C]=CIN=(0)
6755      BUS=A-ASPHI(C052525),          ;A=(052525)
6756      BUS=B-CSPD(C177777),          ;B=(177777)
6757      JGOBUT120B1
(4072) DCS(0.00,0.0,0.0) BM(1011.10,00.11,01.111.000...0.0,0.0,0.0,0.0101...0.0000,0...11,000...000,111,011)
6758
6759      4073: I(FREE)
6760      GOBUT120B1:
6761      SETUP,      RETURN/TEST120B2,          ;EXEC SUBR WHICH:
6762      NEXT, CALL(D15-12)          ;(1) D<15:12> -> IR<15:12>
6763      (4073) DCS(0.00,0.0,0.0) BM(0100.00,11.01,01.010.111.00,0.0,0.0,0.0,0.0000...0.0000,0...11,100...010,011,100)
6764
6765      |CHECK BIT<11:06> = "0101 01"
6766      4652:
6767      TEST120B2:
6768      PO,          LOAD=ENUA(ZTARGET405),          ;INSTRS-E78 OUTPUT FOR BIT<11:06>="010 101"
6769      LOAD=ERROR(TEST120B2),          ;ERROR DIRECTORY KEY
6770      DCS=CTR(C6,);          ;COMPARE AT TARGET
6771      BUMP=VERIFY,          ;COUNT
6772      JGOBUT120B2
(4652) DCS(1.00,1.0,0.0) BM(1001.00,11.11,00.000.101...0.0,0.0,0.0,0.0000...0.0000,0...11,000...000,111,100)
6773
6774      4074: I(FREE)
6775      GOBUT120B2:
6776      SETUP,      RETURN/TEST120B3,          ;EXEC SUBR WHICH:
6777      NEXT, CALL(D11-06)          ;(1) D<11:06> -> IR<11:06>
6778      (4074) DCS(0.00,0.0,0.0) BM(0100.00,11.01,01.001.111.00,0.0,0.0,0.0,0.0000...0.0000,0...11,100...010,011,110)
6779
6780
6781
6782      |CHECK BIT<05:00> = "01 0101"
6783      4651:
6784      TEST120B3:
6785
6786      PO,          LOAD=ENUA(ZTARGET432),          ;INSTRS-E88 OUTPUT FOR BIT<05:00>="010 101"
6787      LOAD=ERROR(TEST120B3),          ;ERROR DIRECTORY KEY
6788      DCS=CTR(C6,);          ;COMPARE AT TARGET
6789      BUMP=VERIFY,          ;COUNT
6790      JGOBUT120B3
(4651) DCS(1.00,1.0,0.0) BM(1001.00,11.11,00.011.010...0.0,0.0,0.0,0.0000...0.0000,0...11,000...000,111,101)
6790
6791      4075: I(FREE)
6792      GOBUT120B3:
6793      SETUP,      RETURN/TEST120B4,          ;EXEC SUBR WHICH:
6794      (4075) DCS(0.00,0.0,0.0) BM(0100.00,11.01,01.001.111.00,0.0,0.0,0.0,0.0000...0.0000,0...11,100...010,011,110)
```

```
6795      NEXT, CALL[D05=00]          (2) BUT(INSTR5) INTO ZTARGET---
(4075) DCS[0,00,0,0,0,0] BM[0100,00,11,01,01,000,011,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,100,0,010,100,000]
6796
6797      !CHECK THAT D[C] GOT A (0) FROM "CINMUX" = (0)
6798      4650:
6799      TEST120B4:
6800      PO,          LOAD=ENUA(ZTARGET402),          !BIT<00> = D[C] = (0)
6801                  LOAD=ERROR(TEST120B4),          !ERROR DIRECTORY KEY
6802                  DCS=CTR(C3,),                    !COMPARE AT TARGET
6803                  BUMP=VERIFY,                      !COUNT
6804                  J/GOBUT120B4
(4650) DCS[1,00,1,0,0,0,1] BM[1100,00,11,11,00,000,010,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,000,0,000,111,110]
6805
6806      4076: 1(FREE)
6807      GOBUT120B4:
6808      SETUP,      RETURN/SCOPE120B,          !RETURN TO SCOPE LOOP TEST WORD
6809      NEXT,        GOTO=PAGE(7),            !BUT TABLE
6810                  J/BUTD(C)A                !D[C] IN BIT <00>
(4076) DCS[0,00,0,0,0,0,0] BM[0100,00,00,01,11,11,11,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,100,0,011,100,100]
6811
6812      4077: 1(FREE)
6813      SCOPE120B:
6814      NEXT,        BUTD[SCOPE],              !NO ERROR: "TEST121A1" (+1, WORDS)
6815                  J/TEST121A1                !ERROR: "ALU120A1" (-15, WORDS)
(4077) DCS[0,00,0,1,0,0,0] BM[0000,00,00,00,00,0,000,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,000,0,110,110,101]
6816
6817
6818
6819
6820
6821      1 - - - - -
6822
6823      !THIS NEXT SET OF 16, TESTS EXERCISES THE ALU FUNCTION
6824      !"A-XOR-B, AND THE CARRYOUT FUNCTIONS OF "ALU00" AND "ALU07" INTO D[C]
6825      1 - - - - -
6826
6827
6828
6829      !TESTS 121A 1-4 VERIFIES THAT WITH:
6830      !ALU=(A-XOR-B), A=(000000), B=(052525), THEN D=(052525)
6831      4665:
6832      TEST121A1:
6833      PO,          LOAD=ENUA(ZTARGET405),          !BIT<15:12> = "0101"
6834                  LOAD=ERROR(TEST121A1),          !ERROR DIRECTORY KEY
6835                  DCS=CTR(C6,),                    !COMPARE AT TARGET
6836                  J/ALU121A1
(4665) DCS[1,00,1,0,0,0,0] BM[1001,00,11,11,00,000,010,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,000,0,111,000,110]
6837
6838      4706:
6839      ALU121A1:
6840      PO,          PUMP=VERIFY,                  !COUNT
6841      P2=T,        D_A=XOR=B, D[C]=ALU00,        !ALU=(A-XOR-B), D[C]=ALU00=(1)
```

```
6842      BUS-A_ASPI(C000000),          !A=(000000)
6843      BUS-B_CSPD(C052525),            !B=(052525)
6844      NEXT,        J/GOBUT121A1          !D=(052525)
(4706) DCS[0,00,0,0,0,0,1] BM[0110,10,00,11,01,100,010,0,0,0,0,0,0,0,0111,0,0,0,0000,0,0,11,000,0,001,000,011]
6845
6846      4103: 1(FREE)
6847      GOBUT121A1:
6848      SETUP,      RETURN/TEST121A2,          !EXEC SUBR WHICH:
6849      NEXT,        CALL[D15=12]              ! (1) D<15:12> => IR<15:12>
6850                  (2) BUT(IR15-12) INTO ZTARGET---
(4103) DCS[0,00,0,0,0,0,0] BM[0100,00,11,01,11,110,011,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,100,0,010,011,100]
6851
6852      !CHECK BIT<11:06> = "0101 01"
6853      4676:
6854      TEST121A2:
6855      PO,          LOAD=ENUA(ZTARGET405),          !INSTR5-Z78 OUTPUT FOR BIT<11:06>="010 101"
6856                  LOAD=ERROR(TEST121A2),          !ERROR DIRECTORY KEY
6857                  DCS=CTR(C6,),                    !COMPARE AT TARGET
6858                  BUMP=VERIFY,                      !COUNT
6859                  J/GOBUT121A2
(4676) DCS[1,00,1,0,0,0,1] BM[1001,00,11,11,00,000,010,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,000,0,001,000,100]
6860
6861      4104: 1(FREE)
6862      GOBUT121A2:
6863      SETUP,      RETURN/TEST121A3,          !EXEC SUBR WHICH:
6864      NEXT,        CALL[D11=06]              ! (1) D<11:06> => IR<11:06>
6865                  (2) BUT(IR11-06) INTO ZTARGET---
(4104) DCS[0,00,0,0,0,0,0] BM[0100,00,11,01,11,101,011,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,100,0,010,011,110]
6866
6867
6868
6869      !CHECK BIT<05:00> = "01 0101"
6870      4675:
6871      TEST121A3:
6872      PO,          LOAD=ENUA(ZTARGET432),          !INSTR5-Z88 OUTPUT FOR BIT<05:00>="010 101"
6873                  LOAD=ERROR(TEST121A3),          !ERROR DIRECTORY KEY
6874                  DCS=CTR(C6,),                    !COMPARE AT TARGET
6875                  J/GOBUT121A3
(4675) DCS[1,00,1,0,0,0,0] BM[1001,00,11,11,00,011,010,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,000,0,001,000,101]
6876
6877      4105: 1(FREE)
6878      GOBUT121A3:
6879      SETUP,      RETURN/TEST121A4,          !EXEC SUBR WHICH:
6880      NEXT,        CALL[D05=00]              ! (1) D<05:00> => IR<05:00>
6881                  (2) BUT(INSTR5) INTO ZTARGET---
(4105) DCS[0,00,0,0,0,0,0] BM[0100,00,11,01,11,100,011,0,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,100,0,010,100,000]
6882
6883      !CHECK THAT D[C] GOT A (1) FROM "ALU00"
6884      4674:
6885      TEST121A4:
6886      PO,          LOAD=ENUA(ZTARGET403),          !BIT<01> = D[C] = (1)
```

```

6887      LOAD-ERROR(TEST121A4),      |ERROR DIRECTORY KEY
6888      DCS-CTR(C3.),                |COMPARE AT TARGET
6889      BUMP-VERIFY,                  |COUNT
6890      J/GOBUT121A4                  |
(4674) DCS(1.00,1.0,0.1) BM(1100.00,11.11.00.000.011...0.0,0.0.0.0...0.0000...0.0000.0...11.000...001.000,110)
6891      4106: 1(FREE)
6892      GOBUT121A4
6893      SETUP, RETURN/TEST121B1,      |RETURN TO START OF NEXT SUBTEST
6894      NEXT, GOTO=PAGE(7),          |BUT TABLE
6895      J/BUTD(C)B                    |D(C)N IN BIT<01>
6896      (4106) DCS(0.00,0.0,0.0) BM(0100.00,11.01.10.011...111...0.0,0.0.0.0...0.0000...0.0000.0...11.100...011.101,000)
6897
6898
6899
6900
6901      ! - - - - -
6902
6903      !TESTS 121B 1-4 VERIFIES THAT WITH:
6904      !ALU=(A-XOR-B), A=(177777), B=(052525), THEN D=(125252)
6905      4663:
6906      TEST121B1:
6907      PO, LOAD-ENUA(ZTARGET412),      |BIT<15:12> = "1010"
6908      LOAD-ERROR(TEST121B1),        |ERROR DIRECTORY KEY
6909      DCS-CTR(C6.),                 |COMPARE AT TARGET
6910      NEXT, J/ALU121B1              |
(4663) DCS(1.00,1.0,0.0) BM(1001.00,11.11.00.001...010...0.0,0.0.0.0...0.0000...0.0000.0...11.000...001.000,111)
6911      4107: 1(FREE)
6912      ALU121B1:
6913      PO, BUMP-VERIFY,              |COUNT
6914      P2-T, P_A-XOR-B, D(C)_ALU00,  |ALU=(A-XOR-B), D(C)=ALU00=(0)
6915      BUS-A_ASPHI(C177777),         |A=(177777)
6916      BUS-B_CSPD(C052525),          |B=(052525)
6917      J/GOBUT121B1                  |D=(125252)
6918      (4107) DCS(0.00,0.0,0.1) BM(0110.10,10.01.10.101...010...0.1,0.0.0.0...0.011...0...0000.0...11.000...001.001,000)
6919
6920      4110: 1(FREE)
6921      GOBUT121B1:
6922      SETUP, RETURN/TEST121B2,      |EXEC SUBR WHICH:
6923      NEXT, CALL[D15-12]             |(1) D<15:12> -> IR<15:12>
6924      (4110) DCS(0.00,0.0,0.0) BM(0100.00,11.01.10.010...111...0.0,0.0.0.0...0.0000...0.0000.0...11.100...010.011,100)
6925      |CHECK BIT<11:06> = "1010 10"
6926      4662:
6927      TEST121B2:
6928      PO, LOAD-ENUA(ZTARGET412),      |INSTRS-E78 OUTPUT FOR BIT<11:06>="101 010"
6929      LOAD-ERROR(TEST121B2),        |ERROR DIRECTORY KEY
6930      DCS-CTR(C6.),                 |COMPARE AT TARGET
6931      BUMP-VERIFY,                  |COUNT
6932      NEXT, J/GOBUT121B2            |
6933

```

```

(4667) DCS(1.00,1.0,0.1) BM(1001.00,11.11.00.001...010...0.0,0.0.0.0...0.0000...0.0000.0...11.000...001.001,001)
6934      4111: 1(FREE)
6935      GOBUT121B2:
6936      SETUP, RETURN/TEST121B3,      |EXEC SUBR WHICH:
6937      NEXT, CALL[D11-06]             |(1) D<11:06> -> IR<11:06>
6938      (4111) DCS(0.00,0.0,0.0) BM(0100.00,11.01.10.001...111...0.0,0.0.0.0...0.0000...0.0000.0...11.100...010.011,110)
6939
6940
6941
6942
6943      |CHECK BIT<05:00> = "10 1010"
6944      4661:
6945      TEST121B3:
6946      PO, LOAD-ENUA(ZTARGET425),      |INSTRS-E88 OUTPUT FOR BIT<05:00>="101 010"
6947      LOAD-ERROR(TEST121B3),        |ERROR DIRECTORY KEY
6948      DCS-CTR(C6.),                 |COMPARE AT TARGET
6949      NEXT, J/GOBUT121B3            |
(4661) DCS(1.00,1.0,0.0) BM(1001.00,11.11.00.010...101...0.0,0.0.0.0...0.0000...0.0000.0...11.000...001.001,010)
6951      4112: 1(FREE)
6952      GOBUT121B3:
6953      SETUP, RETURN/TEST121B4,      |EXEC SUBR WHICH:
6954      NEXT, CALL[D05-00]             |(1) D<05:00> -> IR<05:00>
6955      (4112) DCS(0.00,0.0,0.0) BM(0100.00,11.01.10.000...111...0.0,0.0.0.0...0.0000...0.0000.0...11.100...010.100,000)
6956
6957      |CHECK THAT D(C) GOT A (0) FROM "ALU00"
6958      4660:
6959      TEST121B4:
6960      PO, LOAD-ENUA(ZTARGET401),      |BIT<01> = D(C) = (0)
6961      LOAD-ERROR(TEST121B4),        |ERROR DIRECTORY KEY
6962      DCS-CTR(C3.),                 |COMPARE AT TARGET
6963      BUMP-VERIFY,                  |COUNT
6964      NEXT, J/GOBUT121B4            |
(4660) DCS(1.00,1.0,0.1) BM(1100.00,11.11.00.000.001...0.0,0.0.0.0...0.0000...0.0000.0...11.000...001.001,011)
6966      4113: 1(FREE)
6967      GOBUT121B4:
6968      SETUP, RETURN/SCOPE121C,      |RETURN TO SCOPE LOOP TEST WORD
6969      NEXT, GOTO=PAGE(7),          |BUT TABLE
6970      J/BUTD(C)B                    |D(C)N IN BIT<01>
6971      (4113) DCS(0.00,0.0,0.0) BM(0100.00,10.01.100.111...0.0,0.0.0.0...0.0000...0.0000.0...11.100...011.101,000)
6972
6973
6974
6975      4114: 1(FREE)
6976      SCOPE121C:
6977      NEXT, BUTD[SCOPE],             |NO ERROR: "TEST121C1" (+1. WORDS)
6978      J/TFST121C1                  | ERROR: "ALU121A1" (-17. WORDS)

```

```
(4114) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000...0.0000.0...11.000...111.000.111]
6979
6980
6981
6982 1 - - - - -
6983
6984 17F5T8 121C 1-4 VERIFIES THAT WITH:
6985 1ALU=(A-XOR-B), A=(000000), B=(125252), THEN D=(125252)
6986
6987 TEST121C1:
6988 PO, LOAD-ENUA(ZTARGET412), 1BIT<15:12> = "1010"
6989 LOAD-ERROR(TEST121C1), 1ERROR DIRECTORY KEY
6990 DCS-CTR(C6.), 1COMPARE AT TARGET
6991 NEXT, J/ALU121C1 1
(4707) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.001.010...0.0.0.0.0.0.0.0.0000...0.0000.0...11.000...110.111.000]
6992
6993 46701
6994 ALU121C1:
6995 PO, BUMP-VERIFY, 1COUNT
6996 D_A-XOR-B, D[C]=ALU07, 1ALU=(A-XOR-B), D[C]=ALU07=(1)
6997 BUS-A_ASPhi(C000000), 1A=(000000)
6998 BUS-B_CSPD(C125252), 1B=(125252)
6999 NEXT, J/GOBUT121C1 1D=(125252)
(4670) DCS[0.00.0.0.0.1] BM[0110.10.00.11.01.100.011...0.1.0.0.0.0.0.0.0110...0.0000.0...11.000...001.001.101]
7000
7001 41151 1(FREE)
7002 GOBUT121C1:
7003 SETUP, RETURN/TEST121C2, 1EXEC SUBR WHICH:
7004 NEXT, CALL[D15-12] 1(1) D<15:12> -> IR<15:12>
7005 (4115) DCS[0.00.0.0.0.0] BM[0100.00.11.00.00.101.111...0.0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.011.100] 1(2) BUT(IR15-12) INTO ZTARGET---
7006
7007 1CHECK BIT<11:06> = "1010 10"
7008 46051
7009 TEST121C2:
7010 PO, LOAD-ENUA(ZTARGET412), 1INSTR6=ET8 OUTPUT FOR BIT<11:06>="101 010"
7011 LOAD-ERROR(TEST121C2), 1ERROR DIRECTORY KEY
7012 DCS-CTR(C6.), 1COMPARE AT TARGET
7013 BUMP-VERIFY, 1COUNT
7014 NEXT, J/GOBUT121C2 1
(4605) DCS[1.00.1.0.0.1] BM[1001.00.11.11.00.001.010...0.0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.001.110]
7015
7016 41161 1(FREE)
7017 GOBUT121C2:
7018 SETUP, RETURN/TEST121C3, 1EXEC SUBR WHICH:
7019 NEXT, CALL[D11-06] 1(1) D<11:06> -> IR<11:06>
7020 (4116) DCS[0.00.0.0.0.0] BM[0100.00.11.10.01.111.111...0.0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.011.110] 1(2) BUT(INSTR6) INTO ZTARGET---
7021
7022 1CHECK BIT<05:00> = "10 1010"
7023 47171
7024
```

```
7025 TEST121C3:
7026
7027 PO, LOAD-ENUA(ZTARGET425), 1INSTR5=E88 OUTPUT FOR BIT<05:00>="101 010"
7028 LOAD-ERROR(TEST121C3), 1ERROR DIRECTORY KEY
7029 DCS-CTR(C6.), 1COMPARE AT TARGET
7030 NEXT, J/GOBUT121C3 1
(4717) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.010.101...0.0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.001.111]
7031
7032 41171 1(FREE)
7033 GOBUT121C3:
7034 SETUP, RETURN/TEST121C4, 1EXEC SUBR WHICH:
7035 NEXT, CALL[D05-00] 1(1) D<05:00> -> IR<05:00>
7036 (4117) DCS[0.00.0.0.0.0] BM[0100.00.11.10.01.110.111...0.0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.100.000] 1(2) BUT(INSTR5) INTO ZTARGET---
7037
7038 1CHECK THAT D[C] GOT A (1) FROM "ALU07"
7039 47161
7040 TEST121C4:
7041 PO, LOAD-ENUA(ZTARGET417), 1BIT<02> = D[C] = (1)
7042 LOAD-ERROR(TEST121C4), 1ERROR DIRECTORY KEY
7043 DCS-CTR(C3.), 1COMPARE AT TARGET
7044 BUMP-VERIFY, 1COUNT
7045 NEXT, J/GOBUT121C4 1
(4716) DCS[1.00.1.0.0.1] BM[1100.00.11.11.00.001.111...0.0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.010.000]
7046
7047 41201 1(FREE)
7048 GOBUT121C4:
7049 SETUP, RETURN/TEST121D1, 1RETURN TO START OF NEXT SUBTEST
7050 NEXT, GOTO-PAGE(7), 1BUT TABLE
7051 J/BUTD[C] 1D[C]N IN BIT<02>
(4720) DCS[0.00.0.0.0.0] BM[0100.00.11.10.10.111.111...0.0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.001.000]
7052
7053
7054
7055 1 - - - - -
7056
7057 17F5T8 121D 1-4 VERIFIES THAT WITH:
7058 1ALU=(A-XOR-B), A=(177777), B=(125252), THEN D=(052525)
7059
7060 47271
7061 TEST121D1:
7062 PO, LOAD-ENUA(ZTARGET405), 1BIT<15:12> = "0101"
7063 LOAD-ERROR(TEST121D1), 1ERROR DIRECTORY KEY
7064 DCS-CTR(C3.), 1COMPARE AT TARGET
7065 NEXT, J/ALU121D1 1
(4727) DCS[1.00.1.0.0.0] BM[1001.00.11.11.00.000.101...0.0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.010.001]
7066
7067 41211 1(FREE)
7068 ALU121D1:
7069 PO, BUMP-VERIFY, 1COUNT
7070 D_A-XOR-B, D[C]=ALU07, 1ALU=(A-XOR-B), D[C]=ALU07=(0)
7071 BUS-A_ASPhi(C177777), 1A=(177777)
```

```

0772          BUS-B,CSPD(C125252),          ID=(125252)
7073          NEXT,      J/GOBUT121D1;      ID=(052525)
(4121) DCS[0.00,0.0,0.0,1] BM[0110..10.00..11.01..101..011...0.1,0..0..0.0110...0..0000,0...11.000...001,010,010]

7074          4122: 1(FREE)
7075          GOBUT121D1;
7076          SETUP,      RETURN/TEST121D2,          IEXEC SUBR WHICH:
7077          NEXT,      CALL[D15=12]          I(1) D<15:12> => IR<15:12>
7078          NEXT,      CALL[D15=12]          I(2) BUT(IR15-12) INTO STARGET---
7079          (4122) DCS[0.00,0.0,0.0,0] BM[0100..00.11..10.10..110..111...0.0,0..0..0..0.0000...0..0000,0...11.100...010,011,100]

7080          ICHECK BIT<11:06> = "0101 01"
7081          4726:
7082          TEST121D2;
7083          PO,          LOAD-ENUA(ZTARGET405),          INSTRS=E78 OUTPUT FOR BIT<11:06>="010 101"
7084          LOAD-ERROR(TEST121D2),          ERROR DIRECTORY KEY
7085          DCS-CTRC6,),          COMPARE AT TARGET
7086          BUMP-VERIFY,          COUNT
7087          J/GOBUT121D2
7088          (4726) DCS[1.00,1.0,0.0,1] BM[1001..00.11..11.00..000..101...0.0,0..0..0..0.0000...0..0000,0...11.000...001,010,011]

7089          4123: 1(FREE)
7090          GOBUT121D2;
7091          SETUP,      RETURN/TEST121D3,          IEXEC SUBR WHICH:
7092          NEXT,      CALL[D11=06]          I(1) D<11:06> => IR<11:06>
7093          (4123) DCS[0.00,0.0,0.0,0] BM[0100..00.11..10.11..111..111...0.0,0..0..0..0.0000...0..0000,0...11.100...010,011,110]
7094          I(2) BUT(INSTRS) INTO STARGET---

7095          ICHECK BIT<05:00> = "01 0101"
7096          4737:
7097          TEST121D3;
7098          PO,          LOAD-ENUA(ZTARGET432),          INSTRS=E88 OUTPUT FOR BIT<05:00>="010 101"
7099          LOAD-ERROR(TEST121D3),          ERROR DIRECTORY KEY
7100          DCS-CTRC6,),          COMPARE AT TARGET
7101          NEXT,      J/GOBUT121D3
7102          (4737) DCS[1.00,1.0,0.0,0] BM[1001..00.11..11.00..011..010...0.0,0..0..0..0.0000...0..0000,0...11.000...001,010,100]

7103          4124: 1(FREE)
7104          GOBUT121D3;
7105          SETUP,      RETURN/TEST121D4,          IEXEC SUBR WHICH:
7106          NEXT,      CALL[D05=00]          I(1) D<05:00> => IR<05:00>
7107          (4124) DCS[0.00,0.0,0.0,0] BM[0100..00.11..10.11..110..111...0.0,0..0..0..0.0000...0..0000,0...11.100...010,100,000]
7108          I(2) BUT(INSTRS) INTO STARGET---

7109          ICHECK THAT D[C] GOT A (0) FROM "ALU07"
7110          4736:
7111          TEST121D4;
7112          PO,          LOAD-ENUA(ZTARGET413),          IBIT<02> = D[C] = (0)

```

```

7117      LOAD-ERROR(TEST121D4),                                |ERROR DIRECTORY KEY
7118      DCS-CTR(C3.),                                           |COMPARE AT TARGET
7119      RUMP-VERIFY,                                             |COUNT
7120      J/GOBUT121D4                                           |
(4736) DCS[1.00,1.0,0,1] BM[1100..00,11..11,00..001..011...0,0,0..0..0..0.0000...0..0000,0...11,000...001,010,101]

7121
7122      4125: 1(FREE)
7123      GOBUT121D4:
7124      SETUP, RETURN/SCOPE121D,                                |RETURN TO SCOPE LOOP TEST WORD
7125      NEXT,   GOTO=PAGE(7),                                   |BUT TABLE
7126      J/BUD[C]C                                              |DICJH IN BIT<02>
(4125) DCS[0.00,0.0,0,0] BM[0100..00,00..10,10..110..111...0,0,0..0..0..0.0000...0..0000,0...11,100...011,001,000]

7127
7128      4126: 1(FREE)
7129      SCOPE121D:
7130      PO,                                                    |COUNT
7131      NEXT,   RUTD[SCOPE],                                    |NO ERROR: "TEST122A1" (+1, WORDS)
7132      J/TEST122A1                                           |ERROR: "ALU121C1" (-17, WORDS)
(4126) DCS[0.00,0.1,0,1] BM[0000..00,00..00,00..000..000...0,0,0..0..0..0.0000...0..0000,0...11,000...110,111,001]

7133
7134
7135
7136
7137
7138
7139
7140
7141      |-----
7142
7143      |THIS NEXT SET OF 8. TESTS CHECK THE ALU FUNCTIONS "A"
7144      |AND "A-IOR=B", THE FUNCTION D[C]=CINMUX(1), AND ALSO
7145      |THE BUT(D<14:100>=ZERO&D<15>) BUT FOR D<0>(00000) AND
7146      |D<1>(25252)
7147      4671:
7148      TEST122A1:
7149      PO,            LOAD-ENUA(ZTARGET434),                    |INSTR5 FOR IIR(000000)
7150      LOAD-ERROR(TEST122A1),                                  |ERROR DIRECTORY KEY
7151      DCS-CTR(C6.),                                           |COMPARE AT TARGET
7152      NEXT,          J/ALU122A1                               |
(4671) DCS[1.00,1.0,0,0] BM[1001..00,11..11,00..011..100...0,0,0..0..0..0.0000...0..0000,0...11,000...110,111,010]

7153
7154      4672:
7155      ALU122A1:
7156      PO,            RUMP-VERIFY,                               |COUNT
7157      P2-T,          D_A-TOR-B, D[C]=CINMUX,                  |ALU=(A-IOR-B), D[C]=CINMUX(1)
7158      BUS=A_ASPhi(C000000),                                  |A=(000000)
7159      BUS=B_ASPhi(C000000),                                  |B=(000000)
7160      NEXT,          J/GOBUT122A1                             |D=(000000)
(4672) DCS[0.00,0.0,0,1] BM[1110..01,11..11,01..100..000...0,1,0..0,0..0.0000...0..0000,0...11,000...001,010,111]

7161
7162      4127: 1(FREE)
7163      GOBUT122A1:
7164      SETUP, RETURN/TEST122A2,                                |EXEC SUBR WHICH:

```

```

7165          (1) D -> IR
7166      NEXT, CALL(DZERO)          (2) BUT(INSTRS) INTO ETARGET---
(4127) DCS[0,00,0,0,0,0] BM[0100,00,10,11,11,000,011,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,100,0,010,100,010]

7167
7168
7169
7170      [CHECK THAT D[C] GOT A (1) FROM CINMUX=(1)
7171      45701
7172      TEST122A2:
7173          PO,          LOAD-ENUA(ZTARGET403),          [BIT<00> = D[C] = (1)
7174          LOAD-ERROR(TEST122A2),          [ERROR DIRECTORY KEY
7175          DCS-CTR(C3,),          [COMPARE AT TARGET
7176          NEXT, J/GOBUT122A2
(4570) DCS[1,00,1,0,0,0] BM[1100,00,11,11,00,000,011,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,000,0,001,011,000]

7177
7178      4130: 1(FREE)
7179      GOBUT122A2:
7180          SETUP, RETURN/TEST122A3,          [RETURN TO START OF NEXT SUBTEST
7181          NEXT, GOTO-PAGE(7),          [BUT TABLE
7182          J/BUTD(C3A)          [D[C] IN BIT <00>
(4130) DCS[0,00,0,0,0,0] BM[0100,00,11,10,11,100,011,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,100,0,011,100,100]

7183
7184
7185
7186      [CHECK THAT D<15>=0, [D<14:00>=ZERO]=1 WHEN D=(000000)
7187      47341
7188      TEST122A3:
7189          PO,          LOAD-ENUA(ZTARGET402),          [SETUP FOR D<15:00>=ZERO
7190          LOAD-ERROR(TEST122A3),          [ERROR DIRECTORY KEY
7191          DCS-CTR(C3,),          [COMPARE AT TARGET
7192          NEXT, J/GOBUT122A3
(4734) DCS[1,00,1,0,0,0] BM[1100,00,11,11,00,000,010,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,000,0,001,011,001]

7193
7194      4131: 1(FREE)
7195      GOBUT122A3:
7196          SETUP, RETURN/TEST122A4,          [RETURN TO START OF NEXT SUBTEST
7197          NEXT, GOTO-PAGE(7),          [BUT TABLE
7198          J/BUTD-15-ZERO          [BIT<1>0 = D<15>= D<14:00>=ZERO
(4131) DCS[0,00,0,0,0,0] BM[0100,00,11,10,11,101,011,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,100,0,011,100,001]

7199
7200      [CHECK THAT D<15>=(1), [D<14:00>=ZERO]=(0) WHEN D=(125252)
7201      47351
7202      TEST122A4:
7203          PO,          LOAD-ENUA(ZTARGET401),          [SETUP FOR D<15:00>=(125252)
7204          LOAD-ERROR(TEST122A4),          [ERROR DIRECTORY KEY
7205          DCS-CTR(C4,),          [COMPARE AT TARGET
7206          BUMP-VERIFY,          [COUNT
7207          NEXT, J/SETD122A4
(4735) DCS[1,00,1,0,0,1] BM[1011,00,11,11,00,000,001,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,000,0,001,011,010]

7208
7209      4132: 1(FREE)

```

```

7210      SETD122A4:
7211          P2-T,          D_CSPD(C125252), D[C]_0,          [SETUP D FOR TEST
7212          NEXT, J/GOBUT122A4
(4132) DCS[0,00,0,0,0,0] BM[1010,10,00,00,000,000,0,0,0,0,0,0,0,0110,0,0,0,0000,0,11,000,0,001,011,011]

7213
7214      4133: 1(FREE)
7215      GOBUT122A4:
7216          SETUP, RETURN/SCOPE122A,          [RETURN TO SCOPE LOOP TEST WORD
7217          NEXT, GOTO-PAGE(7),          [BUT TABLE
7218          J/BUTD-15-ZERO          [BIT<1>0 = D15[ D<14:00>=ZERO]
(4133) DCS[0,00,0,0,0,0] BM[0100,00,00,10,11,100,011,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,100,0,011,100,001]

7219
7220
7221
7222      4134: 1(FREE)
7223      SCOPE122A:
7224          PO,          BUSRDIN_EMIT-[1],          [RESET PROC UCON
7225          EN-CLK-IR[15-00],          [
7226          NEXT, BUTD[SCOPE],          [NO ERROR: "TEST130A1" (+1, WORDS)
7227          J/TEST130A1          [ERROR: "ALU122A1" (-8, WORDS)
(4134) DCS[0,00,0,1,0,0] BM[0000,00,00,00,01,000,100,0,0,0,0,0,0,0,11001,0,0,0,0000,0,11,000,0,110,111,011]

7228
7229
7230
7231
7232
7233      1.PAGE=====
7234
7235
7236      .TOC * TEST130-136: ALU ARITHMETIC FUNCTION/CARRY LOOKAHEAD TESTS
7237
7238      1=====
7239      1*
7240      1*      TFST1: 130 - 136          UWORDS: 127 + 160
7241      1*
7242      1*      FUNCTIONS:
7243      1*
7244      1*      ALU ARITHMETIC FUNCTION DECODE, INTERNAL CARRIES, CARRYOUTS, CARRY LOOKAHEAD.
7245      1*
7246      1=====
7247
7248
7249      1      SUMMARY OF ALU ARITHMETIC / CARRY LOOKAHEAD TESTS:
7250      1
7251      1      TEST      OPERANDS EMPLOYED:
7252      1      NIMP      (A/B)+(B/A)+(CIN)=(COUT)(D)      ALU FUNCTION

7253      1      ----
7254      1
7255      1      130A      (0101)+(0101)+(0)=(0)(1010)      A-PLUS-B-PLUS-0
7256      1      130B      (1010)+(1010)+(1)=(1)(0101)      A-PLUS-B-PLUS-1
7257      1
7258      1      131A      (1010)+(0101)+(0)=(0)(1111)      A-PLUS-B-PLUS-PS[C]
7259      1      131B      (0101)+(1010)+(0)=(0)(1111)      DIVIDE/D[C]=0/A-PLUS-B-PLUS-0

```

```
7260 |
7261 | 132A (1010)=(1010)-(0)=(1)(0000) A-MINUS-B-MINUS-0
7262 | 132B (0101)=(0101)-(0)=(1)(0000) DIVIDE/D(C)=1/A-MINUS-B-MINUS-0
7263 |
7264 | 133A (1000)+(1000)+(0)=(1)(0000) A-PLUS-B-PLUS-D(C)
7265 | 133B (0111)+(0111)+(1)=(0)(1111) A-PLUS-NOT-B-PLUS-D(C)
7266 |
7267 | 134A (0100)+(1100)+(0)=(1)(0000) A-PLUS-B-PLUS-D(C)
7268 | 134B (1011)+(0011)+(1)=(0)(1111) A-PLUS-NOT-B-PLUS-D(C)
7269 |
7270 | 135A (1010)+(0110)+(0)=(1)(0000) A-PLUS-B-PLUS-D(C)
7271 | 135B (0101)+(1001)+(1)=(0)(1111) A-PLUS-NOT-B-PLUS-D(C)
7272 |
7273 | 136A (0101)+(1011)+(0)=(1)(0000) A-PLUS-B-PLUS-D(C)
7274 | 136B (1010)+(0100)+(1)=(0)(1111) A-PLUS-NOT-B-PLUS-D(C)
7275 |
7276 |
7277 |
7278 |
7279 | |CHECK INTERNAL ALU CARRIES WITH: (052525)+(052525)+(0)=(125252)
7280 | |ALSO CHECK ALU FUNCTION "A-PLUS-B-PLUS-0", D(C)_COUT15=(0)
7281 |
7282 | 4673:
7283 | TEST130A1:
7284 | PO, LOAD-ENUA(ZTARGET434), |SETUP FOR IR=(000000)/BUTINSTRS TEST
7285 | LOAD-ERROR(TEST130A1), |ERROR DIRECTORY KEY
7286 | DCS-CTR(C8), |COMPARE AT TARGET
7287 | J/ARITH130A1 |
(4673) DCS(1.00,1.0,0.0) BM(0111.00,11.11,00.011.100...0.0,0.0,0.0...0.0000...0.0000,0...11,000...101,100,000)
7288 |
7289 | 4540:
7290 | ARITH130A1:
7291 | P2-T, D_A-PLUS-B-PLUS-0, |ALU=(A-PLUS-B), CIN=(0)
7292 | D(C)_COUT15, |GET CARRYOUT
7293 | BUS-A_ASPhi(C052525), |A=(052525)
7294 | BUS-B_ASPhi(C052525), |B=(052525)
7295 | SR_A-PLUS-B-PLUS-0, |D=(125252), COUT15=(0)
7296 | NEXT, J/COMP130A1 |
(4540) DCS(0.00,0.0,0.0) BM(1001.01,11.11,01.111.110...0.1,1.0,0.0...0.0000...0.0000,0...11,000...001,011,101)
7297 |
7298 | 4135: 1(FREE)
7299 | COMP130A1:
7300 | PO, BUMP-VERIFY, |COUNT
7301 | P2-T, D_SR-XOR-CSPD(C125252), |COMPARE RECEIVED: EXPECTED
7302 | SAVE-D(C), |SAVE CARRY
7303 | NEXT, J/GOBUT130A1 |
(4135) DCS(0.00,0.0,0.1) BM(0110.10,00.00,00.000.111...0.1,0.0,0.0...0.0110...0.0000,0...11,000...001,011,110)
7304 |
7305 | 4136: 1(FREE)
7306 | GOBUT130A1:
7307 | SETUP, RETURN/TEST130A2, |RETURN TO START OF NEXT SUBTEST
7308 | NEXT, CALL(DINTOIR-S) |GO PUT D -> IR, BUT(INSTRS)
(4136) DCS(0.00,0.0,0.0) BM(0100.00,10.11,01.000.111...0.0,0.0,0.0...0.0000...0.0000,0...11,100...010,111,011)
7309 |
7310 |
7311 |
7312 | |CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7313 | 4550:
7314 | TEST130A2:
7315 | PO, LOAD-ENUA(ZTARGET402), |BIT<00> CLEAR
7316 | LOAD-ERROR(TEST130A2), |ERROR DIRECTORY KEY
7317 | DCS-CTR(C3), |COMPARE AT TARGET
7318 | NEXT, J/GOBUT130A2 |
(4550) DCS(1.00,1.0,0.0) BM(1100.00,11.11,00.000.010...0.0,0.0,0.0...0.0000...0.0000,0...11,000...001,011,111)
7319 |
7320 | 4137: 1(FREE)
7321 | GOBUT130A2:
7322 | SETUP, RETURN/TEST130B1, |RETURN TO START OF NEXT SUBTEST
7323 | NEXT, GOTO-PAGE(7), |BUT TABLE
7324 | J/BUTD(C)A |D(C)_H COUT15 H IN BIT<00>
(4137) DCS(0.00,0.0,0.0) BM(0100.00,10.10,11.000.111...0.0,0.0,0.0...0.0000...0.0000,0...11,100...011,100,100)
7325 |
7326 |
7327 |
7328 |
7329 |
7330 | |CHECK INTERNAL ALU CARRIES WITH: (125252)+(125252)+(1)=(052525)
7331 | |ALSO CHECK ALU FUNCTION "A-PLUS-B-PLUS-1", D(C)_COUT15=(1)
7332 |
7333 | 4530:
7334 | TEST130B1:
7335 | PO, LOAD-ENUA(ZTARGET434), |SETUP FOR IR=(000000)/BUTINSTRS TEST
7336 | LOAD-ERROR(TEST130B1), |ERROR DIRECTORY KEY
7337 | DCS-CTR(C8), |COMPARE AT TARGET
7338 | NEXT, J/ARITH130B1 |
(4530) DCS(1.00,1.0,0.0) BM(0111.00,11.11,00.011.100...0.0,0.0,0.0...0.0000...0.0000,0...11,000...001,100,000)
7339 |
7340 | 4140: 1(FREE)
7341 | ARITH130B1:
7342 | P2-T, D_A-PLUS-B-PLUS-1, |ALU=(A-PLUS-B), CIN=(1)
7343 | D(C)_COUT15, |GET CARRYOUT
7344 | BUS-A_ASPhi(C125252), |A=(125252)
7345 | BUS-B_ASPhi(C125252), |B=(125252)
7346 | SR_A-PLUS-B-PLUS-1, |D=(052525), COUT15=(1)
7347 | NEXT, J/COMP130B1 |
(4140) DCS(0.00,0.0,0.0) BM(1100.01,11.11,01.110.110...0.1,1.0,0.0...0.0000...0.0000,0...11,000...001,100,001)
7348 |
7349 | 4141: 1(FREE)
7350 | COMP130B1:
7351 | PO, BUMP-VERIFY, |COUNT
7352 | P2-T, D_SR-XOR-CSPD(C052525), |COMPARE RECEIVED: EXPECTED
7353 | SAVE-D(C), |SAVE CARRY
7354 | NEXT, J/GOBUT130B1 |
(4141) DCS(0.00,0.0,0.1) BM(0110.10,00.00,00.000.111...0.1,0.0,0.0...0.0111...0.0000,0...11,000...001,100,010)
7355 |
7356 | 4142: 1(FREE)
```

```
7309 |
7310 |
7311 |
7312 | |CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7313 | 4550:
7314 | TEST130A2:
7315 | PO, LOAD-ENUA(ZTARGET402), |BIT<00> CLEAR
7316 | LOAD-ERROR(TEST130A2), |ERROR DIRECTORY KEY
7317 | DCS-CTR(C3), |COMPARE AT TARGET
7318 | NEXT, J/GOBUT130A2 |
(4550) DCS(1.00,1.0,0.0) BM(1100.00,11.11,00.000.010...0.0,0.0,0.0...0.0000...0.0000,0...11,000...001,011,111)
7319 |
7320 | 4137: 1(FREE)
7321 | GOBUT130A2:
7322 | SETUP, RETURN/TEST130B1, |RETURN TO START OF NEXT SUBTEST
7323 | NEXT, GOTO-PAGE(7), |BUT TABLE
7324 | J/BUTD(C)A |D(C)_H COUT15 H IN BIT<00>
(4137) DCS(0.00,0.0,0.0) BM(0100.00,10.10,11.000.111...0.0,0.0,0.0...0.0000...0.0000,0...11,100...011,100,100)
7325 |
7326 |
7327 |
7328 |
7329 |
7330 | |CHECK INTERNAL ALU CARRIES WITH: (125252)+(125252)+(1)=(052525)
7331 | |ALSO CHECK ALU FUNCTION "A-PLUS-B-PLUS-1", D(C)_COUT15=(1)
7332 |
7333 | 4530:
7334 | TEST130B1:
7335 | PO, LOAD-ENUA(ZTARGET434), |SETUP FOR IR=(000000)/BUTINSTRS TEST
7336 | LOAD-ERROR(TEST130B1), |ERROR DIRECTORY KEY
7337 | DCS-CTR(C8), |COMPARE AT TARGET
7338 | NEXT, J/ARITH130B1 |
(4530) DCS(1.00,1.0,0.0) BM(0111.00,11.11,00.011.100...0.0,0.0,0.0...0.0000...0.0000,0...11,000...001,100,000)
7339 |
7340 | 4140: 1(FREE)
7341 | ARITH130B1:
7342 | P2-T, D_A-PLUS-B-PLUS-1, |ALU=(A-PLUS-B), CIN=(1)
7343 | D(C)_COUT15, |GET CARRYOUT
7344 | BUS-A_ASPhi(C125252), |A=(125252)
7345 | BUS-B_ASPhi(C125252), |B=(125252)
7346 | SR_A-PLUS-B-PLUS-1, |D=(052525), COUT15=(1)
7347 | NEXT, J/COMP130B1 |
(4140) DCS(0.00,0.0,0.0) BM(1100.01,11.11,01.110.110...0.1,1.0,0.0...0.0000...0.0000,0...11,000...001,100,001)
7348 |
7349 | 4141: 1(FREE)
7350 | COMP130B1:
7351 | PO, BUMP-VERIFY, |COUNT
7352 | P2-T, D_SR-XOR-CSPD(C052525), |COMPARE RECEIVED: EXPECTED
7353 | SAVE-D(C), |SAVE CARRY
7354 | NEXT, J/GOBUT130B1 |
(4141) DCS(0.00,0.0,0.1) BM(0110.10,00.00,00.000.111...0.1,0.0,0.0...0.0111...0.0000,0...11,000...001,100,010)
7355 |
7356 | 4142: 1(FREE)
```

```
7357 G0BUT130B1;
7358     SETUP, RETURN/TEST130B2,      IRETURN TO START OF NEXT SUBTEST
7359     NEXT,  CALLDINTDIR=5;          IGO PUT D -> IR, BUT(INSTRS)
(4142) DCS[0.00,0.0,0.0] BM[0100,00.10,10.11,010,111...0.0,0.0,0.0...0.0000,0...11.100,0010,111,011]
7360
7361
7362 ICHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7363 4532;
7364 TEST130B2;
7365     PO,      LOAD-ENUA(ZTARGET403),      IBIT<00> SET
7366     LOAD-ERROR(TEST130B2),              IERROR DIRECTORY KEY
7367     DCS-CTR(C3),                        ICOMPARE AT TARGET
7368     NEXT,    J/G0BUT130B2
7369 (4532) DCS[1.00,1.0,0.0] BM[1100,00.11,11.00,000,011...0.0,0.0,0.0...0.0000,0...11.000,0001,100,011]
7370
7371 4143; 1(FREE)
7372 G0BUT130B2;
7373     SETUP, RETURN/SCOPE130B,          IRETURN TO SCOPE LOOP TEST WORD
7374     NEXT,   GOTO-PAGE(7);             IOUT TABLE
7375     J/BUTD(C)A                        ID(C)H# COUT15 H IN BIT<00>
(4143) DCS[0.00,0.0,0.0] BM[0100,00.00,11.00,100,111...0.0,0.0,0.0...0.0000,0...11.100,0011,100,100]
7376
7377
7378 4144; 1(FREE)
7379 SCOPE130B;
7380     NEXT,    BUTD(SCOPE),              INO ERROR: "TEST131A1" (+1,WORDS)
7381     J/TEST131A1                       I ERROR: "ARITH130A1" (-1,WORDS)
(4144) DCS[0.00,0.1,0.0] BM[0000,00.00,00.00,000,000...0.0,0.0,0.0...0.0000,0...11.000,0011,100,001]
7382
7383
7384 1 - - - - -
7385
7386 ICHECK INTERNAL ALU CARRIES WITH: (125252)+(052525)+(0)=(177777)
7387 IALSO CHECK ALU FUNCTION "A-PLUS-B-PLUS-PS[C]", D(C)_COUT15=(0)
7388 ICIN=PS[C]=(0) FROM INITIALIZATION ROUTINE
7389 4541;
7390 TEST131A1;
7391     PO,      LOAD-ENUA(ZTARGET434),      ISETUP FOR IR=(000000)/BUTINSTRS TEST
7392     LOAD-ERROR(TEST131A1),              IERROR DIRECTORY KEY
7393     DCS-CTR(C5),                        ICOMPARE AT TARGET
7394     NEXT,    J/ARITH131A1
7395 (4541) DCS[1.00,1.0,0.0] BM[0111,00.11,11.00,011,100...0.0,0.0,0.0...0.0000,0...11.000,0011,010,000]
7396
7397
7398 4520;
7399 ARITH131A1;
7400     P2-T,    D_A-PLUS-B-PLUS-PS[C],      IALU=(A-PLUS-B), CIN=PS[C]=(0)
7401     D(C)_COUT15,                          IGET CARRYOUT
7402     BUS-A_ASPI(C125252),                  I#=(125252)
7403     BUS-B_CSPD(C052525),                  I#=(052525)
```

```
7404     SR_A-PLUS-B-PLUS-PS[C],          ID=(177777), COUT15=(0)
7405     NEXT,    J/COMPI31A1
(4520) DCS[0.00,0.0,0.0] BM[0001,10.00,11.01,110,110...0.1,1.0,0.0...0.0111,0...11.000,0001,100,101]
7406
7407 4145; 1(FREE)
7408 COMPI31A1;
7409     PO,      BUMP-VERIFY,                ICOUNT
7410     P2-T,    D_SR-XOR-CSPD(C177777),      ICOMPARE RECEIVED: EXPECTED
7411     SAVE-D(C),                          ISAVE CARRY
7412     NEXT,    J/G0BUT131A1
(4145) DCS[0.00,0.0,0.1] BM[0110,10.00,00.00,000,111...0.1,0.0,0.0...0.0101,0...11.000,0001,100,110]
7413
7414 4146; 1(FREE)
7415 G0BUT131A2;
7416     SETUP, RETURN/TEST131A2,          IRETURN TO START OF NEXT SUBTEST
7417     NEXT,    CALLDINTDIR=5;          IGO PUT D -> IR, BUT(INSTRS)
(4146) DCS[0.00,0.0,0.0] BM[0100,00.10,11.00,100,111...0.0,0.0,0.0...0.0000,0...11.100,0010,111,011]
7418
7419
7420 ICHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7421 4544;
7422 TEST131A2;
7423     PO,      LOAD-ENUA(ZTARGET401),      IBIT<01> CLEAR
7424     LOAD-ERROR(TEST131A2),              IERROR DIRECTORY KEY
7425     DCS-CTR(C3),                        ICOMPARE AT TARGET
7426     NEXT,    J/G0BUT131A2
(4544) DCS[1.00,1.0,0.0] BM[1100,00.11,11.00,000,001...0.0,0.0,0.0...0.0000,0...11.000,0001,100,111]
7427
7428
7429 4147; 1(FREE)
7430 G0BUT131A2;
7431     SETUP, RETURN/TEST131B1,          IRETURN TO START OF NEXT SUBTEST
7432     NEXT,    GOTO-PAGE(7);             IOUT TABLE
7433     J/BUTD(C)B                        ID(C)H# COUT15 H IN BIT<01>
(4147) DCS[0.00,0.0,0.0] BM[0100,00.10,11.00,110,111...0.0,0.0,0.0...0.0000,0...11.100,0011,101,000]
7434
7435
7436 1 - - - - -
7437
7438 ICHECK INTERNAL ALU CARRIES WITH: (052525)+(125252)+(0)=(177777)
7439 IALSO CHECK ALU FUNCTION "DIVIDE-STEP" = "A-PLUS-B" SINCE D(C)H=(0)
7440 IFROM ABOVE, D(C)_COUT15=(0)
7441 4546;
7442 TEST131B1;
7443     PO,      LOAD-ENUA(ZTARGET434),      ISETUP FOR IR=(000000)/BUTINSTRS TEST
7444     LOAD-ERROR(TEST131B1),              IERROR DIRECTORY KEY
7445     DCS-CTR(C5),                        ICOMPARE AT TARGET
7446     NEXT,    J/ARITH131B1
(4546) DCS[1.00,1.0,0.0] BM[0111,00.11,11.00,011,100...0.0,0.0,0.0...0.0000,0...11.000,0001,101,000]
7447
7448
7449 4150; 1(FREE)
```

```
7450 ARITH131B1:
7451 P2-T, D_DIVIDE-STEP, |ALU=(A-PLUS-B), C1N=(0)
7452 D(C)_COUT15, |GET CARRYOUT
7453 BUS-A_ASPHI(C125252), |A=(000000)
7454 BUS-B_CSPD(C125252), |B=(125252)
7455 SR_DIVIDE-STEP, |D=(177777), COUT15=(0)
7456 J/COMP131B1
(4150) DCS(0.00,0.0,0,0) BM(1000..10.00..11.01..111..110...0.1..0..0...0.0110...0..0000.0...11.000...001.101.001)
7457
7458 4151: I(FREE)
7459 COMP131B1:
7460 P0, BUMP-VERIFY, |COUNT
7461 P2-T, D_SR-XOR-CSPD(C177777), |COMPARE RECEIVED: EXPECTED
7462 SAVE-D(C), |SAVE CARRY
7463 J/GOBUT131B1
(4151) DCS(0.00,0.0,0,0) BM(0110..10.00..00.00..000..111...0.1..0..0...0.0101...0..0000.0...11.000...001.101.010)
7464
7465 4152: I(FREE)
7466 GOBUT131B1:
7467 SETUP, RETURN/TEST131B2, |RETURN TO START OF NEXT SUBTEST
7468 NEXT, CALL(DINTOIR-5) |GO PUT D -> IR, BUT(INSTRS)
(4152) DCS(0.00,0.0,0,0) BM(0100..00.11..11.01..000..111...0.0.0..0...0.0000...0..0000.0...11.100...010.111.011)
7469
7470
7471
7472 |CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7473 4750:
7474 TEST131B2:
7475 P0, LOAD-ENUA(ZTARGET413), |BIT<02> CLEAR
7476 LOAD-ERROR(TEST131B2), |ERROR DIRECTORY KEY
7477 DCS-CTR(C3), |COMPARE AT TARGET
7478 J/GOBUT131B2
(4750) DCS(1.00,1.0,0,0) BM(1100..00.11..11.00..001..011...0.0.0..0...0.0000...0..0000.0...11.000...001.101.011)
7479
7480 4153: I(FREE)
7481 GOBUT131B2:
7482 SETUP, RETURN/SCOPE131B, |RETURN TO SCOPE LOOP TEST WORD
7483 NEXT, GOTO-PAGE(7), |BUT TABLE
7484 J/BUTD(C1C) |D(C)H= COUT15 H IN BIT<02>
(4153) DCS(0.00,0.0,0,0) BM(0100..00.00..11.01..100..111...0.0.0..0...0.0000...0..0000.0...11.100...011.001.000)
7485
7486
7487
7488 4154: I(FREE)
7489 SCOPE131B:
7490 P3, CSPD(05)_EMIT, EMIT/170360, |CONSTANT FOR USE BELOW
7491 NEXT, BUTD(SCOPE), |NO ERROR: "TEST132A1" (+1. WORDS)
7492 J/TEST132A1 | ERROR: "ARITH131A1" (-1. WORDS)
(4154) DCS(0.00,0.1,0,0) BM(1111..10.00..00.11..110..000...0.0.0..0...0.1010...1..0000.0...11.000...101.010.001)
7493
7494
7495
```

```
7496 | - - - - -
7497
7498 |CHECK INTERNAL ALU CARRIES WITH: (125252)-(125252)-(0)=(000000)
7499 |ALSO CHECK ALU FUNCTION "A-MINUS-B", D(C)_COUT15=(1)
7500
7501 4521:
7502 TEST132A1:
7503 P0, LOAD-ENUA(ZTARGET434), |SETUP FOR IR=(000000)/BUTINSTRS TEST
7504 LOAD-ERROR(TEST132A1), |ERROR DIRECTORY KEY
7505 DCS-CTR(C7), |COMPARE AT TARGET
7506 J/ARITH132A1
(4521) DCS(1.00,1.0,0,0) BM(1000..00.11..11.00..011..100...0.0.0..0...0.0000...0..0000.0...11.000...101.010.010)
7507
7508 4522:
7509 ARITH132A1:
7510 P3-T, D_A-MINUS-B, |ALU=(A-MINUS-B-MINUS-1), C1N=(1)
7511 D(C)_COUT15, |GET CARRYOUT
7512 BUS-A_ASPHI(C125252), |A=(125252)
7513 BUS-B_BSPHI(C125252), |B=(125252)
7514 |D=(000000), COUT15=(1)
7515 J/GOBUT132A1
(4522) DCS(0.00,0.0,0,0) BM(1101..01.11..11.01..110..110...1.1.0..0...0.0000...0..0000.0...11.000...001.101.101)
7516
7517 4155: I(FREE)
7518 GOBUT132A1:
7519 P0, BUMP-VERIFY, |COUNT
7520 SETUP, RETURN/TEST132A2, |RETURN TO START OF NEXT SUBTEST
7521 NEXT, CALL(DINTOIR-5) |GO PUT D -> IR, BUT(INSTRS)
(4155) DCS(0.00,0.0,0,0) BM(0100..00.10..10.01..110..111...0.0.0..0...0.0000...0..0000.0...11.100...010.111.011)
7522
7523
7524
7525 |CHECK THAT CARRYOUT OF BIT15 (COUT15) WAS CORRECT
7526 4516:
7527 TEST132A2:
7528 P0, LOAD-ENUA(ZTARGET403), |BIT<01> SET
7529 LOAD-ERROR(TEST132A2), |ERROR DIRECTORY KEY
7530 DCS-CTR(C3), |COMPARE AT TARGET
7531 J/GOBUT132A2
(4516) DCS(1.00,1.0,0,0) BM(1100..00.11..11.00..000..011...0.0.0..0...0.0000...0..0000.0...11.000...001.101.110)
7532
7533 4156: I(FREE)
7534 GOBUT132A2:
7535 SETUP, RETURN/TEST132B1, |RETURN TO START OF NEXT SUBTEST
7536 NEXT, GOTO-PAGE(7), |BUT TABLE
7537 J/BUTD(C1R) |D(C)H= COUT15 H IN BIT<01>
(4156) DCS(0.00,0.0,0,0) BM(0100..00.10..10.01..111..111...0.0.0..0...0.0000...0..0000.0...11.100...011.101.000)
7538
7539
7540
7541 | - - - - -
7542
```

```

7543 JCHECK INTERNAL ALU CARRIES WITH: (052525)-(052525)=0=(000000)
7544 IALSO CHECK ALU FUNCTION "DIVIDE-STEP" = "A-MINUS-B" SINCE D(C)=11
7545 IFROM ABOVE, D(C) COUNT15=(1)
7546 4517:
7547 TEST132B1:
7548 PO, LOAD-ENUA(ZTARGET434), ISETUP FOR IR=(000000)/BUTINSTR5 TEST
7549 LOAD-ERROR(TEST132B1), IERROR DIRECTORY KEY
7550 DCS-CTR(C7.), ICOMPARE AT TARGET
7551 NEXT, J/ARITH132B1
(4517) DCS[1.00,1.0,0,0] BM[1000..00,11..11,00..011..100..0,0,0,0,0..0,0000...0..0000,0...11,000...001,101,111]
7552
7553 4157: 1(FREE)
7554 ARITH132B1:
7555 P3-T, D-DIVIDE-STEP, IALU=(A-MINUS-B-MINUS-1), CTN=(1)
7556 D(C) COUNT15, IGET CARRYOUT
7557 BUS-A ASPHI(C052525), I= (052525)
7558 BUS-B ASPHI(C052525), I= (052525)
7559 I= (000000), COUNT15=(1)
7560 NEXT, J/GOBUT132B1
(4157) DCS[0.00,0,0,0,0] BM[1000..01,11..11,01..111..110...1,1,0,0,0..0,0000...0..0000,0...11,000...001,110,000]
7561
7562 4160: 1(PREF)
7563 GORUT132B1:
7564 PO, BUMP-VERIFY, ICOUNT
7565 SETUP, RETURN/TEST132B2, IRETURN TO START OF NEXT SUBTEST
7566 NEXT, CALL(DINTOIR=5) IGO PUT D -> IR, BUT(INSTR5)
(4160) DCS[0.00,0,0,0,1] BM[0100..00,10..01,11..010..111...0,0,0,0,0..0,0000...0..0000,0...11,100...010,111,011]
7567
7568 JCHECK THAT CARRYOUT OF BIT15 (COUNT15) WAS CORRECT
7569 4472:
7570 TEST132B2:
7571 PO, LOAD-ENUA(ZTARGET417), IBIT<02> SET
7572 LOAD-ERROR(TEST132B2), IERROR DIRECTORY KEY
7573 DCS-CTR(C3.), ICOMPARE AT TARGET
7574 NEXT, J/GOBUT132B2
(4472) DCS[1.00,1.0,0,0,0] BM[1100..00,11..11,00..001..111...0,0,0,0,0..0,0000...0..0000,0...11,000...001,110,001]
7575
7576 4161: 1(PREF)
7577 GORUT132B2:
7578 SETUP, RETURN/SCOPE132B, IRETURN TO SCOPE LOOP TEST WORD
7579 GOTO-PAGE(7), IBUT TABLE
7580 NEXT, J/BUTD(C) ID(C)=COUNT15 H IN BIT <02>
(4161) DCS[0.00,0,0,0,0] BM[0100..00,00..11,10..010..111...0,0,0,0,0..0,0000...0..0000,0...11,100...011,001,000]
7581
7582 4162: 1(FREE)
7583 SCOPE132B:
7584 P3, CNPD[06]_EMIT, EMIT/007417, ICONSTANT FOR USE BELOW
7585 NEXT, BUTD(SCOPE), INO ERROR: "TEST133A1" (+1, WORDS)
7586
7587
7588
7589

```

```

7590      J/TEST133A1                      I ERROR: "ARITH133A1" (-9, WORDS)
(4162) DCS[0.00,0.1,0.0] BM[0000..10.11..11.00..001..111...0.0,0.0..0.0..0.1001...1..0000.0...11.000...101.010,011]

7591
7592
7593
7594
7595      I -----
7596
7597      ICHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
7598      I(103607)-PLUS-(103607)-PLUS-(1)=(007417), COUTIS=(1)
7599      4523:
7600      TEST133A1:
7601      PO,          LOAD=ENUA(ZTARGET434),                      ISETUP FOR IR=(000000)/BUTINSTRS COMPARE
7602      LOAD=ERROR(TEST133A1),                      IERROR DIRECTORY KEY
7603      DCS=CTR(C10,),                      ICOMPARE AT TARGET
7604      NEXT,        J/OPB133A1                      I
(4523) DCS[1.00,1.0,0.0] BM[0101..00.11..11.00..011..100...0.0,0.0..0.0..0.0000...0..0000.0...11.000...101.001,000]

7605      4510:
7606      OPB133A1:
7607      P3,          CSPD[16]_EMIT,                      IA,B-SIDE OPERANDS:
7608      EMIT/103607,                      I"1000 0111 1000 0111"
7609      NEXT,        J/DOPA133A1                      I
7610      DCS[0.00,0.0,0.0] BM[1000..10.01..11.10..000..111...0.0,0.0..0.0..0.0001...1..0000.0...11.000...001.110,011]

7611      4163: I(FREE)
7612      DOPA133A1:
7613      P2=T,        D_CSPD[D16], D[C]_ALU15,                      IOP-A INTO D, D[C]_1
7614      SF_CSPD[D16],                      IOP-A INTO SR TOO
7615      NEXT,        J/COTEST133A1                      I(OPA=OPB)
7616      DCS[0.00,0.0,0.0] BM[1010..10.00..00.00..000..100...0.1,1.0..0.0..0.0001...0..0000.0...11.000...001.110,100]

(4163) DCS[0.00,0.0,0.0] BM[1010..10.00..00.00..000..100...0.1,1.0..0.0..0.0001...0..0000.0...11.000...001.110,100]

7617      4164: I(FREE)
7618      COTEST133A1:
7619      PO,          BUMP=VERIFY,                      ICOUNT
7620      SETUP,       RETURN/TEST133A2,                      IEXEC SUBR WHICH:
7621      I(1) D_OPA=PLUS-OPB-PLUS-1
7622      I(2) D_007417=XOR-D (EQUAL?)
7623      I(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
7624      NEXT,        CALL[ALUCARRY1]
(4164) DCS[0.00,0.0,0.1] BM[0100..00.10..10.00..000..111...0.0,0.0..0.0..0.0000...0..0000.0...11.100...000.000,111]

7625
7626
7627
7628      ICHECK THAT CARRYOUT COUTIS ABOVE GENERATED CORRECTLY AS A (1)
7629      4500:
7630      TEST133A2:
7631      PO,          LOAD=ENUA(ZTARGET403),                      IBIT<00>SET
7632      LOAD=ERROR(TEST133A2),                      IERROR DIRECTORY KEY
7633      DCS=CTR(C3,),                      ICOMPARE AT TARGET
7634      BUMP=VERIFY,                      ICOUNT
7635      NEXT,        J/GOBUT133A2                      I
(4500) DCS[1.00,1.0,0.1] BM[1100..00.11..11.00..000..011...0.0,0.0..0.0..0.0000...0..0000.0...11.000...001.110,101]

7636

```

```
7637 4165: 1(FREE)
7638 GORUT133A2:
7639 SETUP, RETURN/TEST133B1, RETURN TO START OF NEXT SUBTEST
7640 NEXT, GOTO-PAGE(7), BUT TABLE
7641 J/BUTD(C)A IDICIN = COUNTS IN BIT<00>
(4165) DCS[0.00.0.0.0.0] BM[0100.00.10.10.00.001.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.011.100.100]
7642
7643
7644
7645
7646
7647 1 - - - - -
7648
7649 ICHECK CARRY PROPOGATE/GENERATE LOGIC WITH COMPLEMENT OF ABOVE:
7650 1(074170)-PLUS-(074170)-PLUS-(0)=(170360), COUNTS=0)
7651 4501:
7652 TEST133B1:
7653 PO, LOAD-ENUA(ZTARGET434), SETUP FOR IR=(000000)/BUTINSTR5 COMPARE
7654 LOAD-ERROR(TEST133B1), ERROR DIRECTORY KEY
7655 DCS-CTR(C.), COMPARE AT TARGET
7656 NEXT, J/OPA133B1
(4501) DCS[1.00.1.0.0.0] BM[0110.00.11.11.00.011.100.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.001.110.110]
7657
7658 4166: 1(FREE)
7659 OPA133B1:
7660 P2-T, D_NOT-CSPD(D16),
7661 D[C]_ALU15,
7662 SR_NOT-CSPD(D16),
7663
7664
7665
7666 NEXT, J/GOTEST133B1
(4166) DCS[0.00.0.0.0.0] BM[0111.10.00.11.01.101.100.0.0.1.1.0.0.0.0.0001.0.0.0000.0.11.100.001.110.111]
7667
7668 4167: 1(FREE)
7669 GOTEST133B1:
7670 SETUP, RETURN/TEST133B2, IEXEC SUBR WHICH:
7671
7672
7673 NEXT, CALL[ALUCARRY2]
(4167) DCS[0.00.0.0.0.0] BM[0100.00.10.10.00.010.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.000.010.000]
7674
7675
7676
7677 ICHECK THAT CARRYOUT COUNTS ABOVE GENERATED CORRECTLY AS A (0)
7678 4502:
7679 TEST133B2:
7680 PO, LOAD-ENUA(ZTARGET402), BIT<00> CLEAR
7681 LOAD-ERROR(TEST133B2), ERROR DIRECTORY KEY
7682 NEXT, J/GORUT133B2
(4502) DCS[1.00.0.0.0.0] BM[0000.00.11.11.00.000.010.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.001.111.000]
7683
```

```
7684 4170: 1(FREE)
7685 GORUT133B2:
7686 SETUP, RETURN/SCOPE133B, RETURN TO SCOPE LOOP TEST WORD
7687 NEXT, GOTO-PAGE(7), BUT TABLE
7688 J/BUTD(C)A IDICIN = COUNTS IN BIT<00>
(4170) DCS[0.00.0.0.0.0] BM[0100.00.00.11.11.001.111.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.011.100.100]
7689
7690
7691
7692 4171: 1(FREE)
7693 SCOPE133B:
7694 PO, BUMP-VERIFY, ICOUNT
7695 NFXT, BUTD(SCOPE), NO ERROR: "TEST134A1" (+1, WORDS)
7696 J/TEST134A1 ERROR: "OPA133A1" (-1, WORDS)
(4171) DCS[0.00.0.1.0.1] BM[0000.00.00.00.00.000.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.011.101.001]
7697
7698
7699
7700 1 - - - - -
7701
7702 ICHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
7703 1(045513)-PLUS-(141703)-PLUS-(1)=(007417), COUNTS=1)
7704 4511:
7705 TEST134A1:
7706 PO, LOAD-ENUA(ZTARGET434), SETUP FOR IR=(000000)/BUTINSTR5 COMPARE
7707 LOAD-ERROR(TEST134A1), ERROR DIRECTORY KEY
7708 DCS-CTR(C1.), COMPARE AT TARGET
7709 NEXT, J/OPA134A1
(4511) DCS[1.00.1.0.0.0] BM[0100.00.11.11.00.011.100.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.011.101.010]
7710
7711 4512:
7712 OPA134A1:
7713 PO, BUMP-VERIFY, ICOUNT
7714 P3, CSPD[17]_EMIT, IA-SIDE OPERAND:
7715 EMIT/045513, I"0100 1011 0100 1011"
7716 NEXT, J/OPA134A1
(4512) DCS[0.00.0.0.0.1] BM[0100.10.10.11.01.001.011.0.0.0.0.0.0.0.0000.1.0.0000.0.11.100.001.111.010]
7717
7718 4172: 1(FREE)
7719 OPA134A1:
7720 P3, CSPD[16]_EMIT, IB-SIDE OPERAND:
7721 EMIT/141703, I"1100 0011 1100 0011"
7722 NFXT, J/DOPA134A1
(4172) DCS[0.00.0.0.0.0] BM[1100.10.00.11.11.000.011.0.0.0.0.0.0.0.0001.1.0.0000.0.11.100.001.111.011]
7723
7724 4173: 1(FREE)
7725 OPA134A1:
7726 P2-T, D_CSPD(D17), D[C]_ALU00,
7727 SR_CSPD(D17),
7728 NFXT, J/GOTEST134A1
(4173) DCS[0.00.0.0.0.0] BM[1010.10.00.00.00.000.010.0.0.1.1.0.0.0.0000.0.0.0000.0.11.100.001.111.100]
7729
7730 4174: 1(FREE)
```

```

7731 G0TEST134A1;
7732 PO, BUMP-VERIFY, ICOUNT
7733 SETUP, RETURN/TEST134A2, IEXEC SUBR WHICH;
7734 (1) D_00A-PLUS-OPB-PLUS-1
7735 (2) D_007417-XOR-D (EQUAL?)
7736 (3) J/BUTINSTR5/(000000) (CHECK ANSWER)
7737
7738 NEXT, CALL[ALUCARRY1]
7739 (4174) DCS[0.00.0.0.0.1] BM[0100..00.10..10.00..011..111..0.0.0..0.0...0.0000...0..0000.0...11.100...000.000.111]
7740
7741 ICHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (1)
7742 4503;
7743 TEST134A2;
7744 PO, LOAD-ENUA(ZTARGET403), IBIT<01> SET
7745 LOAD-ERROR(TEST134A2), IERROR DIRECTORY KEY
7746 DCS-CTR(C3.), ICOMPARE AT TARGET
7747 BUMP-VERIFY, ICOUNT
7748 J/GOBUT134A2
7749 (4503) DCS[1.00.1.0.0.1] BM[1100..00.11..11.00..000.0011...0.0.0..0.0...0.0000...0..0000.0...11.000...001.111.101]
7750
7751 4175; I(FREE)
7752 GOBUT134A2;
7753 SETUP, RETURN/TEST134B1, IRETURN TO START OF NEXT SUBTEST
7754 NEXT, COTO-PAGE(7), IBUT TABLE
7755 J/BUTD[C15] ID[C] 9 COUT15 IN BIT<01>
7756 (4175) DCS[0.00.0.0.0.0] BM[0100..00.10..10.00..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.101.000]
7757
7758
7759
7760
7761
7762
7763
7764
7765
7766
7767
7768
7769
7770
7771
7772
7773
7774
7775
7776
7777
7778
7779
7780
7781
7782
7783
7784
7785
7786
7787
7788
7789
7790
7791
7792
7793
7794
7795
7796
7797
7798
7799
7800
7801
7802
7803
7804
7805
7806
7807
7808
7809
7810
7811
7812
7813
7814
7815
7816
7817
7818
7819
7820
7821
7822
7823
7824
7825
7826
7827
7828
7829
7830
7831
7832
7833
7834
7835
7836
7837
7838
7839
7840
7841
7842
7843
7844
7845
7846
7847
7848
7849
7850
7851
7852
7853
7854
7855
7856
7857
7858
7859
7860
7861
7862
7863
7864
7865
7866
7867
7868
7869
7870
7871
7872
7873
7874
7875
7876
7877
7878
7879
7880
7881
7882
7883
7884
7885
7886
7887
7888
7889
7890
7891
7892
7893
7894
7895
7896
7897
7898
7899
7900
7901
7902
7903
7904
7905
7906
7907
7908
7909
7910
7911
7912
7913
7914
7915
7916
7917
7918
7919
7920
7921
7922
7923
7924
7925
7926
7927
7928
7929
7930
7931
7932
7933
7934
7935
7936
7937
7938
7939
7940
7941
7942
7943
7944
7945
7946
7947
7948
7949
7950
7951
7952
7953
7954
7955
7956
7957
7958
7959
7960
7961
7962
7963
7964
7965
7966
7967
7968
7969
7970
7971
7972
7973
7974
7975
7976
7977
7978
7979
7980
7981
7982
7983
7984
7985
7986
7987
7988
7989
7990
7991
7992
7993
7994
7995
7996
7997
7998
7999
8000
8001
8002
8003
8004
8005
8006
8007
8008
8009
8010
8011
8012
8013
8014
8015
8016
8017
8018
8019
8020
8021
8022
8023
8024
8025
8026
8027
8028
8029
8030
8031
8032
8033
8034
8035
8036
8037
8038
8039
8040
8041
8042
8043
8044
8045
8046
8047
8048
8049
8050
8051
8052
8053
8054
8055
8056
8057
8058
8059
8060
8061
8062
8063
8064
8065
8066
8067
8068
8069
8070
8071
8072
8073
8074
8075
8076
8077
8078
8079
8080
8081
8082
8083
8084
8085
8086
8087
8088
8089
8090
8091
8092
8093
8094
8095
8096
8097
8098
8099
8100
8101
8102
8103
8104
8105
8106
8107
8108
8109
8110
8111
8112
8113
8114
8115
8116
8117
8118
8119
8120
8121
8122
8123
8124
8125
8126
8127
8128
8129
8130
8131
8132
8133
8134
8135
8136
8137
8138
8139
8140
8141
8142
8143
8144
8145
8146
8147
8148
8149
8150
8151
8152
8153
8154
8155
8156
8157
8158
8159
8160
8161
8162
8163
8164
8165
8166
8167
8168
8169
8170
8171
8172
8173
8174
8175
8176
8177
8178
8179
8180
8181
8182
8183
8184
8185
8186
8187
8188
8189
8190
8191
8192
8193
8194
8195
8196
8197
8198
8199
8200
8201
8202
8203
8204
8205
8206
8207
8208
8209
8210
8211
8212
8213
8214
8215
8216
8217
8218
8219
8220
8221
8222
8223
8224
8225
8226
8227
8228
8229
8230
8231
8232
8233
8234
8235
8236
8237
8238
8239
8240
8241
8242
8243
8244
8245
8246
8247
8248
8249
8250
8251
8252
8253
8254
8255
8256
8257
8258
8259
8260
8261
8262
8263
8264
8265
8266
8267
8268
8269
8270
8271
8272
8273
8274
8275
8276
8277
8278
8279
8280
8281
8282
8283
8284
8285
8286
8287
8288
8289
8290
8291
8292
8293
8294
8295
8296
8297
8298
8299
8300
8301
8302
8303
8304
8305
8306
8307
8308
8309
8310
8311
8312
8313
8314
8315
8316
8317
8318
8319
8320
8321
8322
8323
8324
8325
8326
8327
8328
8329
8330
8331
8332
8333
8334
8335
8336
8337
8338
8339
8340
8341
8342
8343
8344
8345
8346
8347
8348
8349
8350
8351
8352
8353
8354
8355
8356
8357
8358
8359
8360
8361
8362
8363
8364
8365
8366
8367
8368
8369
8370
8371
8372
8373
8374
8375
8376
8377
8378
8379
8380
8381
8382
8383
8384
8385
8386
8387
8388
8389
8390
8391
8392
8393
8394
8395
8396
8397
8398
8399
8400
8401
8402
8403
8404
8405
8406
8407
8408
8409
8410
8411
8412
8413
8414
8415
8416
8417
8418
8419
8420
8421
8422
8423
8424
8425
8426
8427
8428
8429
8430
8431
8432
8433
8434
8435
8436
8437
8438
```

```

(4176) DCS[0.00,0.0,0.0] BM[0111..10.00..11.01..101..010...0.1.1..0.0..0.0000...0..0000.0...11.000...001.111.111]
7779
7780 4177: 1(FREE)
7781 GOTO134B1:
7782 SETUP, RETURN/TEST134B2, IEXEC SUBR WHICH:
7783 I(1) D,(NOT-OPA)=PLUS-(NOT-OPB)
7784 I(2) D,(70360-XOR-D (EQUAL?))
7785 NEXT, CALL[ALUCARRY2] I(3) J/BUTINSTR5/(000000) (CHECK ANSWER)
(4177) DCS[0.00,0.0,0.0] BM[0100..00.10..10.00..101..111...0.0.0..0..0..0.0000...0..0000.0...11.100...000.010.000]
7786
7787
7788
7789 I/CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (0)
7790 4*05:
7791 TEST134B2:
7792 PO, LOAD-ENUA(ZTARGET401), I/BIT<01> CLEAR
7793 LOAD-ERROR(TEST134B2), I/ERROR DIRECTORY KEY
7794 DCS-CTR(C1.), I/COMPARE AT TARGET
7795 NEXT, J/GOTO134B2 I
(4505) DCS[1.00,1.0,0.0] BM[1100..00.11..11.00..000...001...0.0.0..0..0..0.0000...0..0000.0...11.000...010.000.000]
7796
7797 4200: 1(FREE)
7798 GOTO134B2:
7799 SETUP, RETURN/SCOPE134B, I/RETURN TO SCOPE LOOP TEST WORD
7800 NEXT, GOTO-PAGE(7), I/BUT TABLE
7801 J/BUTDI[C]B I/DCIH = COUT15 H IN BIT<01>
(4200) DCS[0.00,0.0,0.0] BM[0100..00.01..00.00..001...111...0.0.0..0..0..0.0000...0..0000.0...11.100...011.101.000]
7802
7803
7804
7805 4201: 1(FREE)
7806 SCOPE134B:
7807 PO, BUMP-VERIFY, I/COUNT
7808 NEXT, BUTDI[SCOPE], I/NO ERROR: "TEST135A1" (+1, WORDS)
7809 J/TEST135A1 I/ERROR: "OPAI34A1" (-11, WORDS)
(4201) DCS[0.00,0.1,0.0] BM[0000..00.00..00.00..000...000...0.0.0..0..0..0.0000...0..0000.0...11.000...101.001.011]
7810
7811
7812
7813
7814
7815 I/CHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
7816 I(122645)-PLUS-(064551)-PLUS-(1)=(007417), COUT15=(1)
7817 4513:
7818 TEST135A1:
7819 PO, LOAD-ENUA(ZTARGET434), I/SETUP FOR IR=(000000)/BUTINSTR5 COMPARE
7820
7821 LOAD-ERROR(TEST135A1), I/ERROR DIRECTORY KEY
7822 DCS-CTR(C11.), I/COMPARE AT TARGET
7823 NEXT, J/OPA135A1 I
(4513) DCS[1.00,1.0,0.0] BM[0100..00.11..11.00..011...100...0.0.0..0..0..0.0000...0..0000.0...11.000...101.001.100]
7823
7824 4514:

```

```

7825 OPA135A1:
7826 PO, BUMP-VERIFY, ICOUNT
7827 P3, CSPD[17]_EMIT, IASIDE OPERAND:
7828 EMIT/122645, I"1010 0101 1010 0101"
7829 NEXT, J/OPB135A1
(4514) DCS[0.00,0.0,0.0,1] BM[1010..10,01..01,10..100..101...0,0,0,0...0,0000...1..0000,0...11,000...010,000,010]

7830
7831 4202: I(FREE)
7832 OPA135A1:
7833 P3, CSPD[16]_EMIT, IB-SIDE OPERAND:
7834 EMIT/064551, I"0110 1001 0110 1001"
7835 NEXT, J/DOPA135A1
(4202) DCS[0.00,0.0,0.0,0] BM[1010..10,10..01,01..101..001...0,0,0,0...0,0001...1..0000,0...11,000...010,000,011]

7836
7837 4203: I(FREE)
7838 DOPA135A1:
7839 P2-T, D_CSPD[D17], D[C]_ALU15, IOP-A INTO D, D[C]_1
7840 SR_CSPD[D17], IOP-A INTO SR TOO
7841 NEXT, J/GOTEST135A1
(4203) DCS[0.00,0.0,0.0,0] BM[1010..10,00..00,00..000..100...0,1,1..0,0...0,0000...0..0000,0...11,000...010,000,100]

7842
7843 4204: I(FREE)
7844 GOTEST135A1:
7845 PO, BUMP-VERIFY, ICOUNT
7846 SETUP, RETURN/TEST135A2, IEXEC SUBR WHICH:
7847 I(1) D_OPA-PLUS-OPS-PLUS-1
7848 I(2) D_007417-XOR-D (EQUAL?)
7849 NEXT, CALL[ALUCARRY1] I(3) J/BUTINSTR5/(000000) (CHECK ANSWER)
(4204) DCS[0.00,0.0,0.0,1] BM[10100..00,10..10,00..110..111...0,0,0,0...0,0000...0..0000,0...11,100...000,000,111]

7850
7851
7852
7853 ICHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (1)
7854 4506:
7855 TEST135A2:
7856 PO, LOAD-ENVA(ZTARGET417), IBIT<02> SET
7857 LOAD-ERROR(TEST135A2), IERROR DIRECTORY KEY
7858 DCS-CTR(C3), ICOMPARE AT TARGET
7859 BUMP-VERIFY, ICOUNT
7860 NEXT, J/GOBUT135A2
(4506) DCS[1.00,1,0,0,1] BM[1100..00,11..11,00..001..111...0,0,0,0...0,0000...0..0000,0...11,000...010,000,101]

7861
7862 4205: I(FREE)
7863 GOBUT135A2:
7864 SETUP, RETURN/TEST135B1, IRETURN TO START OF NEXT SUBTEST
7865 NEXT, GOTO-PAGE(7), IBUT TABLE
7866 J/BUTDICIC ID[C]H = COUT15 IN BIT<02>
(4205) DCS[0.00,0.0,0.0,0] BM[10100..00,10..10,00..111..111...0,0,0,0...0,0000...0..0000,0...11,100...011,001,000]

7867
7868
7869
7870

```

```

7871
7872 | - - - - -
7873
7874 ICHECK CARRY PROPAGATE/GENERATE LOGIC WITH COMPLEMENT OF ABOVE:
7875 I(055132)-PLUS-(113226)-PLUS-(0)=(170360), COUT15=(0)
7876 4507:
7877 TEST135B1:
7878 PO, LOAD-ENVA(ZTARGET434), ISETUP FOR IR=(000000)/BUTINSTR5 COMPARE
7879 LOAD-ERROR(TEST135B1), IERROR DIRECTORY KEY
7880 DCS-CTR(C9), ICOMPARE AT TARGET
7881 NEXT, J/OPA135B1
(4507) DCS[1.00,1,0,0,0] BM[10110..00,11..11,00..011..100...0,0,0,0...0,0000...0..0000,0...11,000...010,000,110]

7882
7883 4206: I(FREE)
7884 OPA135B1:
7885 P2-T, D_NOT-CSPD[D17], I
7886 D[C]_ALU15, IOP-A INTO D, D[C]_(0)
7887 SR_NOT-CSPD[D17], IOP-A INTO SR TOO
7888 IASIDE OPERAND:
7889 I"0101 1010 0101 1010"
7890 IB-SIDE OPERAND WILL BE:
7891 I"1001 0110 1001 0110"
(4206) DCS[0.00,0.0,0.0,0] BM[10111..10,00..11,01..101..100...0,1,1..0,0...0,0000...0..0000,0...11,000...010,000,111]

7892
7893 4207: I(FREE)
7894 GOTEST135B1:
7895 SETUP, RETURN/TEST135B2, IEXEC SUBR WHICH:
7896 I(1) D_(NOT-OPA)-PLUS-(NOT-OPS)
7897 I(2) D_170360-XOR-D (EQUAL?)
7898 NEXT, CALL[ALUCARRY2] I(3) J/BUTINSTR5/(000000) (CHECK ANSWER)
(4207) DCS[0.00,0.0,0.0,0] BM[10100..00,10..01,11..111..111...0,0,0,0...0,0000...0..0000,0...11,100...000,010,000]

7899
7900
7901 ICHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (0)
7902 4477:
7903 TEST135B2:
7904 PO, LOAD-ENVA(ZTARGET413), IBIT<02> CLEAR
7905 LOAD-ERROR(TEST135B2), IERROR DIRECTORY KEY
7906 DCS-CTR(C3), ICOMPARE AT TARGET
7907 NEXT, J/GOBUT135B2
(4477) DCS[1.00,1,0,0,0] BM[1100..00,11..11,00..001..011...0,0,0,0...0,0000...0..0000,0...11,000...010,001,000]

7908
7909 4210: I(FREE)
7910 GOBUT135B2:
7911 SETUP, RETURN/SCOPE135B, IRETURN TO SCOPE LOOP TEST WORD
7912
7913 NEXT, GOTO-PAGE(7), IBUT TABLE
7914 J/BUTDICIC ID[C]H = COUT15 H IN BIT<02>
(4210) DCS[0.00,0.0,0.0,0] BM[10100..00,01..00,01..001..111...0,0,0,0...0,0000...0..0000,0...11,100...011,001,000]

7915
7916
7917

```

```
7918 4211: 1(FREE)
7919 SCOPE135A:
7920 PO, BUMP-VERIFY, |COUNT
7921 NEXT, BUTD[SCOPE1], |NO ERROR: "TEST136A1" (+1, WORDS)
7922 J/TEST136A1 | ERROR: "OPA136A1" (-1, WORDS)
(4711) DCS[0.00,0.1,0.0.1] BM[0000..00,00..00,00..0000..0000..0.0,0.0,0.0,0.0,0.0000..0.0000,0...11,000...101,001,101]
7923
7924
7925 1 - - - - -
7926
7927
7928 |CHECK CARRY PROPOGATE/GENERATE LOGIC WITH:
7929 I(055132)-PLUS-(132264)-PLUS-(1)(007417), COUT15=(1)
7930 4515:
7931 TEST136A1:
7932 PO, LOAD-ENUA(ZTARGET434), |SETUP FOR IR=(000000)/BUTINSTRS COMPARE
7933 LOAD-ERROR(TEST136A1), |ERROR DIRECTORY KEY
7934 DCS-CTR(C10,), |COMPARE AT TARGET
7935 NEXT, J/OPB136A1 |
(4515) DCS[1.00,1.0,0.0.0] BM[0101..00,11..11,00..011..100...0.0,0.0,0.0,0.0,0.0000..0.0000,0...11,000...101,100,010]
7936
7937 4542:
7938 OPA136A1:
7939 P3, CSPD[16]_EMIT, |B-SIDE OPERAND:
7940 EMIT/132264, |"1011 0100 1011 0100"
7941 NEXT, J/DOPA136A1 |
(4542) DCS[0.00,0.0,0.0.0] BM[1011..10,01..00,10..110..100...0.0,0.0,0.0,0.0,0.0001...1..0000,0...11,000...010,001,010]
7942
7943 4212: 1(FREE)
7944 OPA136A1:
7945 P2-T, D_NOT-CSPD(D17), D[C]_COUT07, |OP-A INTO D, D[C]_1
7946 SP_NOT-CSPD(D17), |OP-A INTO SR TOO
7947 NEXT, J/GOTEST136A1 |"0101 1010 0101 1010"
(4212) DCS[0.00,0.0,0.0.0] BM[0111..10,00..11,01..101..101...0.1,1.0,0.0,0.0,0.0000..0.0000,0...11,000...010,001,011]
7948
7949 4213: 1(FREE)
7950 GOTEST136A1:
7951 PO, BUMP-VERIFY, |COUNT
7952 SETUP, RETURN/TEST136A2, |EXEC SUBR WHICH:
7953 |{(1) D_OPA-PLUS-OPB-PLUS-1
7954 |{(2) D_007417-XOR-D EQUAL?}
7955 NEXT, CALL[ALUCARRY1] |{(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
(4213) DCS[0.00,0.0,0.0.1] BM[0100..00,10..01,11..110..111...0.0,0.0,0.0,0.0,0.0000..0.0000,0...11,100...000,000,111]
7956
7957
7958
7959 |CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (1)
7960 4476:
7961 TEST136A2:
7962 PO, LOAD-ENUA(ZTARGET403), |BIT<00> SET
7963 LOAD-ERROR(TEST136A2), |ERROR DIRECTORY KEY
7964 DCS-CTR(C3,), |COMPARE AT TARGET
7965 BUMP-VERIFY, |COUNT
```

```
7966 NEXT, J/GOBUT136A2
(4476) DCS[1.00,1.0,0.0.1] BM[1100..00,11..11,00..0000..011...0.0,0.0,0.0,0.0,0.0000..0.0000,0...11,000...010,001,100]
7967
7968 4214: 1(FREE)
7969 GOBUT136A2:
7970 SETUP, RETURN/TEST136B1, |RETURN TO START OF NEXT SUBTEST
7971 NEXT, COTO-PAGE(7), |BUT TABLE
7972 J/BUTD[C1] |ID[C]H = COUT15 IN BIT<00>
(4214) DCS[0.00,0.0,0.0.0] BM[0100..00,10..01,11..101..111...0.0,0.0,0.0,0.0,0.0000..0.0000,0...11,100...011,100,100]
7973
7974
7975
7976
7977
7978 1 - - - - -
7979
7980 |CHECK CARRY PROPOGATE/GENERATE LOGIC WITH COMPLEMENT OF ABOVE:
7981 I(122645)-PLUS-(132264)-PLUS-(0)=(170360), COUT15=(0)
7982 4475:
7983 TEST136B1:
7984 PO, LOAD-ENUA(ZTARGET434), |SETUP FOR IR=(000000)/BUTINSTRS COMPARE
7985 LOAD-ERROR(TEST136B1), |ERROR DIRECTORY KEY
7986 DCS-CTR(C9,), |COMPARE AT TARGET
7987 NEXT, J/OPA136B1 |
(4475) DCS[1.00,1.0,0.0.0] BM[0110..00,11..11,00..011..100...0.0,0.0,0.0,0.0,0.0000..0.0000,0...11,000...010,001,101]
7988
7989 4215: 1(FREE)
7990 OPA136B1:
7991 P2-T, D_CSPD(D17), |
7992 D[C]_0, |OP-A INTO D, D[C]_0
7993 SR_CSPD(D17), |OP-A INTO SR TOO
7994 |A-SIDE OPERAND:
7995 |"1010 0101 1010 0101"
7996 |B-SIDE OPERAND WILL BE:
7997 |"0100 1011 0100 1011"
(4215) DCS[0.00,0.0,0.0.0] BM[1010..10,00..00,00..0000..0000...0.1,1.0,0.0,0.0,0.0000..0.0000,0...11,000...010,001,110]
7998
7999 4216: 1(FREE)
8000 GOTEST136B1:
8001 SETUP, RETURN/TEST136B2, |EXEC SUBR WHICH:
8002 |{(1) D_(NOT-OPA)-PLUS-(NOT-OPB)
8003 |{(2) D_170360-XOR-D (EQUAL?)
8004 NEXT, CALL[ALUCARRY2] |{(3) J/BUTINSTRS/(000000) (CHECK ANSWER)
(4216) DCS[0.00,0.0,0.0.0] BM[0100..00,10..01,11..011..111...0.0,0.0,0.0,0.0,0.0000..0.0000,0...11,100...000,010,000]
8005
8006
8007
8008 |CHECK THAT CARRYOUT COUT15 ABOVE GENERATED CORRECTLY AS A (0)
8009 4473:
8010 TEST136B2:
8011 PO, LOAD-ENUA(ZTARGET402), |BIT<00> CLEAR
8012 LOAD-ERROR(TEST136B2), |ERROR DIRECTORY KEY
```

```
0013      DCS=CTR(C3.),      |COMPARE AT TARGET
0014      NEXT, J/GOBUT136B2
(4473) DCS(1.00,1.0,0.0) BM(1100..00,11..11,00..0000..010...0.0,0.0,0.0...0.0000...0..0000,0...11,000...010,001,111)
0015
0016      4217: 1(FREE)
0017      GOBUT136B2:
0018      SETUP, RETURN/SCOPE136B,      |RETURN TO SCOPE LOOP TEST WORD
0019      NEXT, GOTO-PAGE(7),      |BUT TABLE
0020      J/BUTD1C1A      |D(C)H B COUT15 H IN BIT400
(4217) DCS(0.00,0.0,0.0) BM(0100..00,01..00,10..0000..111...0.0,0.0,0.0...0.0000...0..0000,0...11,100...011,100,100)
0021
0022
0023      4220: 1(FREE)
0024      SCOPE136B:
0025      PO, BUMP-VERIFY,      |COUNT
0026      NEXT, BUTD[SCOPE],      |NO ERROR: "TEST320A" (+1, WORDS)
0027      J/TEST320A      |ERROR: "DPA136A1" (-11, WORDS)
0028
(4220) DCS(0.00,0.1,0.1) BM(0000..00,00..00,00..0000..000...0.0,0.0,0.0...0.0000...0..0000,0...11,000...101,100,011)
0029
0030
0031
0032
0033
0034
0035      |-----
0036
0037      | THE FOLLOWING TWO SUBROUTINES ARE USED IN THE ABOVE CARRY LOOKAHEAD/INTERNAL CARRIES TESTS:
0038
0039
0040      7007: 1(FREE)
0041      ALUCARRY1:
0042      P3-T, D_A-PLUS-B-PLUS-D(C),      |D <- SUM OF A, B; D(C) WAS SET PREVIOUSLY
0043      D(C)_COUT15,      |GET CARRYOUT FOR EXAMINATION LATER
0044      SR-A-PLUS-B-PLUS-D(C),      |ALSO GET IT INTO THE SR
0045      BUS-A-SR,      |A-SIDE OPERAND WAS IN THE SR
0046      BUS-B-CSPD(D16),      |B-SIDE OPERAND WAS IN CSP(16)
0047      NEXT, J/ALUCARRY1
(7007) DCS(0.00,0.0,0.0) BM(0100..10,00..00,00..0000..110...1.1,1.0,0.0...0.0001...0..0000,0...11,000...000,001,111)
0048
0049      7017: 1(FREE)
0050      ALUCARRY1A:
0051      P2-T, D_SR-XOR-CSPD(D06), SAVE=D(C),      |COMPARE RECEIVED : (007417)
0052      NEXT, J/DINTOIRS      |GO PUT D => IR, BUT(INSTRS) FOR (000000)
(7017) DCS(0.00,0.0,0.0) BM(0110..10,00..00,00..0000..111...0.1,0.0,0.0...0.1001...0..0000,0...11,000...010,111,011)
0053
0054
0055
0056
0057      7020: 1(FREE)
0058      ALUCARRY2:
0059      P3-T, D_A-PLUS-NOT-B-PLUS-D(C),      |D <- DIFF OF A, B; D(C) WAS SET PREVIOUSLY
0060      D(C)_COUT15,      |GET CARRYOUT FOR EXAMINATION LATER
```

```
0061      SR-A-PLUS-NOT-B-PLUS-D(C),      |ALSO GET IT INTO THE SR
0062      BUS-A-SR,      |A-SIDE OPERAND WAS IN THE SR
0063      BUS-B-CSPD(D16),      |B-SIDE OPERAND WAS IN CSP(16)
0064      NEXT, J/ALUCARRY2A
(7020) DCS(0.00,0.0,0.0) BM(0101..10,00..00,00..0000..110...1.1,1.0,0.0...0.0001...0..0000,0...11,000...000,010,001)
0065
0066      7021: 1(FREE)
0067      ALUCARRY2A:
0068      P2-T, D_SR-XOR-CSPD(D05), SAVE=D(C),      |COMPARE RECEIVED : (170360)
0069      NEXT, J/DINTOIRS      |GO PUT D => IR, BUT(INSTRS) FOR (000000)
(7021) DCS(0.00,0.0,0.0) BM(0110..10,00..00,00..0000..111...0.1,0.0,0.0...0.1010...0..0000,0...11,000...010,111,011)
0070
0071
0072
0073
0074
0075      |.PAGE=====
0076
0077      .TNC * TEST320: D(C) SELECTION / COUT07-DOUT07 / D<14:00>=ZERO&BIT<00>
0078
0079
0080
0081      |*****
0082      |*
0083      |* TESTS: 320 A - F      |WORDS: 026 + 014
0084      |*
0085      |* FUNCTIONS:
0086      |*
0087      |* THESE SIX TESTS CHECK THE D(C) ADDRESS SELECTION LOGIC, COUT07 CARRY
0088      |* AND DOUT07 BITS, AND THE D<14:00>=ZERO BUT, WHEN ONLY BIT<00>=1 (NOT
0089      |* CHECKED IN TEST410).
0090      |*
0091      |*****
0092
0093
0094
0095      |TEST-320-A SETS ONLY THE D(C) INPUT "ALU00" (CODE=010), AND THEN CHECKS THAT D(C)_ALU00
0096      | RESULTS IN A "1".
0097      4543:
0098      TEST320A:
0099      PO, LOAD-ENUA(ZTARGET403),      |BIT<00> SET
0100      LOAD-ERROR(TEST320A),      |ERROR DIRECTORY KEY
0101      DCS=CTR(C6.),      |COMPARE AT TARGET
0102      NEXT, J/SETONE320A
(4543) DCS(1.00,1.0,0.0) BM(1001..00,11..11,00..0000..011...0.0,0.0,0.0...0.0000...0..0000,0...11,000...111,000,000)
0103
0104      4700:
0105      SETONE320A:
0106      PO, BUMP-VERIFY,      |COUNT
0107      P3, CSPD(17)_EMIT, EMIT/0000001,      |A ONE
0108      NEXT, J/SETD320A
(4700) DCS(0.00,0.0,0.1) BM(0000..10,00..00,00..0000..001...0.0,0.0,0.0...0.0000...1..0000,0...11,000...010,010,001)
0109
0110      4721: 1(FREE)
```

```
8111 SETD320A;
8112 P2-T, D_CSPD(D17), D(C)=0, ID GETS (000001)
8113 SR_CSPD(D17), IS0 DOES SR
8114 P3, A#BSPI(11)_D, STORE THE CONSTANT
8115 NEXT, J/GETDC320A
(4221) DCS[0.00,0.0,0.0] BM[1010.10.11.00.01.000.000.0.0.1.1.0.0.0.0000.0.1011.0.11.000.010.010.010]
8116
8117 4222: 1(FREE)
8118 GETDC320A;
8119 P2-T, D_A=PLUS-B=PLUS-0, D(C)=ALU00, [D(C) CODE (010)
8120 BUS=A_SR, IS0(000001)
8121 BUS=B_CSPD(C000000), IS0(000000)
8122 NEXT, J/GOBUT320A ID=(000001), ONLY D(C)=ALU00 SET
(4222) DCS[0.00,0.0,0.0] BM[1001.10.00.00.00.000.010.0.0.1.0.0.0.0.0100.0.0000.0.11.000.010.010.011]
8123
8124 4223: 1(FREE)
8125 GOBUT320A;
8126 PO, BUMP=VERIFY, ICOUNT
8127 SETUP, RETURN/TEST320B, RETURN TO START OF NEXT SUBTEST
8128 NEXT, GOTO=PAGE(7), BUT TABLE
8129 J/BUTD(C1) ID(C) IN BIT<00>
(4223) DCS[0.00,0.0,0.1] BM[0100.00.10.11.01.010.111.0.0.0.0.0.0.0.0000.0.0000.0.11.100.011.100.100]
8130
8131
8132
8133
8134 1 - - - - -
8135
8136 TEST-320-B CHECKS THAT D<14:00>=ZERO IS NOT SET WHEN D=(000001)
8137 4552:
8138 TEST320B;
8139 PO, LOAD=ENUA(ZTARGET400), [BIT<01> CLEAR
8140 LOAD=ERROR(TEST320B), ERROR DIRECTORY KEY
8141 DCS=CTR(C3), COMPARE AT TARGET
8142 BUMP=VERIFY, ICOUNT
8143 NEXT, J/GOBUT320B
(4552) DCS[1.00,1.0,0.1] BM[1100.00.11.11.00.000.000.0.0.0.0.0.0.0.0000.0.0000.0.11.000.010.010.100]
8144
8145 4224: 1(FREE)
8146 GOBUT320A;
8147 SETUP, RETURN/TEST320C, RETURN TO START OF NEXT SUBTEST
8148 NEXT, GOTO=PAGE(7), BUT TABLE
8149 J/BUTD=IS=ZERO ICHECK ON IT
(4224) DCS[0.00,0.0,0.0] BM[0100.00.10.11.10.010.111.0.0.0.0.0.0.0.0000.0.0000.0.11.100.011.100.001]
8150
8151
8152
8153 1 - - - - -
8154
8155 TEST-320-C CHECKS THAT THE COUT15/COUT07 SIGNALS DON'T TRACK EACH OTHER
8156 4567:
8157 TEST320C;
```

```
8158 PO, LOAD=ENUA(ZTARGET413), [BIT<02> CLEAR
8159 LOAD=ERROR(TEST320C), ERROR DIRECTORY KEY
8160 DCS=CTR(C6), COMPARE AT TARGET
8161 NEXT, J/SETONE320C
(4562) DCS[1.00,1.0,0.0] BM[1001.00.11.11.00.001.011.0.0.0.0.0.0.0.0000.0.0000.0.11.000.010.010.101]
8162
8163 4225: 1(FREE)
8164 SETONE320C;
8165 P3, CSPD[16]_EMIT, EMIT/100000, 1A ONE IN B15
8166 NEXT, J/SETD320C
(4225) DCS[0.00,0.0,0.0] BM[1000.10.00.00.00.000.000.0.0.0.0.0.0.0.0001.1.0000.0.11.000.010.010.110]
8167
8168 4226: 1(FREE)
8169 RETD320C;
8170 P2-T, D_CSPD(D16), D(C)=ALU15, ID GETS (100000)
8171 SR_CSPD(D16), IS0 DOES SR
8172 P3, A#BSPI(13)_D, STORE THE CONSTANT
8173 NEXT, J/GETDC320C
(4226) DCS[0.00,0.0,0.0] BM[1010.10.11.00.01.001.100.0.0.1.1.0.0.0.0001.0.0.1011.0.11.000.010.010.111]
8174
8175 4227: 1(FREE)
8176 GETDC320C;
8177 PO, BUMP=VERIFY, ICOUNT
8178 P2-T, D_A=PLUS-B=PLUS-0, D(C)=COUT07, [D(C) GETS COUT07=0, COUT15=1
8179 BUS=A_SR, IS0(100000)
8180 BUS=B_CSPD(C125252), IS0(125252)
8181 NEXT, J/GOBUT320C ID=(025252)
(4227) DCS[0.00,0.0,0.1] BM[1001.10.00.00.00.000.010.0.0.1.0.0.0.0.0110.0.0000.0.11.000.010.011.000]
8182
8183 4230: 1(FREE)
8184 GOBUT320C;
8185 SETUP, RETURN/TEST320D, RETURN TO START OF NEXT SUBTEST
8186 NEXT, GOTO=PAGE(7), BUT TABLE
8187 J/BUTD(C1) ID(C) IN BIT<02>
(4230) DCS[0.00,0.0,0.0] BM[0100.00.10.10.11.111.111.0.0.0.0.0.0.0.0000.0.0000.0.11.100.011.001.000]
8188
8189
8190
8191
8192 1 - - - - -
8193
8194
8195 TEST-320-D CHECKS THAT THE BUTR(COUT07#DOUT07) SEES THE (01*) THAT WAS GENERATED
8196 4537:
8197 TEST320D;
8198 PO, LOAD=ENUA(ZTARGET403), [BIT<2:1> = "01"
8199 LOAD=ERROR(TEST320D), ERROR DIRECTORY KEY
8200 DCS=CTR(C3), COMPARE AT TARGET
8201 NEXT, J/GOBUT320D
(4537) DCS[1.00,1.0,0.0] BM[1100.00.11.11.00.000.011.0.0.0.0.0.0.0.0000.0.0000.0.11.000.010.011.001]
8202
8203 4231: 1(FREE)
8204 GOBUT320A;
```

```
8205      SETUP, RETURN/TEST320E, IRETURN TO START OF NEXT SUBTEST
8206      NEXT, GOTO=PAGE(7), I BUT TABLE
8207      J/BUTCOUT7DOUT7 I COUT7DOUT7 IN BIT<211>
(4231) DCS[0.00.0.0.0.0] BM[0100..00.10..11.01..100..111...0.0.0..0..0.0000...0..0000.0...11.100...011.001.001]

8208
8209
8210
8211
8212      I - - - - -
8213
8214      ITEST-320-E SETS ONLY THE D[C] INPUT "COUT07" (CODE=101), AND THEN CHECKS THAT D[C]_COUT07
8215      I RESULTS IN A "1".
8216      45541
8217      TEST320E1
8218      PO, LOAD=ENUA(ZTARGET403), I BIT<01> SET
8219      LOAD=ERROR(TEST320E), I ERROR DIRECTORY KEY
8220      DCS=CTR(C.), I COMPARE AT TARGET
8221      NEXT, J/SETONE320E
(4554) DCS[1.00.1.0.0.0] BM[1001..00.11..11.00..000..011...0.0.0..0..0.0000...0..0000.0...11.000...010.011.010]

8222
8223      42321 I(FREE)
8224      SETONE320E1
8225      PO, BUMP-VERIFY, I COUNT
8226      P3, CSPD[15]_EMIT, EMIT/000200, I A ONE IN BIT07
8227      NEXT, J/SETD320E
(4232) DCS[0.00.0.0.0.1] BM[0000..10.00..00.10..000..000...0.0.0..0..0.0010...1..0000.0...11.000...010.011.011]

8228
8229      42331 I(FREE)
8230      SETD320E1
8231      P2-T, D_CSPD[D15], D[C]=0, I D GETS (000200)
8232      SR_CSPD[D15], I SO DOES SR
8233      P3, A&BSPHI[15]_D, I STORE THE CONSTANT
8234      NFXT, J/GETDC320E
(4233) DCS[0.00.0.0.0.0] BM[1010..10.11..00.01..010..000...0.1.1..0..0.0010...0..1011.0...11.000...010.011.100]

8235
8236      42341 I(FREE)
8237      GETDC320E1
8238      P2-T, D_A-PLUS-B-PLUS=0, D[C]_COUT07, I D[C] CODE (101)
8239      BUS-A_SR, I A=(000200)
8240      BUS-B_CSPD[D15], I B=(000200)
8241      NFXT, J/GOBUT320E I D=(000400), ONLY D[C]=COUT07 SET
(4234) DCS[0.00.0.0.0.0] BM[1001..10.00..00.00..000...101...0.1.0..0..0.0010...0..0000.0...11.000...010.011.101]

8242
8243      42351 I(FREE)
8244      GOBUT320E1
8245      SETUP, RETURN/TEST320F, IRETURN TO START OF NEXT SUBTEST
8246      NFXT, GOTO=PAGE(7), I BUT TABLE
8247      J/BUTD[C] I D[C] IN BIT<01>
(4235) DCS[0.00.0.0.0.0] BM[0100..00.10..11.01..110..111...0.0.0..0..0.0000...0..0000.0...11.100...011.101.000]

8248
8249
8250
```

```
8251
8252      I - - - - -
8253
8254      ITEST-320-F CHECKS THAT THE BUTR(COUT07DOUT07) SEES THE (10*) THAT WAS GENERATED
8255      45561
8256      TEST320F1
8257      PO, LOAD=ENUA(ZTARGET405), I BIT<211> = "10"
8258      LOAD=ERROR(TEST320F), I ERROR DIRECTORY KEY
8259      DCS=CTR(C.), I COMPARE AT TARGET
8260      NEXT, J/GOBUT320F
(4556) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..000..101...0.0.0..0..0.0000...0..0000.0...11.000...010.011.110]

8261
8262      42361 I(FREE)
8263      GOBUT320F1
8264      SETUP, RETURN/SCOPE320, IRETURN TO SCOPE LOOP TEST WORD
8265      NEXT, GOTO=PAGE(7), I BUT TABLE
8266      J/BUTCOUT7DOUT7 I COUT7DOUT7 IN BIT<211>
(4236) DCS[0.00.0.0.0.0] BM[0100..00.01..00.11..111..111...0.0.0..0..0.0000...0..0000.0...11.100...011.001.001]

8267
8268
8269
8270      42371 I(FREE)
8271      SCOPE3201
8272      PO, BUMP-VERIFY, I COUNT
8273      BUSDIN_EMIT-[I], I RESET PROC UCONS
8274      EN=CLK-IR[15=00], I
8275      NEXT, BUTD[SCOPE], I NO ERROR: "TEST350" (+1. WORDS)
8276      J/TEST350 I ERROR: "SETONE320A" (-20. WORDS)
(4237) DCS[0.00.0.1.0.1] BM[0000..00.00..00.01..000..100...0.0.0..0..0.11001...0..0000.0...11.000...111.000.001]

8277
8278
8279
8280
8281
8282
8283      I .PAGE=====
8284
8285
8286      .TNC * TEST350-352: ASP/BSP HI/LO ADDRESSING MODES, DATA VALIDITY
8287
8288      I*****
8289      I*
8290      I* TESTS: 350 - 352 UMWORDS: 075 + 125
8291      I*
8292      I* FUNCTION8:
8293      I*
8294      I* VERIFIES THE ASP/BSP ADDRESSING MODES.
8295      I*
8296      I*****

8297
8298
8299
8300      ITEST 350 A-D USES THE A/B SP "SF" AND "PF" ADDRESS MODES TO WRITE A UNIQUE PATTERN TO
```

LOCATIONS (00107) OF ASPHI, ASPLO, BSPHI, BSPLO, THRU USE OF A COUNT LOOP.
THE GENERATED PATTERN IS THEN READ BACK, AND COMPARED TO A REGENERATED ORIGINAL,
TO CHECK FOR ADDRESSING CORRECTNESS, ABILITY TO READ/WRITE, AND DATAPATH VALIDITY.

FIRST FILL UP THE SCRATCHPADS:

AT THE END OF THE FILLUP LOOP, A/B SP HI/LO LOOK LIKE THIS:

LOCTN	ASPHI	BSPHI	ASPLO	BSPLO
00	080700	080700	057077	057077
01	051611	051611	056166	056166
02	052522	052522	055255	055255
03	053433	053433	054344	054344
04	054344	054344	053433	053433
05	055255	055255	052522	052522
06	056166	056166	051611	051611
07	057077	057077	050700	050700
10	000001	000001	000125	000125
11	000001	000001	000125	000125
12	000152	000152	000125	000125
13	100000	100000	000125	000125
14	000200	000200	000125	000125
15	000200	000200	000125	000125
16	000200	000200	000125	000125
17	000200	000200	000125	000125

NOTE: CONSTANTS FOR DCS IN ASPHI/BSPHI 01/03/05/07 WERE DESTROYED,
AND *MUST* BE RESTORED AFTER LEAVING THIS TEST SEQUENCE.

```

47011
TEST350:
P0,    LOAD-ERROR( TEST350),          IERROR DIRECTORY KEY
P3,    CSPD(15)_EMIT, EMIT/177067,     IADDED TO PATTERN "N" TO GET "N+1"
NEXT,  J/START350
(4701) DCS(1,00,0,0,0,0) BM(1111..10.11..10.00..110..111...0,0,0,0,0,0...0,0010...1..0000,0...11,000...010,100,000)

42401 I(FREE)
START350:
P0,    BUMP-VERIFY,
P3,    CSPD(14)_EMIT, EMIT/057077,     ICOUNT
NEXT,  J/LOADSRD350                    ITHE INITIAL PATTERN
(4240) DCS(0,00,0,0,0,1) BM(0101..10.11..10.00..111..111...0,0,0,0,0,0...0,0011...1..0000,0...11,000...010,100,001)

42411 I(FREE)
LOADSRD350:
P2=T,  D_CSPD(D14),                    IINITIAL PATTERN
SR_CSPD(D14),                          ID=>IR, SR=TEMP

```

```

NEXT,    GOTO-PAGE(7),
J/DINTOIR350
(4241) DCS(0,00,0,0,0,0) BM(1010..10.00..00.00..000..111...0,1,1,0,0,0...0,0011...0..0000,0...11,100...011,101,011)

*** LOOP BACK POINT FOR SP FILLUP ***

7353:
DINTOIR350:
SETUP,  RETURN/WRTDF350,                ICALL SUBR FOR D -> IR
NEXT,  CALL(DINTOIR)
(7353) DCS(0,00,0,0,0,0) BM(0100..00.00..00.00..010..111...0,0,0,0,0,0...0,0000...0..0000,0...11,100...010,110,111)

4002: I(FREE)
WRTDF350:
P0,    DCS-CTR(C15,),                    IHOLD UP
P3,    A#BSPHI(DF)_D,                    IUSE DF TO WRITE A/B HI, W/ WR(AB,H,A)
NEXT,  GOTO-PAGE(7),
J/WRTSF350
(4002) DCS(0,00,1,0,0,0) BM(0000..00.00..00.10..000..111...0,0,0,0,0,0...0,0000...0..1011,0...11,100...000,001,110)

7016: I(FREE)
WRTSF350:
P3,    A#BSPLO(SF)_D,                    IUSE SF TO WRITE A/B LO, W/ WR(AB,L,B)
NEXT,  J/NEXTPAT350
(7016) DCS(0,00,0,0,0,0) BM(0000..00.01..00.11..000..000...0,0,0,0,0,0...0,0000...0..0011,0...11,000...000,010,011)

7023: I(FREE)
NEXTPAT350:
P2=T,  SR=D_SR-PLUS-CSPD(D15), SAVE=D(C), IINCREMENT PATTERN
NEXT,  BUTR(DN0),                        IIF TRUE: "LOADHI350" (+1, WORDS) EXIT LOOP
J/DINTOIR350                             IIF FALSE: "DINTOIR350" (-3, WORDS) KEEP GOING
(7023) DCS(0,00,0,0,0,0) BM(1001..10.00..00.00..000..111...0,1,1,0,0,0...0,0010...0..0000,0...01,001...011,101,011)

ICOME HERE IF DONE LOOPING
7357:
LOADHI350:
*** IF ANY ASP/BSP HI/LO BIT<03> IN ADDR STUCK ZERO,
ONE OF THE FOLLOWING TWO WORDS WILL OVERWRITE
A PREVIOUSLY WRITTEN LOCATION (IE, 02 OR 03) ***
P2=T,  D_CSPD(D01), SAVE=D(C),          IPATTERN OF (000152)
P3,    A#BSPHI(12)_D,                    IWRITE SPHI WITH ADDR BIT<03> SET, DF=0, W/ WR(AB,H,A)
NEXT,  J/LOADHI350                       I FOR USE IN TEST373
(7357) DCS(0,00,0,0,0,0) BM(1010..10.10..00.00..001..111...0,1,0,0,0,0...0,1110...0..1011,0...11,000...000,010,100)

7024: I(FREE)
LOADHI350:
P2=T,  D_CSPD(D02), SAVE=D(C),          IPATTERN OF (000125)
P3,    A#BSPLO(13)_D,                    IWRITE SPHI WITH ADDR BIT<03> SET, DF=0, W/ WR(AB,L,B)
NEXT,  J/AGTWSRD350
(7024) DCS(0,00,0,0,0,0) BM(1010..10.11..00.01..001..111...0,1,0,0,0,0...0,1101...0..0011,0...11,000...000,010,101)

```

```
0398 7025: 1(FREE)
0399 AGAINSRD350:
0400 P2-T, D_CSPD(D14), IAGAIN RESET D, SR TO INITIAL PATTERN
0401 SR_CSPD(D14), IOF (057077)
0402 NEXT, GOTO-PAGE(4), I
0403 J/DINTOIR350 I
(7025) DCS(0.00,0.0,0.0) BM(1010..10.00..00.00..0000..100...0.1.1.0.0...0.0011...0..0000,0...11.100...101.110,011)
0404
0405 *** LOOP BACK ENTRY POINT FOR TESTS ***
0406
0407 4563:
0408 DINTOIR350:
0409 SETUP, RETURN/TEST350A, IGO TO SUBR WHICH PUTS SR -> IR FOR SF/DF/DH
0410 NEXT, CALL(SRINTOIR) I
(4563) DCS(0.00,0.0,0.0) BM(10100..00.11..11.01..010..111...0.0,0.0.0.0...0.0000...0..0000,0...11.100...010.110,110)
0411
0412 I - - - - -
0413
0414
0415 ITEST 350A CHECKS BSP-ADDRS/SF, BSPLO ADDR FOR ERRORS
0416 4752:
0417 TEST350A:
0418 PO, LOAD-ENUA(ZTARGET434), IINSTRS FOR IR=(000000)
0419 LOAD-ERROR(TEST350A), IERROR DIRECTORY KEY
0420 DCS-CTR(C7.), ICOMPARE AT TARGET
0421 NEXT, J/COMP350A I
(4752) DCS(1.00,1.0,0.0) BM(1000..00.11..11.00..011..100...0.0,0.0.0.0...0.0000...0..0000,0...11.000...111.000,010)
0422
0423 4702:
0424 COMP350A:
0425 P2-T, D_SR-XOR-BSPLO(SF), SAVE-D(C), ICOMPARE EXPECTED:RECEIVED, BITWISE
0426 NEXT, J/GOBUT350A I
(4702) DCS(0.00,0.0,0.0) BM(0110..00.01..00.00..0000..111...0.1,0.0.0.0...0.0000...0..0000,0...11.000...010.100,011)
0427
0428 4243: 1(FREE)
0429 GOBUT350A:
0430 PO, BUMP-VERIFY, ICOUNT
0431 SETUP, RETURN/RESETIR350A, IGO TO SUBR WHICH:
0432 NEXT, CALL(DINTOIR=5) I D -> IR, BUT(INSTRS)
(4243) DCS(0.00,0.0,0.1) BM(0111..00.00..00.10..010..111...0.0,0.0.0.0...0.0000...0..0000,0...11.100...010.111,011)
0433
0434 7022: 1(FREE)
0435 RESETIR350A:
0436 SETUP, RETURN/TEST350B, IGO TO SUBR WHICH PUTS SR -> IR FOR SF/DF/DH
0437 NEXT, CALL(SRINTOIR) I
(7022) DCS(0.00,0.0,0.0) BM(0100..00.11..11.01..011..111...0.0,0.0.0.0...0.0000...0..0000,0...11.100...010.110,110)
0438
0439 I - - - - -
0440
0441
0442 ITEST 350B CHECKS BSP-ADDRS/DF, BSPHI ADDR FOR ERRORS
```

```
0443 4753:
0444 TEST350B:
0445 PO, LOAD-ENUA(ZTARGET434), IINSTRS FOR IR=(000000)
0446 LOAD-ERROR(TEST350B), IERROR DIRECTORY KEY
0447 DCS-CTR(C7.), ICOMPARE AT TARGET
0448 NEXT, J/COMP350B I
(4753) DCS(1.00,1.0,0.0) BM(1000..00.11..11.00..011..100...0.0,0.0.0.0...0.0000...0..0000,0...11.000...010.100,100)
0449
0450 4244: 1(FREE)
0451 COMP350B:
0452 P2-T, D_SR-XOR-BSPHI(DF), SAVE-D(C), ICOMPARE EXPECTED:RECEIVED, BITWISE
0453 NEXT, J/GOBUT350B I
(4244) DCS(0.00,0.0,0.0) BM(0110..01.00..00.00..0000..111...0.1,0.0.0.0...0.0000...0..0000,0...11.000...010.100,101)
0454
0455 4245: 1(FREE)
0456 GOBUT350B:
0457 SETUP, RETURN/RESETIR350B, IGO TO SUBR WHICH:
0458 NEXT, CALL(DINTOIR=5) I D -> IR, BUT(INSTRS)
(4245) DCS(0.00,0.0,0.0) BM(0111..00.00..00.10..110..111...0.0,0.0.0.0...0.0000...0..0000,0...11.100...010.111,011)
0459
0460 7026: 1(FREE)
0461 RESETIR350B:
0462 SETUP, RETURN/TEST350C, IGO TO SUBR WHICH PUTS SR -> IR FOR SF/DF/DH
0463 NEXT, CALL(SRINTOIR) I
(7026) DCS(0.00,0.0,0.0) BM(0100..00.11..11.00..010..111...0.0,0.0.0.0...0.0000...0..0000,0...11.100...010.110,110)
0464
0465 I - - - - -
0466
0467
0468
0469 ITEST 350C CHECKS ASP-ADDRS/DF, ASPLO ADDR FOR ERRORS
0470 4742:
0471 TEST350C:
0472 PO, LOAD-ENUA(ZTARGET434), IINSTRS FOR IR=(000000)
0473 LOAD-ERROR(TEST350C), IERROR DIRECTORY KEY
0474 DCS-CTR(C7.), ICOMPARE AT TARGET
0475 BUMP-VERIFY, ICOUNT
0476 NEXT, J/COMP350C I
(4742) DCS(1.00,1.0,0.1) BM(1000..00.11..11.00..011..100...0.0,0.0.0.0...0.0000...0..0000,0...11.000...010.100,110)
0477
0478 4246: 1(FREE)
0479 COMP350C:
0480 P2-T, D_ASPLO(DF)-XOR-BSPHI(SF), SAVE-D(C), ICOMPARE RECEIVED:EXPECTED, BITWISE
0481 NEXT, J/GOBUT350C I
(4746) DCS(0.00,0.0,0.0) BM(0110..01.01..10.10..0000..111...0.1,0.0.0.0...0.0000...0..0000,0...11.000...010.100,111)
0482
0483 4247: 1(FREE)
0484 GOBUT350C:
0485 SETUP, RETURN/RESETIR350C, IGO TO SUBR WHICH:
0486 NEXT, CALL(DINTOIR=5) I D -> IR, BUT(INSTRS)
(4247) DCS(0.00,0.0,0.0) BM(0111..00.00..00.10..111..111...0.0,0.0.0.0...0.0000...0..0000,0...11.100...010.111,011)
0487
```



```
8583      SETUP, RETURN/TEST351B,      !GO TO SUBR WHICH:
8584      NEXT,  CALL[DINTOIR=S]         ! D -> IR, BUT(INSTRS)
(4256) DCS[0.00,0.0,0.0,1] BM[0100..00.11..10.10..010..111...0.0,0..0..0..0.0000...0..0000,0...11.100...010.111,011]

8585
8586
8587      | - - - - -
8588
8589      !TEST 351B CHECKS ASPLO RIF ADDR "001 1"=(03)
8590      4722:
8591      TEST351B:
8592      PO,      LOAD=ENUA(ZTARGET434),      !INSTRS FOR IR=(000000)
8593      LOAD=ERROR(TEST351B),              !ERROR DIRECTORY KEY
8594      DCS=CTR(C7,),                      !COMPARE AT TARGET
8595      NEXT,    J/COMP351B
(4722) DCS[1.00,1.0,0.0,0] BM[1000..00.11..11.00..011..100...0.0,0..0..0..0.0000...0..0000,0...11.000...010.101,111]

8596
8597      4257: I(FREE)
8598      COMP351B:
8599      P2=T,    D_CSPP[06]=XOR=ASPLO(R03), SAVE=D(C), !COMPARE EXPECTED:RECEIVED, BITWISE
8600      NEXT,    J/GOBUT351B
(4257) DCS[0.00,0.0,0.0,0] BM[0110..10.00..10.01..101..111...0.1,0..0..0..0.1001...0..0000,0...11.000...010.110,000]

8601
8602      4260: I(PREF)
8603      GOBUT351B:
8604      SETUP,   RETURN/TEST351C,          !GO TO SUBR WHICH:
8605      NEXT,    CALL[DINTOIR=S]           ! D -> IR, BUT(INSTRS)
(4260) DCS[0.00,0.0,0.0,0] BM[0100..00.11..10.10..011..111...0.0,0..0..0..0.0000...0..0000,0...11.100...010.111,011]

8606
8607      | - - - - -
8608
8609
8610      !TEST 351C CHECKS ASPHI RIF ADDR "010 0"=(04)
8611      4723:
8612      TEST351C:
8613      PO,      LOAD=ENUA(ZTARGET434),      !INSTRS FOR IR=(000000)
8614      LOAD=ERROR(TEST351C),              !ERROR DIRECTORY KEY
8615      DCS=CTR(C7,),                      !COMPARE AT TARGET
8616      BUMP=VERIFY,                        !COUNT
8617      NEXT,    J/COMP351C
(4723) DCS[1.00,1.0,0.0,1] BM[1000..00.11..11.00..011..100...0.0,0..0..0..0.0000...0..0000,0...11.000...010.110,001]

8618
8619      4261: I(FREE)
8620      COMP351C:
8621      P2=T,    D_CSPP[06]=XOR=ASPHI(R04), SAVE=D(C), !COMPARE EXPECTED:RECEIVED, BITWISE
8622      NEXT,    J/GOBUT351C
(4261) DCS[0.00,0.0,0.0,0] BM[0110..10.00..11.00..110..111...0.1,0..0..0..0.1001...0..0000,0...11.000...010.110,010]

8623
8624      4262: I(FREE)
8625      GOBUT351C:
8626      SETUP,   RETURN/TEST351D,          !GO TO SUBR WHICH:
8627      NEXT,    CALL[DINTOIR=S]           ! D -> IR, BUT(INSTRS)
```

```
(4262) DCS[0.00,0.0,0.0,0] BM[0100..00.11..10.01..010..111...0.0,0..0..0..0.0000...0..0000,0...11.100...010.111,011]

8628
8629      | - - - - -
8630
8631
8632      !TEST 351D CHECKS ASPHI RIF ADDR "010 1"=(05)
8633      4712:
8634      TEST351D:
8635      PO,      LOAD=ENUA(ZTARGET434),      !INSTRS FOR IR=(000000)
8636      LOAD=ERROR(TEST351D),              !ERROR DIRECTORY KEY
8637      DCS=CTR(C7,),                      !COMPARE AT TARGET
8638      NEXT,    J/COMP351D
(4712) DCS[1.00,1.0,0.0,0] BM[1000..00.11..11.00..011..100...0.0,0..0..0..0.0000...0..0000,0...11.000...010.110,011]

8639
8640      4263: I(FREE)
8641      COMP351D:
8642      PO,      BUMP=VERIFY,                !COUNT
8643      P2=T,    D_CSPP[05]=XOR=ASPHI(R05), SAVE=D(C), !COMPARE EXPECTED:RECEIVED, BITWISE
8644      NEXT,    J/GOBUT351D
(4263) DCS[0.00,0.0,0.0,1] BM[0110..10.00..11.01..110..111...0.1,0..0..0..0.1010...0..0000,0...11.000...010.110,100]

8645
8646      4264: I(FREE)
8647      GOBUT351D:
8648      PO,      BUMP=VERIFY,                !COUNT
8649      SETUP,   RETURN/SCOPE351,           !GO TO SUBR WHICH:
8650      NEXT,    CALL[DINTOIR=S]           ! D -> IR, BUT(INSTRS)
(4264) DCS[0.00,0.0,0.0,1] BM[0100..00.01..01.10..101..111...0.0,0..0..0..0.0000...0..0000,0...11.100...010.111,011]

8651
8652
8653
8654      4265: I(FREE)
8655      SCOPE351:
8656      NEXT,    BUTD[SCOPE],                !NO ERROR: "TEST352A" (+1, WORDS)
8657      J/TEST352A                          ! ERROR: "COMP351A" (-1, WORDS)
(4265) DCS[0.00,0.1,0.0,0] BM[0000..00.00..00.00..000..000...0.0,0..0..0..0.0000...0..0000,0...11.000...111.000,101]

8658
8659
8660
8661
8662      | - - - - -
8663
8664      !TESTS 352 A-D VERIFY THAT THE RIF ADDRESS, WITH B&P HI/LO IMMED0/1 MODES, OPERATES CORRECTLY
8665
8666      | - - - - -
8667
8668      !TEST 352A CHECKS BSPLO RIF ADDR "001 0"=(02)

8669      4705:
8670      TEST352A:
8671      PO,      LOAD=ENUA(ZTARGET434),      !INSTRS FOR IR=(000000)
8672      LOAD=ERROR(TEST352A),              !ERROR DIRECTORY KEY
8673      DCS=CTR(C7,),                      !COMPARE AT TARGET
8674      NEXT,    J/COMP352A
(4705) DCS[1.00,1.0,0.0,0] BM[1000..00.11..11.00..011..100...0.0,0..0..0..0.0000...0..0000,0...11.000...111.001,100]
```

```
8675
8676 4714:
8677 COMP352A:
8678 P2-T, D_ASPL0[02]-XOR-BSPLO(R02), SAVE-D[C], [COMPARE EXPECTED:RECEIVED, BITWISE
8679 NEXT, J/GOBUT352A [DATA=(09255)]
(4714) DCS[0.00,0.0.0.0] BM[0110.00.10.10.00.101.111...0.1.0.0.0...0.0000...0.0000,0...11.000...010.110,110]

8680
8681 4266: 1(FREE)
8682 GOBUT352A:
8683 PO, BUMP-VERIFY, [COUNT
8684 SETUP, RETURN/TEST352B, [GO TO SUBR WHICH:
8685 NEXT, CALL[DINTOIR-S] [D -> IR, BUT(INSTRS)
(4266) DCS[0.00,0.0.0.1] BM[0100.00.11.10.01.011.111...0.0.0.0.0...0.0000...0.0000,0...11.100...010.111,011]

8686
8687 1 - - - - -
8688
8689 [TEST 352B CHECKS BSPLO RIF ADDR "001 1"=(03)
8690 4713:
8691 TEST352B:
8692 PO, LOAD-ENUA(ZTARGET434), [INSTRS FOR IR=(000000)
8693 LOAD-ERROR(TEST352B), [ERROR DIRECTORY KEY
8694 DCS-CTR(C7), [COMPARE AT TARGET
8695 NEXT, J/COMP352B
(4713) DCS[1.00,1.0.0.0] BM[1000.00.11.11.00.011.100...0.0.0.0.0...0.0000...0.0000,0...11.000...010.110,111]

8697
8698 4267: 1(FREE)
8699 COMP352B:
8700 P2-T, D_ASPL0[03]-XOR-BSPLO(R03), SAVE-D[C], [COMPARE EXPECTED:RECEIVED, BITWISE
8701 NEXT, J/GOBUT352B [DATA=(094344)]
(4267) DCS[0.00,0.0.0.0] BM[0110.00.11.10.01.101.111...0.1.0.0.0...0.0000...0.0000,0...11.000...010.111,000]

8702
8703 4270: 1(FREE)
8704 GOBUT352B:
8705 SETUP, RETURN/TEST352C, [GO TO SUBR WHICH:
8706 NEXT, CALL[DINTOIR-S] [D -> IR, BUT(INSTRS)
(4270) DCS[0.00,0.0.0.0] BM[0100.00.11.10.10.100.111...0.0.0.0.0...0.0000...0.0000,0...11.100...010.111,011]

8707
8708 1 - - - - -
8709
8710 [TEST 352C CHECKS BSPLO RIF ADDR "010 0"=(04)
8711 4724:
8712 TEST352C:
8713 PO, LOAD-ENUA(ZTARGET434), [INSTRS FOR IR=(000000)
8714 LOAD-ERROR(TEST352C), [ERROR DIRECTORY KEY
8715 DCS-CTR(C7), [COMPARE AT TARGET
8716 BUMP-VERIFY, [COUNT
8717 NEXT, J/COMP352C
(4724) DCS[1.00,1.0.0.1] BM[1000.00.11.11.00.011.100...0.0.0.0.0...0.0000...0.0000,0...11.000...010.111,001]

8719
8720 4271: 1(FREE)
```

```
8721 COMP352C:
8722 P2-T, D_ASPL0[04]-XOR-BSPLO(R04), SAVE-D[C], [COMPARE EXPECTED:RECEIVED, BITWISE
8723 NEXT, J/GOBUT352C [DATA=(093433)]
(4271) DCS[0.00,0.0.0.0] BM[0110.00.10.10.00.110.111...0.1.0.0.0...0.0000...0.0000,0...11.000...010.111,010]

8724
8725 4272: 1(FREE)
8726 GOBUT352C:
8727 SETUP, RETURN/TEST352D, [GO TO SUBR WHICH:
8728 NEXT, CALL[DINTOIR-S] [D -> IR, BUT(INSTRS)
(4272) DCS[0.00,0.0.0.0] BM[0100.00.11.10.10.101.111...0.0.0.0.0...0.0000...0.0000,0...11.100...010.111,011]

8729
8730 1 - - - - -
8731
8732 [TEST 352D CHECKS BSPLO RIF ADDR "010 1"=(05)
8733 4725:
8734 TEST352D:
8735 PO, LOAD-ENUA(ZTARGET434), [INSTRS FOR IR=(000000)
8736 LOAD-ERROR(TEST352D), [ERROR DIRECTORY KEY
8737 DCS-CTR(C7), [COMPARE AT TARGET
8738 NEXT, J/COMP352D
(4725) DCS[1.00,1.0.0.0] BM[1000.00.11.11.00.011.100...0.0.0.0.0...0.0000...0.0000,0...11.000...010.111,011]

8740
8741 4273: 1(FREE)
8742 COMP352D:
8743 PO, BUMP-VERIFY, [COUNT
8744 P2-T, D_ASPL0[05]-XOR-BSPLO(R05), SAVE-D[C], [COMPARE EXPECTED:RECEIVED, BITWISE
8745 NEXT, J/GOBUT352D [DATA=(092522)]
(4273) DCS[0.00,0.0.0.1] BM[0110.00.11.10.01.110.111...0.1.0.0.0...0.0000...0.0000,0...11.000...010.111,100]

8746
8747 4274: 1(FREE)
8748 GOBUT352D:
8749 PO, BUMP-VERIFY, [COUNT
8750 SETUP, RETURN/SCOPE352, [GO TO SUBR WHICH:
8751 NEXT, CALL[DINTOIR-S] [D -> IR, BUT(INSTRS)
(4274) DCS[0.00,0.0.0.1] BM[0100.00.01.01.11.101.111...0.0.0.0.0...0.0000...0.0000,0...11.100...010.111,011]

8752
8753 1 - - - - -
8754
8755 4275: 1(FREE)
8756 SCOPE352:
8757 P2-T, D_CSPD(D13), D[C]_0, [RESTORE CONSTANT (000000)
8758 P3, A[BSPHI[01]_D, [
8759 NEXT, RUTD(SCOPE), [NO ERROR: "RESTORE01" (+1, WORDS)
8760 J/RESTORE01 [ERROR: "COMP352A" (-1, WORDS)
(4275) DCS[0.00,0.1.0.0] BM[1010.10.11.00.01.100.000...0.1.0.0.0...0.0100...0.1011,0...11.000...111.001,101]

8761
8762 1 - - - - -
8763
8764 1 THIS SECTION OF CODE FINISHES RESTORING THE 4 CONSTANTS IN ASPHI/BSPHI
8765 1 THAT WERE WIPED OUT IN THE PREVIOUS GROUP OF TESTS.
8766 1
```

```

0767      4715:
0768      RESTORE01:
0769      P2-T,      D_CSPD(D10),      1(082908)
0770      P3,      A$B$PHI[07]_D,      1
0771      NEXT,      GOTO=PAGE(7),      1
0772      J/RESTORE02      1
0773      (4715) DCS[0,00,0,0,0,0] BM[1010..10.11..00,01..111..111...0,1,0..0..0...0,0111...0..1011,0...11,100...000,011,000]
0774
0775      7030: 1(FREE)
0776      RESTORE02:
0777      P2-T,      D_CSPD(D11), D[C]_0,      1(128252)
0778      P3,      A$B$PHI[05]_D,      1
0779      NEXT,      J/RESTORE03      1
0780      (7030) DCS[0,00,0,0,0,0] BM[1010..10.11..00,01..110..000...0,1,0..0..0...0,0110...0..1011,0...11,000...000,011,010]
0781
0782      7032: 1(FREE)
0783      RESTORE03:
0784      P2-T,      D_CSPD(D12), D[C]_0,      1(177777)
0785      P3,      A$B$PHI[03]_D,      1
0786      NEXT,      J/RESTORE04      1
0787      (7032) DCS[0,00,0,0,0,0] BM[1010..10.11..00,01..101..000...0,1,0..0..0...0,0101...0..1011,0...11,000...000,011,011]
0788
0789      7033: 1(FREE)
0790      RESTORE04:
0791      SETUP,      RETURN/TEST361A,      1GO TO SUBR THAT RESTORES CURRENT
0792      NEXT,      GOTO=PAGE(7),      1 DCS MICROCODE VERSION NUMBER,
0793      J/INSERTREVO      1 B<18>B(0), INTO B,M, GPR "R5"
0794      (7033) DCS[0,00,0,0,0,0] BM[0110..00,10..11.11..011..111...0,0,0..0..0...0,0000...0..0000,0...11,100...010,001,110]
0795
0796
0797      1.PAGE=====
0798
0799
0800      .TDC * TEST361-372: TESTING SR, GUARD, RES, AND XMUX
0801
0802      !=====
0803      !*
0804      !* TESTS: 361 - 372 UNWORDS: 125 + 117
0805      !*
0806      !* FUNCTIONS: TESTS THAT SR CAN BE LOADED FROM ALU OUT, READ THRU
0807      !* SR AND FLOAT PORTS OF XMUX, SR CAN BE SHIFTED LEFT, RIGHT,
0808      !* AND NOP, GUARD-ENABLED, DISABLED, SHIFTED, AND THAT
0809      !* ANY "BUTTABLE" BITS CAN BE TESTED.
0810      !* PREVIOUSLY TESTED IN TEST 105 A-E WAS THE ABILITY TO
0811      !* LOAD/READ THE SR THRU XMUX/SR, AND BUT(SR3-0).
0812      !* THESE FUNCTIONS ARE NOT TESTED HERE.
0813      !*
0814      !=====
0815

```

```

0016
0017
0018
0019
0020
0021
0022
0023
0024
0025
0026
0027
0028
0029
0030
0031
0032
0033
0034
0035
0036
0037
0038
0039
0040
0041
0042
0043
0044
0045
0046
0047
0048
0049
0050
0051
0052
0053
0054
0055
0056
0057
0058
0059
0060
0061
0062
0063
0064
0065
0066
0067
0068
0069
0070
0071
0072
0073
0074
0075
0076
0077
0078
0079
0080
0081
0082
0083
0084
0085
0086
0087
0088
0089
0090
0091
0092
0093
0094
0095
0096
0097
0098
0099
0100
0101
0102
0103
0104
0105
0106
0107
0108
0109
0110
0111
0112
0113
0114
0115
0116
0117
0118
0119
0120
0121
0122
0123
0124
0125
0126
0127
0128
0129
0130
0131
0132
0133
0134
0135
0136
0137
0138
0139
0140
0141
0142
0143
0144
0145
0146
0147
0148
0149
0150
0151
0152
0153
0154
0155
0156
0157
0158
0159
0160
0161
0162
0163
0164
0165
0166
0167
0168
0169
0170
0171
0172
0173
0174
0175
0176
0177
0178
0179
0180
0181
0182
0183
0184
0185
0186
0187
0188
0189
0190
0191
0192
0193
0194
0195
0196
0197
0198
0199
0200
0201
0202
0203
0204
0205
0206
0207
0208
0209
0210
0211
0212
0213
0214
0215
0216
0217
0218
0219
0220
0221
0222
0223
0224
0225
0226
0227
0228
0229
0230
0231
0232
0233
0234
0235
0236
0237
0238
0239
0240
0241
0242
0243
0244
0245
0246
0247
0248
0249
0250
0251
0252
0253
0254
0255
0256
0257
0258
0259
0260
0261
0262
0263
0264
0265
0266
0267
0268
0269
0270
0271
0272
0273
0274
0275
0276
0277
0278
0279
0280
0281
0282
0283
0284
0285
0286
0287
0288
0289
0290
0291
0292
0293
0294
0295
0296
0297
0298
0299
0300
0301
0302
0303
0304
0305
0306
0307
0308
0309
0310
0311
0312
0313
0314
0315
0316
0317
0318
0319
0320
0321
0322
0323
0324
0325
0326
0327
0328
0329
0330
0331
0332
0333
0334
0335
0336
0337
0338
0339
0340
0341
0342
0343
0344
0345
0346
0347
0348
0349
0350
0351
0352
0353
0354
0355
0356
0357
0358
0359
0360
0361
0362
0363
0364
0365
0366
0367
0368
0369
0370
0371
0372
0373
0374
0375
0376
0377
0378
0379
0380
0381
0382
0383
0384
0385
0386
0387
0388
0389
0390
0391
0392
0393
0394
0395
0396
0397
0398
0399
0400
0401
0402
0403
0404
0405
0406
0407
0408
0409
0410
0411
0412
0413
0414
0415
0416
0417
0418
0419
0420
0421
0422
0423
0424
0425
0426
0427
0428
0429
0430
0431
0432
0433
0434
0435
0436
0437
0438
0439
0440
0441
0442
0443
0444
0445
0446
0447
0448
0449
0450
0451
0452
0453
0454
0455
0456
0457
0458
0459
0460
0461
0462
0463
0464
0465
0466
0467
0468
0469
0470
0471
0472
0473
0474
0475
0476
0477
0478
0479
0480
0481
0482
0483
0484
0485
0486
0487
0488
0489
0490
0491
0492
0493
0494
0495
0496
0497
0498
0499
0500
0501
0502
0503
0504
0505
0506
0507
0508
0509
0510
0511
0512
0513
0514
0515
0516
0517
0518
0519
0520
0521
0522
0523
0524
0525
0526
0527
0528
0529
0530
0531
0532
0533
0534
0535
0536
0537
0538
0539
0540
0541
0542
0543
0544
0545
0546
0547
0548
0549
0550
0551
0552
0553
0554
0555
0556
0557
0558
0559
0560
0561
0562
0563
0564
0565
0566
0567
0568
0569
0570
0571
0572
0573
0574
0575
0576
0577
0578
0579
0580
0581
0582
0583
0584
0585
0586
0587
0588
0589
0590
0591
0592
0593
0594
0595
0596
0597
0598
0599
0600
0601
0602
0603
0604
0605
0606
0607
0608
0609
0610
0611
0612
0613
0614
0615
0616
0617
0618
0619
0620
0621
0622
0623
0624
0625
0626
0627
0628
0629
0630
0631
0632
0633
0634
0635
0636
0637
0638
0639
0640
0641
0642
0643
0644
0645
0646
0647
0648
0649
0650
0651
0652
0653
0654
0655
0656
0657
0658
0659
0660
0661
0662
0663
0664
0665
0666
0667
0668
0669
0670
0671
0672
0673
0674
0675
0676
0677
0678
0679
0680
0681
0682
0683
0684
0685
0686
0687
0688
0689
0690
0691
0692
0693
0694
0695
0696
0697
0698
0699
0700
0701
0702
0703
0704
0705
0706
0707
0708
0709
0710
0711
0712
0713
0714
0715
0716
0717
0718
0719
0720
0721
0722
0723
0724
0725
0726
0727
0728
0729
0730
0731
0732
0733
0734
0735
0736
0737
0738
0739
0740
0741
0742
0743
0744
0745
0746
0747
0748
0749
0750
0751
0752
0753
0754
0755
0756
0757
0758
0759
0760
0761
0762
0763
0764
0765
0766
0767
0768
0769
0770
0771
0772
0773
0774
0775
0776
0777
0778
0779
0780
0781
0782
0783
0784
0785
0786
0787
0788
0789
0790
0791
0792
0793
0794
0795
0796
0797
0798
0799
0800
0801
0802
0803
0804
0805
0806
0807
0808
0809
0810
0811
0812
0813
0814
0815
0816
0817
0818
0819
0820
0821
0822
0823
0824
0825
0826
0827
0828
0829
0830
0831
0832
0833
083
```

```

0863
0864
0865
0866
0867
0868
0869
0870
0871
0872
0873
0874
0875
0876
0877
0878
0879
0880
0881
0882
0883
0884
0885
0886
0887
0888
0889
0890
0891
0892
0893
0894
0895
0896
0897
0898
0899
0900
0901
0902
0903
0904
0905
0906
0907
0908
0909
0910

```

I - - - - -
 I*** TEST 361D ***
 ITEST 361D CHECKS THAT THE BUT ON SR<1:0> READS THE (052825) IN THE SR CORRECTLY
 6771:
 TEST361D:
 PO, LOAD=ENUA(ZTARGET403), IEXPECTED VALUE "01" IN SR<1:0>
 LOAD=ERROR(TEST361D), IERROR DIRECTORY KEY
 DCS=CTR(C3,), ICOMPARE ENUA;TNUA AT TARGET
 NEXT, J/GOBUT361D I
 (6771) DCS[1.00,1.0,0,0] BM[1100..00,11..11,00..000,0011...0,0,0..0...0.0000...0..0000,0...11,000...000,000,101]

6005: I(FREE)
 GORUT361D:
 SETUP, RETURN/TEST361E, IRETURN TO START OF NEXT SUBTEST
 NEXT, GOTO-PAGE(7), IBUT'S ARE ON PAGE 7
 J/BUTSR1=0 IGO BUY ON SR<1:0>
 (6005) DCS[0.00,0,0,0,0] BM[0110..00,11..11,10..111..111...0,0,0..0..0...0.0000...0..0000,0...11,100...011,001,101]

I - - - - -
 I*** TEST 361E ***
 ITEST 361E CHECKS THAT GD<3:2> IS CLEARED UNDER SR=LOAD, GUARD-ENABLED
 6767:
 TEST361E:
 PO, LOAD=ENUA(ZTARGET400), IEXPECTED VALUE "00" IN GUARD<3:2>
 LOAD=ERROR(TEST361E), IERROR DIRECTORY KEY
 DCS=CTR(C3,), ICOMPARE ENUA;TNUA AT TARGET
 BUMP=VERIFY, ICOUNT
 NEXT, J/GOBUT361E I
 (6767) DCS[1.00,1.0,0,1] BM[1100..00,11..11,00..000,000...0,0,0..0..0...0.0000...0..0000,0...11,000...000,000,110]

6006: I(FREE)
 GORUT361E:
 SETUP, RETURN/TEST362A, IRETURN TO START OF NEXT SUBTEST
 NEXT, GOTO-PAGE(7), IBUT'S ARE ON PAGE 7
 J/BUTGD3=2 IGO BUY ON GD<3:2>
 (6006) DCS[0.00,0,0,0,0] BM[0110..00,11..11,10..101..111...0,0,0..0..0...0.0000...0..0000,0...11,100...011,001,100]

```

0911      I
0912
0913      *** TEST 362A ***
0914      !TEST-362A CHECKS THAT THE SRGD CAN BE SHIFTED RIGHT 1 POSITION, INSERTING
0915      !A "1" FROM D<00> INTO SR<15>, SHIFTING THE "1" IN SR<00> INTO GD<3>
0916      47651
0917      TEST362A:
0918          P0,          LOAD=ENUA(ZTARGET434),          !SETUP FOR IR=(000000)/INSTR5 TEST
0919          LOAD=ERROR(TEST362A),          !ERROR DIRECTORY KEY
0920          DCS=CTRC(C10,),          !COMPARE ENUA:THUA AT TARGET
0921          NEXT,        J/SETRES362A          !
(6765) DCS[1.00.1.0.0.0] BM[0101.00.11.11.00.011.100...0.0.0.0.0...0.0000...0.0000.0...11.000...000.000,111]
0922
0923      6007: 1(FREE)
0924      SFTRES362A:
0925          P0,          BUMP=VERIFY,          !COUNT
0926          P3,          CSPD[16]_EMIT,          !CSP GETS
0927          EMITC,        SENDMUX=4567=SEL,          !RES REG VALUES
0928          SR=RIGHT, GUARD=EN,          !
0929          NEXT,        J/SETDDC362A          !
(4007) DCS[0.00.0.0.0.1] BM[0010.10.10.00.00.000.000...0.0.0.0.0...0.0001...1.0000.0...11.000...000.001,000]
0930
0931      6010: 1(FREE)
0932      SETDDC362A:
0933          P2-T,        D_BSPhi(C000001), DIC1_0,          !SETUP D<00>=1 FOR SHIFT
0934          NEXT,        J/LOADRES362A          !
(6010) DCS[0.00.0.0.0.0] BM[1010.01.11.00.00.000.000...0.0.1.0.0.0...0.0000...0.0000.0...11.000...000.001,001]
0935
0936      6011: 1(FREE)
0937      LOADRES362A:
0938          P2,          RES_CSPR(B16),          !STORE RES
0939          P3-T,        SR_SR=RIGHT-1,          !SHIFT SR RIGHT, GUARD=ENABLED
0940          NEXT,        J/COMP362A          !
(6011) DCS[0.00.0.0.0.0] BM[0000.11.01.00.00.000.000...1.0.1.0.0.0...0.0000...0.1000.1...11.000...000.001,010]
0941
0942      6012: 1(FREE)
0943      COMP362A:
0944          P0,          BUMP=VERIFY,          !COUNT
0945          P2-T,        D_SR=XOR-BSPHi(C125252), SAVE=DIC1,          !D = (125252)=SR, BITWISE
0946          NEXT,        J/GOBUT362A          !
(6012) DCS[0.00.0.0.0.1] BM[0110.01.11.00.00.110.111...0.0.1.0.0.0...0.0000...0.0000.0...11.000...000.001,011]
0947
0948      6013: 1(FREE)
0949      GOBUT362A:
0950          SFTUP,        RETURN/TEST362B,          !RETURN TO START OF NEXT SUBTEST
0951          NEXT,        CALL(DINTDIR-S)          !SUBR: D -> IR, BUT(INSTR5)
(6013) DCS[0.00.0.0.0.0] BM[0110.00.11.11.10.011.111...0.0.0.0.0...0.0000...0.0000.0...11.100...010.111,011]
0952
0953      I
0954
0955
0956
0957

```

```

8958
8959 1*** TEST 362B ***
8960 1TEST-362B CHECKS THAT SR-XMUX-FLOAT PORT CAN BE READ, WITH SR=(125252)
8961 6763:
8962 TEST362B:
8963 PO, LOAD-ENUA(ZTARGET434), 1SETUP FOR IR=(000000)/INSTR5 TEST
8964 LOAD-ERROR(TEST362B), 1ERROR DIRECTORY KEY
8965 DCS-CTR(C9.), 1COMPARE ENUA:TNUA AT TARGET
8966 NEXT, J/EXPEC362B
(6763) DCS[1.00.1.0.0.0] BM[0111.00.11.11.00.011.100...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.001.100]
8967 6014: 1(FREE)
8968 EXPEC362B:
8969 PO, BUMP-VERIFY, 1COUNT
8970 P3, CSPD[16]_EMIT, 1EXPECTED VALUE OF XMUX-FLOAT
8971 CSPT[000052, 1(000052)
8972 NEXT, J/COMP362B
(6014) DCS[0.00.0.0.0.1] BM[0000.10.00.00.00.101.010...0.0.0.0.0.0.0.0001...1.0000.0...11.000...000.001.101]
8974 6015: 1(FREE)
8975 COMP362B:
8976 SETUP, RETURN/TEST362D, 1RETURN TO START OF NEXT SUBTEST
8977 NEXT, CALL(CSP16XORFLTTOIR-S) 1SUBR: CSP(16),XOR,FLOAT -> IR, BUT(INSTR5)
8978 (6015) DCS[0.00.0.0.0.0] BM[0110.00.11.11.01.111.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...000.011.101]
8979
8980
8981
8982
8983
8984 1 - - - - -
8985
8986 1*** TEST 362D ***
8987 1TEST-362D CHECKS THAT THE BUT ON SR<110> READS THE (125252) IN THE SR CORRECTLY
8988 6757:
8989 TEST362D:
8990 PO, LOAD-ENUA(ZTARGET405), 1EXPECTED VALUE "10" IN SR<110>
8991 LOAD-ERROR(TEST362D), 1ERROR DIRECTORY KEY
8992 DCS-CTR(C3.), 1COMPARE ENUA:TNUA AT TARGET
8993 NEXT, J/GOBUT362D
(6757) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.000.101...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.001.110]
8994 6016: 1(FREE)
8995 GOBUT362D:
8996 SETUP, RETURN/TEST362E, 1RETURN TO START OF NEXT SUBTEST
8997 NEXT, GOTO-PAGE(7), 1BUT'S ARE ON PAGE 7
8998 J/BUTSR1-0 1GO BUT ON SR<110>
8999 (6016) DCS[0.00.0.0.0.0] BM[0110.00.11.11.01.101.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.001.101]
9000
9001
9002
9003
9004

```

```

9005 1 - - - - -
9006
9007 1*** TEST 362E ***
9008 1TEST-362E CHECKS THAT GD<312> RECEIVED THE "1" IN SR<00> AFTER THE SHIFT RIGHT
9009 6755:
9010 TEST362E:
9011 PO, LOAD-ENUA(ZTARGET402), 1EXPECTED VALUE "10" IN GUARD<312>
9012 LOAD-ERROR(TEST362E), 1ERROR DIRECTORY KEY
9013 DCS-CTR(C3.), 1COMPARE ENUA:TNUA AT TARGET
9014 NEXT, J/GOBUT362E
(6755) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.000.010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.001.111]
9015 6017: 1(FREE)
9016 GOBUT362E:
9017 SETUP, RETURN/TEST363A, 1RETURN TO START OF NEXT SUBTEST
9018 PO, BUMP-VERIFY, 1COUNT
9019 NEXT, GOTO-PAGE(7), 1BUT'S ARE ON PAGE 7
9020 J/BUTGD3-2 1GO BUT ON GD<312>
(6017) DCS[0.00.0.0.0.1] BM[0110.00.11.11.01.011.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.001.100]
9022
9023
9024
9025
9026
9027 1 - - - - -
9028
9029 1*** TEST 363A ***
9030 1TEST-363A CHECKS THAT THE SR&GD CAN AGAIN BE SHIFTED RIGHT 1 POSITION, INSERTING
9031 1A "0" FROM D<00> INTO SR<15>, SHIFTING THE "0" IN SR<00> INTO GD<3>
9032 6753:
9033 TEST363A:
9034 PO, LOAD-ENUA(ZTARGET434), 1SETUP FOR IR=(000000)/INSTR5 TEST
9035 LOAD-ERROR(TEST363A), 1ERROR DIRECTORY KEY
9036 DCS-CTR(C9.), 1COMPARE ENUA:TNUA AT TARGET
9037 NEXT, J/SETDDC363A
(6753) DCS[1.00.1.0.0.0] BM[0110.00.11.11.00.011.100...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.010.000]
9038 6020: 1(FREE)
9039 SETDDC363A:
9040 P3-T, D_NOT-ASPHI(C000001), D(C)-1, 1SETUP D=(177776), D(C)=0 FOR SHIFT RIGHT
9041 NEXT, J/SHIFT363A
(6020) DCS[0.00.0.0.0.0] BM[0000.00.00.11.01.000.000...1.1.0.0.0.0.0.0000...0.0000.0...11.000...000.010.001]
9043 6021: 1(FREE)
9044 SHIFT363A:
9045 PO, BUMP-VERIFY, 1COUNT
9046 P2-T, SR_SR=RIGHT-1, 1SHIFT SR RIGHT, GUARD-ENABLED
9047 NEXT, J/COMP363A
(6021) DCS[0.00.0.0.0.1] BM[0000.00.00.00.00.000.000...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.010.010]
9049 6022: 1(FREE)
9050 COMP363A:
9051 P2-T, D_SR=XOR-ASPHI(C052525), SAVE-D(C), 1D = (052525)=SR, BITWISE
9052

```

```
9053      NEXT, J/GOBUT363A
(6022) DCS[0,00,0,0,0,0] BM[0110,01,11,00,00,111,111,0,1,0,0,0,0,0,0000,0,0,0000,0,11,000,0,000,010,011]
9054
9055      6023: 1(FREE)
9056      GOBUT363A:
9057      SETUP, RETURN/TEST363B, RETURN TO START OF NEXT SUBTEST
9058      NEXT, CALL[DINTDIR=5] ISUBR; D => IN, BUT(INSTR6)
(6023) DCS[0,00,0,0,0,0] BM[0110,00,11,11,01,001,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,0,010,111,011]
9059
9060
9061
9062
9063
9064      1 - - - - -
9065
9066      1*** TEST 363B ***
9067      1TEST-363B CHECKS THE GUARD REGISTER WAS SHIFTED RIGHT ALSO
9068      6751:
9069      TEST363B:
9070      PD, LOAD-ENUA(ZTARGET401), EXPECTED VALUE "01" IN GUARD<3:2>
9071      LOAD-ERROR(TEST363B), ERROR DIRECTORY KEY
9072      DCS-CTR(C3), COMPARE ENUA;TWUA AT TARGET
9073      NEXT, J/GOBUT363B
(6751) DCS[1,00,1,0,0,0] BM[1100,00,11,11,00,000,001,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,0,000,010,100]
9074
9075      6024: 1(FREE)
9076      GOBUT363B:
9077      SETUP, RETURN/TEST364A, RETURN TO START OF NEXT SUBTEST
9078      NEXT, GOTO-PAGE(7), BUT'S ARE ON PAGE 7
9079      J/BUTGD3=2 GO BUT ON GD<3:2>
(6024) DCS[0,00,0,0,0,0] BM[0110,00,11,11,00,111,111,0,0,0,0,0,0,0,0000,0,0,0000,0,11,100,0,011,001,100]
9080
9081
9082
9083
9084
9085      1 - - - - -
9086
9087      1*** TEST 364A ***
9088      1TEST-364A/364B CHECK THAT WE ARE ABLE TO FILTER A "1" DOWN TO GD<0>, AND
9089      1SUBSEQUENTLY ABLE TO RECOVER IT, SO GD<3:0>="0101"
9090      6747:
9091      TEST364A:
9092      PD, LOAD-ENUA(ZTARGET402), EXPECTED VALUE "10" IN GUARD <3:2>
9093      LOAD-ERROR(TEST364A), ERROR DIRECTORY KEY
9094      DCS-CTR(C4), COMPARE ENUA;TWUA AT TARGET
9095      BUMP-VERIFY, COUNT
9096      NEXT, J/SETDDC364A
(6747) DCS[1,00,1,0,0,1] BM[1011,00,11,11,00,000,010,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,0,000,010,101]
9097
9098      6025: 1(FREE)
9099      SETDDC364A:
```

```
9100      P2-T, D_NOT=BSPHI(C000001), D[C]=1, 1SETUP D=(177776), D[C]=0 FOR SHIFT RIGHT
9101      NEXT, J/GOBUT364A 1 WHERE SR<15> <= (0)
(6025) DCS[0,00,0,0,0,0] BM[0000,00,00,11,01,000,000,0,0,1,0,0,0,0,0,0000,0,0,0000,0,11,000,0,000,010,111]
9102
9103      6026: 1(FREE)
9104      GOBUT364A:
9105      SETUP, RETURN/TEST364B, RETURN TO START OF NEXT SUBTEST
9106      P3-T, SR_SR=RIGHT-1, 1SHIFT SR&GD RIGHT
9107      NEXT, GOTO-PAGE(7), BUT'S ARE ON PAGE 7
9108      J/BUTGD3=2 GO BUT ON GD<3:2>
(6026) DCS[0,00,0,0,0,0] BM[0110,00,11,11,00,101,111,1,0,1,0,0,0,0,0,0000,0,0,0000,0,11,100,0,011,001,100]
9109
9110
9111
9112
9113
9114      1 - - - - -
9115
9116      1*** TEST 364B ***
9117      1TEST-364B CHECKS THAT GD<3:2> READS AS "01"
9118      6745:
9119      TEST364B:
9120      PD, LOAD-ENUA(ZTARGET401), EXPECTED VALUE "01" IN GUARD<3:2>
9121      LOAD-ERROR(TEST364B), ERROR DIRECTORY KEY
9122      DCS-CTR(C4), COMPARE ENUA;TWUA AT TARGET
9123      NEXT, J/SETDDC364A
(6745) DCS[1,00,1,0,0,0] BM[1011,00,11,11,00,000,001,0,0,0,0,0,0,0,0000,0,0,0000,0,11,000,0,000,010,111]
9124
9125      6027: 1(FREE)
9126      SETDDC364B:
9127      P3-T, D_BSPHI(C000001), D[C]=0, 1SETUP D, D[C] FOR SR<15>="1"
9128      NEXT, J/GOBUT364B
(6027) DCS[0,00,0,0,0,0] BM[0110,01,11,00,00,000,000,1,1,0,0,0,0,0,0,0000,0,0,0000,0,11,000,0,000,011,000]
9129
9130      6030: 1(FREE)
9131      GOBUT364B:
9132      SETUP, RETURN/TEST365A, RETURN TO START OF NEXT SUBTEST
9133      P2-T, SR_SR=RIGHT-1, 1SHIFT SR&GD RIGHT AGAIN
9134      NEXT, GOTO-PAGE(7), BUT'S ARE ON PAGE 7
9135      J/BUTGD3=2 GO BUT ON GD<3:2>
(6030) DCS[0,00,0,0,0,0] BM[0110,00,11,11,00,011,111,0,0,1,0,0,0,0,0,0000,0,0,0000,0,11,100,0,011,001,100]
9136
9137
9138
9139
9140
9141      1 - - - - -
9142
9143      1*** TEST 365A ***
9144      1TEST-365A CHECKS THAT CAN SHIFT SR=RIGHT, W/GUARD-DISABLED
9145      6743:
9146      TEST365A:
```

```

9147      PO,      LOAD-ENUA(ZTARGET434),      |SETUP FOR IR=(000000)/INSTR5 TEST
9148      LOAD-ERROR(TESTR365A),      |ERROR DIRECTORY KEY
9149      DCS-CTR(C10.),      |COMPARE ENUA:TNUA AT TARGET
9150      NEXT,      J/SETRES365A
(6743) DCS[1.00,1.0,0,0] BM[0101..00.11..11.00..011..100...0.0,0..0..0.0000...0..0000,0...11,000...000,011,001]

9151      6031: 1(FREE)
9152      SETRES365A:
9153      PO,      BUMP-VERIFY,      |COUNT
9154      P3,      CSPD[16]_EMIT,      |CSP GETS
9155      EMITC,      SENDMUX-4567-SEL,      |RES REG VALUES
9156      SR-RIGHT, GUARD-DIS,
9157      J/LOADRES365A
(6031) DCS[0.00,0.0,0,0.1] BM[0010..10.00..00.00..000..000...0.0,0..0..0.0001...1..0000,0...11,000...000,011,010]

9159      6032: 1(FREE)
9160      LOADRES365A:
9161      P2,      RES_CSPB(B16),      |STORE RES
9162      P3-T,      SR-SR-RIGHT-1,      |SHIFT SR RIGHT, GUARD-DISABLED
9163      NFXT,      J/EXPEC365A
(6032) DCS[0.00,0.0,0,0.0] BM[0000..11.01..00.00..000..000...1,0,1..0..0.0000...0..1000,1...11,000...000,011,011]

9165      6033: 1(FREE)
9166      EXPEC365A:
9167      PO,      BUMP-VERIFY,      |COUNT
9168      P3,      CSPD[16]_EMIT,      |EXPECTED VALUE OUT OF XMUX-SR
9169      EMIT/145252,      |145252)
9170      NEXT,      J/COMP365A
(6033) DCS[0.00,0.0,0,0.1] BM[1100..10.10..10.10..101..010...0.0,0..0..0.0001...1..0000,0...11,000...000,011,100]

9172      6034: 1(FREE)
9173      COMP365A:
9174      SETUP,      RETURN/TEST365B,      |RETURN TO START OF NEXT SUBTEST
9175      NEXT,      CALL(CSPI6XORSRT0IR-5)      |SUBR: CSP(16),XOR,SR -> IR, BUT(INSTR5)
9176      (6034) DCS[0.00,0.0,0,0.0] BM[0110..00.11..11.00..001..111...0.0,0..0..0.0000...0..0000,0...11,100...000,011,001]

9177
9178
9179
9180
9181
9182
9183
9184      | - - - - -
9185      |*** TEST 365B ***
9186      |TEST-365B CHECKS THAT THE GUARD WASN'T ALTERED ON SHIFT RIGHT/DISABLED
9187      6741:
9188      TEST365B:
9189      PO,      LOAD-ENUA(ZTARGET401),      |EXPECTED VALUE "01" IN GUARD<3:2>
9190      LOAD-ERROR(TESTR365B),      |ERROR DIRECTORY KEY
9191      DCS-CTR(C3.),      |COMPARE ENUA:TNUA AT TARGET
9192      NFXT,      J/GOBUT365B
(6741) DCS[1.00,1.0,0,0] BM[1100..00.11..11.00..000..001...0.0,0..0..0.0000...0..0000,0...11,000...000,011,101]

9192
9193      6035: 1(FREE)

```

```

9194      GOBUT365B:
9195      SETUP,      RETURN/TEST366A,      |RETURN TO START OF NEXT SUBTEST
9196      NFXT,      GOTO-PAGE(7),      |BUT'S ARE ON PAGE 7
9197      J/BUTGD3-2      |GO BUT ON GD<3:2>
(6035) DCS[0.00,0.0,0,0.0] BM[0110..00.11..10.11..111..111...0.0,0..0..0.0000...0..0000,0...11,100...011,001,100]

9199
9200
9201
9202
9203
9204
9205      | - - - - -
9206      |*** TEST 366A ***
9207      |TEST-366A CHECKS THAT SR CAN SHIFT LEFT, GUARD-DISABLED & NOT ALTERED, "0" IN
9208      |D(C) SHIFTED INTO SR<00>
9209      6737:
9210      TEST366A:
9211      PO,      LOAD-ENUA(ZTARGET434),      |SETUP FOR IR=(000000)/INSTR5 TEST
9212      LOAD-ERROR(TESTR366A),      |ERROR DIRECTORY KEY
9213      DCS-CTR(C11.),      |COMPARE ENUA:TNUA AT TARGET
9214      NFXT,      J/SETRES366A
(6737) DCS[1.00,1.0,0,0] BM[0100..00.11..11.00..011..100...0.0,0..0..0.0000...0..0000,0...11,000...000,011,110]

9216      6036: 1(FREE)
9217      SETRES366A:
9218      PO,      BUMP-VERIFY,      |COUNT
9219      P3,      CSPD[16]_EMIT,      |CSP GETS
9220      EMITC,      SENDMUX-4567-SEL,      |RES REG VALUES
9221      SR-LEFT, GUARD-DIS,
9222      J/LOADRES366A
(6036) DCS[0.00,0.0,0,0.1] BM[0001..10.00..00.00..000..000...0.0,0..0..0.0001...1..0000,0...11,000...000,011,111]

9224      6037: 1(FREE)
9225      LOADRES366A:
9226      P2-T,      RES_CSPB(B16),      |STORE RES
9227      D_ZERO, D(C)_ALU07,      |AND SET D, D(C)
9228      NFXT,      J/SHIFT366A
(6037) DCS[0.00,0.0,0,0.0] BM[0011..11.01..00.00..000..011...0.1,0..0..0.0000...0..1000,1...11,000...000,100,000]

9229      6040: 1(FREE)
9230      SHIFT366A:
9231      PO,      BUMP-VERIFY,      |COUNT
9232      P3-T,      SR-SR-LEFT-1,      |SHIFT SR LEFT 1, SR<00>_D(C)="0"
9233      NFXT,      J/EXPEC366A
(6040) DCS[0.00,0.0,0,0.1] BM[0000..00.00..00.00..000..000...1,0,1..0..0.0000...0..0000,0...11,000...000,100,001]

9235      6041: 1(FREE)
9236      EXPEC366A:
9237      P3,      CSPD[16]_EMIT,      |EXPECTED RESULT AFTER SHIFT
9238      EMIT/112524,      |112524)
9239      NFXT,      J/COMP366A
(6041) DCS[0.00,0.0,0,0.0] BM[1001..10.01..01.01..010..100...0.0,0..0..0.0001...1..0000,0...11,000...000,100,010]

9240

```

```
9241 6042: 1(FREE)
9242 COMP366A:
9243 SETUP, RETURN/TEST366B, RETURN TO START OF NEXT SUBTEST
9244 NEXT, CALL(CSP16XORSRTOIR-5) SUBR: CSP(16),XOR,SR -> IR, BUT(INSTRS)
(6042) DCS[0,00,0,0,0,0] BM[0110,00,11,10,11,101,111...0,0,0,0,0,0,0,0000...0,0000,0...11,100...000,011,001]

9245
9246
9247
9248
9249
9250
9251 |-----|
9252
9253 |*** TEST 366B ***
9254 |TEST-366B CHECKS THAT GUARD WASN'T ALTERED ON SHIFT LEFT/GUARD-DISABLED
9255 6735:
9256 TEST366B:
9257 PO, LOAD-ENUA(ZTARGET401), IGUARD SHOULD STILL BE "01"-"01"
9258 LOAD-ERROR(TEST366B), ERROR DIRECTORY KEY
9259 DCS-CTR(C3,), ICOMPARE ENUA:TNUA AT TARGET
9260 NEXT, J/GOBUT366B
(6735) DCS[1,00,1,0,0,0] BM[1100,00,11,11,00,000,001...0,0,0,0,0,0,0,0000...0,0000,0...11,000...000,100,011]

9261
9262 6043: 1(FREE)
9263 GOBUT366B:
9264 SETUP, RETURN/TEST366C, RETURN TO START OF NEXT SUBTEST
9265 NEXT, GOTO-PAGE(7), BUT'S ARE ON PAGE 7
9266 J/BUTGD3-2 100 BUT ON GUARD 4312>
(6043) DCS[0,00,0,0,0,0] BM[0110,00,11,10,11,011,111...0,0,0,0,0,0,0,0000...0,0000,0...11,100...011,001,100]

9267
9268
9269
9270
9271
9272
9273 |-----|
9274
9275 |*** TEST 366C ***
9276 |TEST-366C CHECKS THAT SR CAN AGAIN SHIFT LEFT, GUARD-DISABLED, SR<00>_D[C]="1"
9277 6731:
9278 TEST366C:
9279 PO, LOAD-ENUA(ZTARGET434), ISETUP FOR IR=(000000)/INSTRS TEST
9280 LOAD-ERROR(TEST366C), ERROR DIRECTORY KEY
9281 DCS-CTR(C10,), ICOMPARE ENUA:TNUA AT TARGET
9282 NEXT, J/SETDDC366C
(6733) DCS[1,00,1,0,0,0] BM[0101,00,11,11,00,011,100...0,0,0,0,0,0,0,0000...0,0000,0...11,000...000,100,100]

9283
9284 6044: 1(FREE)
9285 SETDDC366C:
9286 P2-T, D_ASPhi(C177777), D[C]_1, ISETUP D, D[C]
9287 NEXT, J/SHIFT366C
(6044) DCS[0,00,0,0,0,0] BM[1111,00,00,11,01,101,000...0,1,0,0,0,0,0,0000...0,0000,0...11,000...000,100,101]

9288
9289
9290
9291
9292
9293
9294
9295
9296
9297
9298
9299
9300
9301
9302
9303
9304
9305
9306
9307
9308
9309
9310
9311 |-----|
9312
9313 |*** TEST 367A ***
9314 |TEST-367A CHECKS THAT LOAD/GUARD-DISABLED LOADS SR, GUARD NOT ALTERED
9315 |CHECK THAT BUTA(CLEAR-FLAGS) CLEARS RES TO SR-LOAD/GUARD-DIS
9316 6731:
9317 TEST367A:
9318 PO, LOAD-ENUA(ZTARGET401), IGUARD SHOULD STILL BE "01"-"01" AFTER LOAD
9319 LOAD-ERROR(TEST367A), ERROR DIRECTORY KEY
9320 DCS-CTR(C6,), ICOMPARE ENUA:TNUA AT TARGET
9321 NEXT, J/SETRES367A
(6731) DCS[1,00,1,0,0,0] BM[1001,00,11,11,00,000,001...0,0,0,0,0,0,0,0000...0,0000,0...11,000...000,101,000]

9322
9323 6050: 1(FREE)
9324 SETRES367A:
9325 PO, BUMP-VERIFY, ICOUNT
9326 P3, CSPD[16]_EMIT, ICSP GETS
9327 FWITC, SENDMUX-4567-SEL, IRES VALUES
9328 SR-NOP, GUARD-EN, IFIRST LOAD COMPLEMENT OF THOSE AFTER BUTA(CLEAR-FLAGS)
9329 NEXT, J/LOADRES367A
(6050) DCS[0,00,0,0,0,1] BM[0011,10,10,00,00,000,000...0,0,0,0,0,0,0,0001...1,0000,0...11,000...000,101,001]

9330
9331 6051: 1(FREE)
9332 LOADRES367A:
9333 PO, BUMP-VERIFY, ICOUNT
9334 P2-T, RES_CSPB(R16), ISTORE RES
9335 D_ZERO, D[C]_ALU07, ISET D, D[C]
```

```
9288
9289 6045: 1(FREE)
9290 SHIFT366C:
9291 P2-T, SR_SR-LEFT-1, ISHIFT SR LEFT 1, SR<00>_D[C]="1"
9292 NEXT, J/EXPEC366C
(6045) DCS[0,00,0,0,0,0] BM[0000,00,00,00,00,000,000...0,0,1,0,0,0,0,0000...0,0000,0...11,000...000,100,110]

9293
9294 6046: 1(FREE)
9295 EXPEC366C:
9296 PO, BUMP-VERIFY, ICOUNT
9297 P3, CSPD[16]_EMIT, IEXPECTED VALUE AFTER SHIFT
9298 FWITC, EMIT/025251, I(025251)
9299 NEXT, J/COMP366C
(6046) DCS[0,00,0,0,0,1] BM[0010,10,10,10,10,101,001...0,0,0,0,0,0,0,0001...1,0000,0...11,000...000,100,111]

9300
9301 6047: 1(FREE)
9302 COMP366C:
9303 SETUP, RETURN/TEST367A, RETURN TO START OF NEXT SUBTEST
9304 NEXT, CALL(CSP16XORSRTOIR-5) SUBR: CSP(16),XOR,SR -> IR, BUT(INSTRS)
(6047) DCS[0,00,0,0,0,0] BM[0110,00,11,10,11,001,111...0,0,0,0,0,0,0,0000...0,0000,0...11,100...000,011,001]

9305
9306
9307
9308
9309
9310
9311 |-----|
9312
9313 |*** TEST 367A ***
9314 |TEST-367A CHECKS THAT LOAD/GUARD-DISABLED LOADS SR, GUARD NOT ALTERED
9315 |CHECK THAT BUTA(CLEAR-FLAGS) CLEARS RES TO SR-LOAD/GUARD-DIS
9316 6731:
9317 TEST367A:
9318 PO, LOAD-ENUA(ZTARGET401), IGUARD SHOULD STILL BE "01"-"01" AFTER LOAD
9319 LOAD-ERROR(TEST367A), ERROR DIRECTORY KEY
9320 DCS-CTR(C6,), ICOMPARE ENUA:TNUA AT TARGET
9321 NEXT, J/SETRES367A
(6731) DCS[1,00,1,0,0,0] BM[1001,00,11,11,00,000,001...0,0,0,0,0,0,0,0000...0,0000,0...11,000...000,101,000]

9322
9323 6050: 1(FREE)
9324 SETRES367A:
9325 PO, BUMP-VERIFY, ICOUNT
9326 P3, CSPD[16]_EMIT, ICSP GETS
9327 FWITC, SENDMUX-4567-SEL, IRES VALUES
9328 SR-NOP, GUARD-EN, IFIRST LOAD COMPLEMENT OF THOSE AFTER BUTA(CLEAR-FLAGS)
9329 NEXT, J/LOADRES367A
(6050) DCS[0,00,0,0,0,1] BM[0011,10,10,00,00,000,000...0,0,0,0,0,0,0,0001...1,0000,0...11,000...000,101,001]

9330
9331 6051: 1(FREE)
9332 LOADRES367A:
9333 PO, BUMP-VERIFY, ICOUNT
9334 P2-T, RES_CSPB(R16), ISTORE RES
9335 D_ZERO, D[C]_ALU07, ISET D, D[C]
```

```
9336      P3,      BUTA(CLP-FLAG-RES-UCON),      RESET RES TO SR-LOAD, GUARD-DIS
9337      NEXT,     J/LOADSR367A
(6051) DCS[0.00,0.0,0.0,1] BM[0011..11.01..00.00..0000..011..0.1.0.0..0.0000...0..1000.1...11.010...000.101.010]
9338
9339      6052: 1(FREE)
9340      LOADSR367A:
9341      P2-T,      SR_BSPHI(C052525),      SR_(052525)
9342      NEXT,     J/GOBUT367B
(6052) DCS[0.00,0.0,0.0,0] BM[1010..01.11..00.00..111..000...0.0.1.0.0..0.0000...0..0000.0...11.000...000.101.011]
9343
9344      6053: 1(FREE)
9345      GOBUT367B:
9346      SETUP,     RETURN/TEST370A,      RETURN TO START OF NEXT SUBTEST
9347      NEXT,     GOTO-PAGE(7),      BUT'S ARE ON PAGE 7
9348      J/BUTGD3-2      GO BUT ON GUARD <3:2>
(6053) DCS[0.00,0.0,0.0,0] BM[1010..00.11..10.10..101..111...0.0.0.0..0.0000...0..0000.0...11.100...011.001.100]
9349
9350
9351
9352
9353
9354
9355      I - - - - -
9356
9357      I*** TEST 370A ***
9358      ITEST-370A CHECKS THAT SHIFT LEFT/GUARD-ENABLED SHIFTS GD<3>="0" INTO SR<00>
9359      6725:
9360      TEST370A:
9361      P0,      LOAD-ENUA(ZTARGET434),      ISETUP FOR IR(000000)/INSTRS TEST
9362      LOAD-ERROR(TEST370A),      ERROR DIRECTORY KEY
9363      DCS-CTR(C10.),      ICOMPARE ENUA:TWUA AT TARGET
9364      NEXT,     J/SETRES370A
(6725) DCS[1.00,1.0,0.0,0] BM[1011..00.11..11.00..011..100...0.0.0.0..0.0000...0..0000.0...11.000...000.101.100]
9365
9366      6054: 1(FREE)
9367      SETRES370A:
9368      P0,      BUMP-VERIFY,      ICOUNT
9369      P3,      CSPD[16]-EMIT,      ICSP GETS
9370      EMITC,     SENDMUX-4567-SEL,      IRES VALUES
9371      SR-LEFT, GUARD-EN,
9372      NEXT,     J/LOADRES370A
(6054) DCS[0.00,0.0,0.0,1] BM[0001..10.10..00.00..0000..000...0.0.0.0..0.0001...1..0000.0...11.000...000.101.101]
9373
9374      6055: 1(FREE)
9375      LOADRES370A:
9376      P2-T,      RES_CSPB(B16),      ISTORE RES
9377      D_BSPHI(C000000), D[C]-1,      ISETUP D, D[C]
9378      NEXT,     J/SHIFT370A
(6055) DCS[0.00,0.0,0.0,0] BM[1111..11.01..11.01..100..000...0.1.0.0..0.0000...0..1000.1...11.000...000.101.110]
9379
9380      6056: 1(FREE)
9381      SHIFT370A:
9382      P2-T,      SR_SP-LEFT=1,      ISHIFT SR LEFT 1, SR<00>_GD<3>="0"
```

```
9383      NEXT,     J/COMP370A
(6056) DCS[0.00,0.0,0.0,0] BM[0000..00.00..00.00..0000..000...0.0.1.0.0..0.0000...0..0000.0...11.000...000.101.111]
9384
9385      6057: 1(FREE)
9386      COMP370A:
9387      P2-T,      D_SR-XOR-BSPHI(C125252), SAVE-DIC],      10 - (125252)=SR, BITWISE
9388      NEXT,     J/GOBUT370A
(6057) DCS[0.00,0.0,0.0,0] BM[1010..01.11..00.00..110..111...0.1.0.0..0.0000...0..0000.0...11.000...000.110.000]
9389
9390      6060: 1(FREE)
9391      GOBUT370A:
9392      SETUP,     RETURN/TEST370B,      RETURN TO START OF NEXT SUBTEST
9393      NEXT,     CALL(DINTOIR-5]      ISUBR; D -> IR, BUT(INSTR5)
(6060) DCS[0.00,0.0,0.0,0] BM[0110..00.11..10.10..011..111...0.0.0.0..0.0000...0..0000.0...11.100...010.111.011]
9394
9395
9396
9397
9398
9399
9400      I - - - - -
9401
9402      I*** TEST 370B ***
9403      ITEST-370B CHECKS THAT THE GUARD, NOW ENABLED, WAS ALSO SHIFTED LEFT
9404      6723:
9405      TEST370B:
9406      P0,      LOAD-ENUA(ZTARGET402),      IGUARD IS NOW "10"="10"
9407      LOAD-ERROR(TEST370B),      ERROR DIRECTORY KEY
9408      DCS-CTR(C3.),      ICOMPARE ENUA:TWUA AT TARGET
9409      NEXT,     J/GOBUT370B
(6723) DCS[1.00,1.0,0.0,0] BM[1100..00.11..11.00..0000..010...0.0.0.0..0.0000...0..0000.0...11.000...000.110.001]
9410
9411      6061: 1(FREE)
9412      GOBUT370B:
9413      SETUP,     RETURN/TEST370C,      RETURN TO START OF NEXT SUBTEST
9414      P0,      BUMP-VERIFY,      ICOUNT
9415      NEXT,     GOTO-PAGE(7),      BUT'S ARE ON PAGE 7
9416      J/RUTGD3-2      GO BUT ON GUARD <3:2>
(6061) DCS[0.00,0.0,0.0,1] BM[0110..00.11..10.10..001..111...0.0.0.0..0.0000...0..0000.0...11.100...011.001.100]
9417
9418
9419
9420
9421
9422
9423
9424      I - - - - -
9425
9426      I*** TEST 370C ***
9427      ITEST-370C CHECKS THAT SHIFT LEFT/GUARD-ENABLED SHIFTS GD<3>="1" INTO SR<00>
9428      6721:
9429      TEST370C:
```

```
9430      PO,      LOAD-ENUA(ZTARGET434),      !SETUP FOR IR=(000000)/INSTRS TEST
9431      LOAD-ERROR(TEST370C),      !ERROR DIRECTORY KEY
9432      DCS-CTR(C9.),      !COMPARE ENUA;TNUA AT TARGET
9433      NEXT,      J/SETDDC370C
(6721) DCS[1.00.1.0.0.0] BM[0110.00.11.11.00.011.100...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.110.010]
9434
9435      6062: I(FREE)
9436      SETDDC370C:
9437      PO,      RUMP-VERIFY,      !COUNT
9438      P2-T,      D_ZERO, DIC=ALU07,      !SET D, DIC]
9439      NEXT,      J/SHIFT370C
(6062) DCS[0.00.0.0.0.1] BM[0011.00.00.00.00.000.011...0.1.0.0.0.0.0.0000...0.0000.0...11.000...000.110.011]
9440
9441      6063: I(FREE)
9442      SHIFT370C:
9443      P3-T,      SR-SR-LEFT-1,      !SHIFT SR LEFT, SR<00>_GD<3>="1"
9444      NEXT,      J/COMP370C
(6063) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000...1.0.1.0.0.0.0.0000...0.0000.0...11.000...000.110.100]
9445
9446      6064: I(FREE)
9447      COMP370C:
9448      P2-T,      D_SR=XOR-BSPHI(C052525), SAVE-DIC1,      ID = (052525)=SR, BITWISE
9449      NEXT,      J/GOBUT370C
(6064) DCS[0.00.0.0.0.0] BM[0110.01.11.00.00.111.111...0.1.0.0.0.0.0.0000...0.0000.0...11.000...000.110.101]
9450
9451      6065: I(FREE)
9452      GOBUT370C:
9453      SETUP,      RETURN/TEST370D,      !RETURN TO START OF NEXT SUBTEST
9454      NEXT,      CALL(DINVOIR=3)      !SUBR: D -> IR, BUT(SWSTRS)
(6065) DCS[0.00.0.0.0.0] BM[0110.00.11.10.01.111.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.111.011]
9455
9456
9457
9458
9459
9460
9461
9462
9463
9464      !*** TEST 370D ***
9465      !TEST-370D CHECKS THAT THE "1" THAT GOT PUT IN GD<0> CAN BE SHIFTED BACK
9466      6717:
9467      TEST370D:
9468      PO,      LOAD-ENUA(ZTARGET401),      !GUARD IS NOW "01"0"00"
9469      LOAD-ERROR(TEST370D),      !ERROR DIRECTORY KEY
9470      DCS-CTR(C3.),      !COMPARE ENUA;TNUA AT TARGET
9471      NEXT,      J/GOBUT370D
(6717) DCS[1.00.1.0.0.0] BM[1100.00.11.11.00.000.001...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.110.110]
9472
9473      6066: I(FREE)
9474      GOBUT370D:
9475      SETUP,      RETURN/TEST371A,      !RETURN TO START OF NEXT SUBTEST
9476      NEXT,      GOTO=PAGE(7),      !BUT'S ARE ON PAGE 7
```

```
9477      J/BUTGD3=2      !GO BUT ON GUARD <312>
(6066) DCS[0.00.0.0.0.0] BM[0110.00.11.10.01.101.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.001.100]
9478
9479
9480
9481
9482
9483
9484
9485
9486
9487      !*** TEST 371A ***
9488      !TEST-371A CHECKS THAT SR-NOP FUNCTION DOES NOTHING
9489      6715:
9490      TEST371A:
9491      PO,      LOAD-ENUA(ZTARGET434),      !SETUP FOR IR=(000000)/INSTRS TEST
9492      LOAD-ERROR(TEST371A),      !ERROR DIRECTORY KEY
9493      DCS-CTR(C10.),      !COMPARE ENUA;TNUA AT TARGET
9494      NEXT,      J/SETRES371A
(6715) DCS[1.00.1.0.0.0] BM[0101.00.11.11.00.011.100...0.0.0.0.0.0.0.0000...0.0000.0...11.000...000.110.111]
9495
9496      6067: I(FREE)
9497      SETRES371A:
9498      PO,      RUMP-VERIFY,      !COUNT
9499      P3,      CSPD[16]_EMIT,      !CSP GETS
9500      EMITC,      SENDMUX=4567-SEL,      !RES VALUES
9501      SR-NOP,      GUARD=EN,      !
9502      NEXT,      J/LOADRES371A
(6067) DCS[0.00.0.0.0.1] BM[0011.10.10.00.00.000.000...0.0.0.0.0.0.0.0001...1.0000.0...11.000...000.111.000]
9503
9504      6070: I(FREE)
9505      LOADRES371A:
9506      P2-T,      RES_CSPB(816),      !STORE RES
9507      D_ASPI(C000000), DIC=1,      !SETUP D, DIC]
9508      NEXT,      J/SHIFT371A
(6070) DCS[0.00.0.0.0.0] BM[1111.11.01.11.01.100.000...0.1.0.0.0.0.0.0000...0.1000.1...11.000...000.111.001]
9509
9510      6071: I(FREE)
9511      SHIFT371A:
9512      PO,      RUMP-VERIFY,      !COUNT
9513      P2-T,      CLK-SR,      !DO AN SR-NOP
9514      NEXT,      J/COMP371A
(6071) DCS[0.00.0.0.0.1] BM[0000.00.00.00.00.000.000...0.0.1.0.0.0.0.0000...0.0000.0...11.000...000.111.010]
9515
9516      6072: I(FREE)
9517      COMP371A:
9518      P2-T,      D_SR=XOR-BSPHI(C052525), SAVE-DIC1,      ID = (052525)=SR, BITWISE
9519      NEXT,      J/GOBUT371A
(6072) DCS[0.00.0.0.0.0] BM[0110.01.11.00.00.111.111...0.1.0.0.0.0.0.0000...0.0000.0...11.000...000.111.011]
9520
9521      6073: I(FREE)
9522      GOBUT371A:
9523      SETUP,      RETURN/TEST371B,      !RETURN TO START OF NEXT SUBTEST
```

```

9524      PO,      BUMP-VERIFY,      |COUNT
9525      NEXT,      CALL(DINTOIR=5)  |SUBR: D -> IR, BUT(INSTRS)
(6073) DCS(0.00,0.0,0.0,1) BM(0110.00,11.10,01.011.111.00,0.0,0.0,0.0000.0.0000,0.11,100.010,111,011)

9526
9527
9528
9529
9530
9531
9532
9533      |-----
9534
9535      |*** TEST 371B ***
9536      |TEST-371B CHECKS THAT THE GUARD WASN'T ALTERED EITHER
9537      |7113:
9538      |TEST371B:
9539      |      PO,      LOAD=ENUA(2TARGET401),      |GUARD IS STILL "01"0"00"
9540      |      LOAD=ERROR(TEST371B),      |ERROR DIRECTORY KEY
9541      |      DCS=CTR(C3,),      |COMPARE ENUA1TNUA AT TARGET
9542      |      NEXT,      J/GOBUT371B
(6713) DCS(1.00,1.0,0.0,1) BM(1100.00,11.11,00.000.001.00,0.0,0.0,0.0000.0.0000,0.11,100.000,111,100)

9543
9544      |6074: |(FREE)
9545      |GOBUT371B:
9546      |      SETUP,      RETURN/SCOPE371,      |RETURN TO SCOPE LOOP TEST WORD
9547      |      NEXT,      GOTO-PAGE(7),      |BUT'S ARE ON PAGE 7
9548      |      J/BUTGD3-2      |GO BUT ON GUARD <3:2>
(6074) DCS(0.00,0.0,0.0,1) BM(0110.00,00.01,11.101.111.00,0.0,0.0,0.0000.0.0000,0.11,100.011,001,100)

9549
9550
9551      |SCOPE LOOP TEST FOR SR, GUARD, XMUX, RES AREA CODE
9552      |6075: |(FREE)
9553      |SCOPE371:
9554      |      PO,      RUMP-VERIFY,      |COUNT
9555      |      P2,      RES=CSPD(C000000),      |RESET RES TO SR-LOAD/GUARD-DIS
9556      |      NEXT,      PUTD(SCOPE),      |NO ERROR: "TEST372A" [+3, WORDS]
9557      |      J/TEST372A      |      ERROR: "SETRES361A" [-95, WORDS]
(6075) DCS(0.00,0.1,0.1,1) BM(0000.10,00.00,00.000.000.00,0.0,0.0,0.0100.0.1000,1.11,100.101,111,001)

9558
9559
9560
9561
9562      |-----
9563
9564      |      THE FOLLOWING TWO SUBROUTINES ARE ALSO USED IN THE ABOVE TESTS:
9565
9566
9567      |7031: |(FREE)
9568      |CSP16XORRPTOIR5:
9569      |      P2=T,      D=SR-XOR-CSPB(B16), SAVE-DIC),      |COMPARE SR-XMUX; EXPECTED VALUE, BITWISE
9570      |      NEXT,      J/DINTOIRS      |      AND PUT IN IR TO DO INSTR5 TEST
(7031) DCS(0.00,0.0,0.0,1) BM(0110.11,01.00,00.000.111.00,0.1,0.0,0.0000.0.0000,0.11,100.010,111,011)

```

```

9571
9572
9573      |7035: |(FREE)
9574      |CSP16XORFLTTTOIRS:
9575      |      P2=T,      D=FLTPY-XOR-CSPB(B16), SAVE-DIC),      |COMPARE FLTPY-XMUX; EXPECTED VALUE, BITWISE
9576      |      NEXT,      J/DINTOIRS      |      AND PUT IN IR TO DO INSTR5 TEST
(7035) DCS(0.00,0.0,0.0,1) BM(0110.11,01.00,01.000.111.00,0.1,0.0,0.0000.0.0000,0.11,100.010,111,011)

9577
9578
9579
9580
9581
9582      |PAGE=====
9583
9584
9585      |TNC * TEST372A-372B: TESTING CUA, PROCESSOR MUX, AND BUTA(SUBR A)
9586
9587      |=====
9588      |*
9589      |* TFSTS: 372A - 372B      UNWORDS: 015 + 002
9590      |*
9591      |* FUNCTIONS:      TESTS THAT CUA IS LOADED, AND CAN BE READ THRU PROCESSOR MUX.
9592      |*      ALSO TESTS THAT CAN BE PUT INTO D, AND BUT(SUBR A) LOADS
9593      |*      D<14103> INTO RETURN.
9594      |*
9595      |=====
9596
9597
9598
9599
9600
9601      |-----
9602
9603      |*** TEST 372A ***
9604      |TEST-372A CHECKS THE CUA -> D -> SUBR A -> RETURN PATH WITH PATTERN [6222]
9605      |6571:
9606      |TFST372A:
9607      |      PO,      LOAD=ENUA(CUA372A),      |WHERE WE BUT (RETURN) TO
9608      |      LOAD=ERROR(TEST372A),      |ERROR DIRECTORY KEY
9609      |      DCS=CTR(C7,),      |IN 7, MICROWORDS
9610      |      NEXT,      J/LOOP372A
(6571) DCS(1.00,1.0,0.0,1) BM(1000.00,11.00,10.010.010.00,0.0,0.0,0.0000.0.0000,0.11,100.101,100,000)

9611
9612      |4540:
9613      |LOOP372A:
9614      |      SELECT, UCON=PROC,      |SELECT PROCESSOR UCON CONTROL
9615      |      FNABLF, BUSDIN_CUA[14-03],      |PUT 08CUA[11-00]#EXFLAG<2:1>#FOVP ON BUSDIN
9616      |      PO,      SET=UCON=CONTROL,      |LOAD UCON REGISTER AT P0
9617      |      BUMP-VERIFY,      |COUNT
9618      |      P3,      BUTA(CUA=TRACK),      |RESET TRACKING OF CUA
9619      |      NEXT,      J/STDC372A
(6540) DCS(0.00,0.0,0.0,1) BM(0000.00,00.01,01.000.000.00,0.0,0.0,0.0000.0.0000,0.11,001.000,111,110)

9620
9621      |6076: |(FREE)

```

```

9672 SETDC372A:
9673 P2-T, D_ASPHI(C000000), D(C)=1, ISET D(C) FLAG = (1) FOR FIRST LOOP
9674 NEXT, J/CUA372A
(6076) DCS[0.00,0.0,0.0] BM[1111..00.00..11.01..100..000...0.1.0..0..0.0000...0..0000.0...11.000...010.010.010]
9675
9676 6222:
9677 CUA372A:
9678 P3, CSDP[16]_BUSDIN, RETURN/BUTERRORS, I COPY CUA [WHICH IS ADDRESS OF THIS WORD] INTO CSP
9679 NEXT, BUTR(D(C)=B), I IF D(C) SET [PASS 1], GO TO "LOAD372A"
9680 J/TEST372B I IF D(C) RESET [PASS 2], GO TO "TEST372B"
(6222) DCS[0.00,0.0,0.0] BM[0110..10.01..11.11..110..000...0.0.0..0..0.0001...1..0000.0...10.011...111.000.001]
9681
9682 6703:
9683 LOAD372A:
9684 P0, BUMP-VERIFY, I COUNT
9685 P2-T, D_CSPB(816), D(C)=0, I PUT CUA FROM CSP INTO D, RESET D(C)
9686 NEXT, J/SUBRA372A
(6703) DCS[0.00,0.0,0.1] BM[1010..11.01..00.00..000..000...0.1.0..0..0.0.0000...0..0000.0...11.000...000.111.111]
9687
9688 6077: 1(FREE)
9689 SUBRA372A:
9690 SETUP, RETURN/BUTERRORS, I IF BUTA(SUBR-B) USED INSTEAD
9691 NEXT, PAGE(7), I SUBROUTINE IS ON PAGE 7
9692 BUTA(SUBR-A), I LOAD PAGE, LOAD RETURN FROM D
9693 J/TARGET555 I THE SUBROUTINE
(6077) DCS[0.00,0.0,0.0] BM[0110..00.01..11.11..110..111...0.0.0..0..0.0.0000...0..0000.0...11.101...101.101.101]
9694
9695 I NEXT WORD COMES FROM "ZTARGET555", WHICH DOES ONLY A BUTA(RETURN),
9696 I WHICH SHOULD RETURN TO "CUA372A" [-3, WORDS]
9697
9698
9699
9700
9701
9702
9703
9704
9705
9706
9707
9708
9709
9710
9711
9712
9713
9714

```

```

(6100) DCS[0.00,0.0,0.0] BM[0101..00.01..11.11..110..101...0.0.0..0..0.0000...0..0000.0...11.100...000.000.000]
9669
9670 5000: 1(FREE)
9671 SFTDC372B:
9672 P0, BUMP-VERIFY, I COUNT
9673 P2-T, D_ASPHI(C000000), D(C)=1, I SET D(C) FLAG = (1) FOR FIRST LOOP
9674 P3, BUTA(CUA-TRACK), I RESET TRACKING OF CUA
9675 NEXT, J/CUA372B
(5000) DCS[0.00,0.0,0.1] BM[1111..00.00..11.01..100..000...0.1.0..0..0.0.0000...0..0000.0...11.001...101.101.101]
9676
9677 5555:
9678 CUA372B:
9679 P3, CSDP[16]_BUSDIN, RETURN/BUTERRORS, I COPY CUA [WHICH IS ADDRESS OF THIS WORD] INTO CSP
9680 NEXT, BUTR(D(C)=B), I IF D(C) SET [PASS 1], GO TO "LOAD372B"
9681 J/SCOPE372B I IF D(C) RESET [PASS 2], GO TO "SCOPE372B"
(5555) DCS[0.00,0.0,0.0] BM[0101..10.01..11.11..110..000...0.0.0..0..0.0.0001...1..0000.0...10.011...101.000.001]
9682
9683 5503:
9684 LOAD372B:
9685 P2-T, D_CSPB(816), D(C)=0, I PUT CUA FROM CSP INTO D, RESET D(C)
9686 NEXT, J/SUBRA372B
(5503) DCS[0.00,0.0,0.0] BM[1010..11.01..00.00..000..000...0.1.0..0..0.0.0000...0..0000.0...11.000...010.010.011]
9687
9688 5223: 1(FREE)
9689 SUBRA372B:
9690 P0, BUMP-VERIFY, I COUNT
9691 SETUP, RETURN/BUTERRORS, I IF BUTA(SUBR-B) USED INSTEAD
9692 NEXT, PAGE(7), I SUBROUTINE IS ON PAGE 7
9693 BUTA(SUBR-A), I LOAD PAGE, LOAD RETURN FROM D
9694 J/RESETUCOMP I THE SUBR: RESETS PROC UCOW: BUSDIN_EMIT, EN-CLK-TR
(5223) DCS[0.00,0.0,0.1] BM[0101..00.01..11.11..110..111...0.0.0..0..0.0.0000...0..0000.0...11.101...010.111.001]
9695
9696 I NEXT WORD COMES FROM "RESETUCOMP", WHICH DOES ONLY A BUTA(RETURN),
9697 I WHICH SHOULD RETURN TO "CUA372B" [-3, WORDS]
9698
9699
9700 5501:
9701 SCOPE372B:
9702 NEXT, GOTO-PAGE(6), I XFER
9703 J/TEST373A BUTR[SCOPE], I NO ERROR: "TEST373A" [+1, WORDS]
9704 I ERROR: "LOOP372A" [-12, WORDS]
(5501) DCS[0.00,0.1,0.0] BM[0000..00.00..00.00..000..110...0.0.0..0..0.0.0000...0..0000.0...11.100...101.100.001]
9705
9706
9707
9708
9709
9710
9711
9712
9713
9714

```

```

9715 1*****
9716 1*
9717 1* TESTS: 373 A - B UNORDS: 007 + 014
9718 1*
9719 1* FUNCTIONS:
9720 1*
9721 1* THE FOLLOWING SET OF TWO TESTS PERFORMS SEVERAL FUNCTIONS:
9722 1*
9723 1* TEST-373-A CHECKS THAT CONTROL CAN BE PASSED TO THE BASE MACHINE, VIA
9724 1* BUTA(DIAGNOSE), AND SEVERAL B.M. WORDS EXECUTED. A BUTA(DIAGNOSE) IN
9725 1* THE B.M. SHOULD THEN BE ENCOUNTERED, RETURNING CONTROL TO THE DCS VIA
9726 1* A JAMUPP FORCE.
9727 1*
9728 1* TEST-373-B THEN CHECKS THAT THE B.M. MACHINE CODE CORRECTLY
9729 1* ASSERTED THE "FLPADR-L" EXTENSION SET, FORCING A READ, VIA THE ASP/DF
9730 1* FIELD MODE, OF A SCRATCHPAD.
9731 1*
9732 1*****
9733
9734
9735
9736 1THIS TEST GOES TO THE B.M. VIA BUTA(DIAGNOSE)
9737 6541:
9738 TEST373A:
9739 P0, LOAD-ENUA(4777), JAMUPPS GO HERE IN DCS
9740 LOAD-ERROR(TEST373A), ERROR DIRECTORY KEY
9741 DCS-CTR(C7), COMPARE AT JAMUPP
9742 NEXT, J/RETURN373A
(6541) DCS(1.00,1.0,0,0) BM(1000.00,10.01,11.111.111...0.0,0.0...0.0000...0.0000,0...11.000...101.110,010)
9743
9744 6562:
9745 RETURN373A:
9746 P3, CSPD(00) EMIT, RETURN/TEST373B, RETURN AFTER PROCESSING, TO NEXT TEST
9747 NEXT, GOTO-PAGE(7), JAMUPP ROUTINE EXPECTS RETURN ADDRESS IN CSP(00)
9748 J/SETIR373A
(6562) DCS(0.00,0.0,0,0) BM(0110.10,10.11.11.010.111...0.0,0.0...0.1111...1.0000,0...11.100...000,011,100)
9749
9750 7034: 1(FREE)
9751 SETIR373A:
9752 P2=0, IR_EMIT, EMIT/000002, 1SET DF=2 IN IR
9753 P3, BUTA(DIAGNOSE), 1BEGIN THE TRANSFER SEQUENCE
9754 NEXT, J/SETSR373A
(7034) DCS(0.00,0.0,0,0) BM(0000.00,00.00,00.000.010...0.0,0.0...1.1010...0.0000,0...11.011...000,011,111)
9755
9756 7037: 1(FREE)
9757 SETSR373A:
9758 P2=T, SR_CSPD(C052525), 1SET SR<00>=1 FOR JAMUPP EXPECTED (BY JAMUPP SERVICER)
9759 NEXT, GOTO-PAGE(3), 1SET PAGE TO BE ACTUAL (3)
9760 J/MED25 1POINT UP TO B.M. DISP, ON PAGE
(7037) DCS(0.00,0.0,0,0) BM(1010.10,00.00,00.000.011...0.0,1.0...0.0111...0.0000,0...11.100...000,010,000)
9761
9762 1THE SEQUENCE OF CONTROL SHOULD NOW BE COMING FROM THE B.M.

```

```

9763
9764 13020:
9765 1MED25:
9766 1 NEXT, GOTO-PAGE(2), XFER
9767 1 J/MED25A
9768 1
9769 12071:
9770 1MED25A:
9771 1 P2=T, D_ASPHI(DF)-TOP, SAVE-D(C), 1READ FROM ASPHI/DF, WITH FLPADR ASSERTED
9772 1 1SINCE DF=2, THE SPADDRS=12 WILL BE FORCED
9773 1 P3, BUTA(DIAGNOSE), 1ALSO START TO RETURN XFER TO DCS, VIA JAMUPP
9774 1 NEXT, J/MED19A
9775 1
9776 12066:
9777 1MED19A:
9778 1 NEXT, GOTO-PAGE(3), 1THIS WORD ALSO GETS EXECUTED
9779 1 J/MED19 1BUT NOW JAMUPP GETS FORCED, AND WE GO NO FURTHER
9780 1
9781 1*** CONTROL NOW COMES BACK TO DCS JAMUPP POINT ***
9782 1
9783 14777:
9784 1JAMUPP001:
9785 1 P3=T, SP_D, 1SAVE OLD D IN SR
9786 1 NEXT, RUTR(SR00), 1TEST SR<00>:
9787 1 1IF=0, NOT-EXPECTED JAM, GOTO(JAMUPP003), (SEE JAMUPP ROUTINE)
9788 1 J/JAMUPP003 1IF=1, EXPECTED JAM, GOTO(JAMUPP002B)
9789 1
9790 1* COME HERE FOR EXPECTED JAM *
9791 14757:
9792 1JAMUPP002R:
9793 1 P2=T, D_CSPD(000), SAVE-D(C), 1GET RETURN ADDRESS, STORED IN CSP(00)
9794 1 NEXT, J/JAMUPP002C
9795 1
9796 1=4000(4777)
9797 1JAMUPP002C:
9798 1 P0, RETURN_D(14-03), PAGE(7), 1PUT RETURN ADDRESS INTO RETURN REGISTER
9799 1 P2=T, D_SR, SAVE-D(C), 1RESTORE OLD D
9800 1 NEXT, J/JAMUPP002D
9801 1
9802 1=7000(7377)
9803 1JAMUPP002D:
9804 1 P2=T, SR_ZERO, 1ZERO OUT SR<00>, FURTHER JAMUPPS NOW ILLEGAL
9805 1 NEXT, BUTA(RETURN), 1AND RETURN TO FLOWS
9806 1 J/RUTERR07 1ONLY IF ERROR
9807 1
9808
9809 1AT THIS POINT, CONTROL SHOULD NOW RETURN TO THIS POINT, THE NEXT TEST
9810 1THIS TEST NOW CHECKS TO SEE THAT THE B.M. FUNCTION WAS EXECUTED CORRECTLY
9811 1NOTE THAT IN TEST-350 ( A WHILE BACK ), A=BSPHI(12) WERE LOADED WITH (000152) DATA
9812
9813 4572:
9814 TEST373B:
9815 P0, LOAD-ENUA(2TARGET425), 1FOR INSTR5-E78-(425) DECODEF
9816 LOAD-ERROR(TEST373B), 1ERROR DIRECTORY KEY

```

```

9817          DCS=CTR(C6.),          ICOMPARE AT TARGET
9818          NEXT,          J/GOBUT373B
(6577) DCS[1,00,1,0,0,0] BM[1001,00,11,11,00,010,101,0,0,0,0,0,0,0,0000,0,0,0,0000,0,0,11,000,0001,000,001]
9819
9820          6101: 1(FREE)
9821          GOBUT373B:
9822          SETUP,          RETURN/SCOPE373,          IRETURN TO SCOPE LOOP TEST WORD
9823          NEXT,          CALL(DINTOIP-S)          IGO DO D -> DBUF -> IR, BUT(INSTR5)
(6101) DCS[0,00,0,0,0,0] RM[0110,00,00,10,00,010,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,010,111,011]
9824
9825
9826
9827          6102: 1(FREE)
9828          SCOPE373:
9829          PO,          PUMP-VERIFY,          ICOUNT
9830          NEXT,          BUTD[SCOPE],          INO ERROR; "TEST374" (+1, WORDS)
9831          J/TEST374          I ERROR; "RETURN373A" (-12, WORDS)
(6102) DCS[0,00,0,1,0,1] BM[0000,00,00,00,00,0000,0000,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0,101,110,011]
9832
9833
9834
9835
9836
9837
9838
9839
9840
9841
9842
9843
9844
9845
9846
9847
9848
9849
9850
9851
9852
9853
9854
9855
9856
9857
9858
9859
9860
9861
9862
9863
9864
9865
9866

```

[illegible]

```
9917
9918 5536:
9919 ZERO374A1:
9920 P0, BUMP-VERIFY, ICOUNT
9921 SETUP, RETURN/DOWRITE374A1, IEXEC SUBR WHICH:
9922 NFXT, CALL[ZEROSF04DF02] I(1) (000402) -> IR,
9923 (5536) DCS[0.00.0.0.0.0.1] BM[0101.00.01.00.10.101.111.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.100.000.100.001] I(2) WRITES ZEROES TO A/B SP HI/LO SF/DF

9924
9925 5225: 1(FREE)
9926 DOWRITE374A1:
9927 P2-T, D_CSPD(D01), D[C]_0, IDATA WITH BIT<07> SET
9928 ASP-ADDRS-R[SF], IADDRESS ASP WITH SF MODE
9929 BSP-ADDRS-R[DF], IADDRESS BSP WITH DF MODE
9930 P3, WR(A,LO,A-ADDR), ISELECT THE PARTICULAR FUNCTION TO TEST,
9931 NFXT, J/GETTEM374A1 I USING A-ADDR FOR REWRITE
9932 (5225) DCS[0.00.0.0.0.0] BM[1010.10.00.00.11.000.000.0.0.1.0.0.0.0.110.0.0.0001.0.0.11.000.0.010.010.110]

9933
9934 5226: 1(FREE)
9935 GETTEM374A1:
9936 P3, CSPD[00]_EMIT, RETURN/ZERO374A2, I(SEE DESCRIPT OF SUBR FOR FUNCTION)
9937 NFXT, CALL[SPDFT08R]
9938 (5226) DCS[0.00.0.0.0.0] BM[0101.10.01.00.10.111.101.0.0.0.0.0.0.0.1111.0.0.0000.0.0.11.100.0.111.100.110]

9939
9940 5227: 1(FREE)
9941 ZERO374A2:
9942 SETUP, RETURN/DOWRITE374A2, IAGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
9943 NFXT, CALL[ZEROSF04DF02]
9944 (5227) DCS[0.00.0.0.0.0] BM[0101.00.01.00.11.000.0.111.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.100.000.100.001]

9945
9946 5230: 1(FREE)
9947 DOWRITE374A2:
9948 P2-T, D_CSPD(D01), D[C]_0, IDATA WITH BIT<07> SET
9949 ASP-ADDRS-R[SF], IADDRESS ASP WITH SF MODE
9950 BSP-ADDRS-R[DF], IADDRESS BSP WITH DF MODE
9951 P3, WR(A,LO,B-ADDR), IUSE SAME FUNCTION AS ABOVE,
9952 NFXT, J/GETTEM374A2 I ONLY USE B-ADDR FOR REWRITE THIS TIME
9953 (5230) DCS[0.00.0.0.0.0] BM[1010.10.00.00.11.000.000.0.0.1.0.0.0.0.1110.0.0.0.0101.0.0.11.000.0.010.011.001]

9954
9955 5231: 1(FREE)
9956 GETTEM374A2:
9957 P3, CSPD[00]_EMIT, RETURN/TEST374A2, I(SEE DESCRIPT OF SUBR FOR FUNCTION)
9958 NFXT, CALL[SPDFT08R]
9959 (5231) DCS[0.00.0.0.0.0] BM[0101.10.11.00.00.100.101.0.0.0.0.0.0.0.1111.0.0.0000.0.0.11.100.0.111.100.110]

9960
9961 5604:
9962 TEST374A2:
9963 P0, LOAD-ENWA(2TARGET434), INOW SETUP FOR IR=ZERO COMPARE
9964 LOAD=ERROR(TEST374A2), IERROR DIRECTORY KEY
9965 DCS=CTR(C6.), ICOMPARE AT TARGET
```

```
9963
9964 NFXT, J/GETEST374A2 ICOUNT
9965 (5604) DCS[1.00.1.0.0.0.1] BM[1001.00.11.11.00.011.100.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.000.0.010.011.010]

9966
9967 5232: 1(FREE)
9968 GETEST374A2:
9969 SETUP, RETURN/SCOPE374A, IGO EXEC SUBR THAT:
9970 NFXT, CALL[DINTOIR=3] I PUTS D -> IR, BUT(INSTRS) TO TEST FOR ZERO
9971 (5232) DCS[0.00.0.0.0.0] BM[0101.00.01.00.11.011.111.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.100.0.010.111.011]

9972
9973
9974 5233: 1(FREE)
9975 SCOPE374A:
9976 NFXT, RUTD[SCOPE], INO ERROR: "EXPEC374B1" (+1. WORDS)
9977 (5233) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.000.0.101.011.111] I ERROR: "ZERO374A1" (-8. WORDS)

9978
9979
9980
9981
9982
9983
9984
9985
9986
9987 5537:
9988 EXPEC374B1:
9989 P3, CSPD[02]_EMIT, EMIT/100010, IEXPECTED "SERIAL" REPRESENTATION OF RESULT
9990 NFXT, J/ZERO374B1
9991 (5537) DCS[0.00.0.0.0.0] BM[1000.10.00.00.00.001.000.0.0.0.0.0.0.0.1101.0.0.0000.0.0.11.000.0.101.101.110]

9992
9993 5556:
9994 ZERO374B1:
9995 P0, BUMP-VERIFY, ICOUNT
9996 SETUP, RETURN/DOWRITE374B1, IEXEC SUBR WHICH:
9997 NFXT, CALL[ZEROSF04DF02] I(1) (000402) -> IR,
9998 (5556) DCS[0.00.0.0.0.0.1] BM[0101.00.01.00.11.100.111.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.100.000.100.001] I(2) WRITES ZEROES TO A/B SP HI/LO SF/DF

9999
10000 5234: 1(FREE)
10001 DOWRITE374B1:
10002 P2-T, D_CSPD(D01), D[C]_0, IDATA WITH BIT<07> SET
10003 ASP-ADDRS-R[DF], IADDRESS ASP WITH DF MODE
10004 BSP-ADDRS-R[SF], IADDRESS BSP WITH SF MODE
10005 P3, WR(A,HI,A-ADDR), ISELECT THE PARTICULAR FUNCTION TO TEST,
10006 NFXT, J/GETTEM374B1 I USING A-ADDR FOR REWRITE
10007 (5234) DCS[0.00.0.0.0.0] BM[1010.10.01.00.10.000.000.0.0.1.0.0.0.0.1110.0.0.1001.0.0.11.000.0.010.011.101]

10008
10009 5235: 1(FREE)
10010 GETTEM374B1:
10011 P3, CSPD[00]_EMIT, RETURN/ZERO374B2, I(SEE DESCRIPT OF SUBR FOR FUNCTION)
```

```
10009 NEXT, CALL(SFDFTOBR)
(5235) DCS(0.00,0.0,0.0) BM(0101..10.01..00.11..110..101...0.0.0..0...0.1111...1..0000,0...11,100...111,100.110)
10010
10011
10012
10013 5236: 1(FREE)
10014 ZERO374B2:
10015 SETUP, RETURN/DOWRITE374B2, |AGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
10016 NEXT, CALL(ZEROSF04DF02)
(5236) DCS(0.00,0.0,0.0) BM(0101..00.01..00.11..111..111...0.0.0..0...0.0000...0..0000,0...11,100...000,100.001)
10017
10018 5237: 1(FREE)
10019 DOWRITE374B2:
10020 P2-T, D_CSPD(D01), D(C)0, |DATA WITH BIT<07> SET
10021 ASP-ADDRS-R(DF), |ADDRESS ASP WITH DF MODE
10022 BSP-ADDRS-R(SF), |ADDRESS BSP WITH SF MODE
10023 P3, WP(A,HI,B-ADDR), |USE SAME FUNCTION AS ABOVE,
10024 NEXT, J/GETTEM374B2 | ONLY USE B-ADDR FOR REWRITE THIS TIME
(5237) DCS(0.00,0.0,0.0) BM(1010..10.01..00.10..000..000...0.1.0..0...0.1110...0..1101,0...11,000...010,100.000)
10025
10026 5240: 1(FREE)
10027 GETTEM374B2:
10028 P3, CSPD(00)_EMIT, RETURN/TEST374B2, |(SEE DESCRIPT OF SUBR FOR FUNCTION)
10029 NEXT, CALL(SFDFTOBR)
(5240) DCS(0.00,0.0,0.0) BM(0101..10.11..00.00..110..101...0.0.0..0...0.1111...1..0000,0...11,100...111,100.110)
10030
10031 5606:
10032 TEST374B2:
10033 P0, LOAD-ENVAL(ZTARGET434), |NOW SETUP FOR IR=ZERO COMPARE
10034 LOAD-ERROR(TEST374B2), |ERROR DIRECTORY KEY
10035 DCS-CTR(C6), |COMPARE AT TARGET
10036 RUMP-VERIFY, |COUNT
10037 NEXT, J/GOTEST374B2
(5606) DCS(1.00,1.0,0.1) BM(1001..00.11..11.00..011..100...0.0.0..0...0.0000...0..0000,0...11,000...010,100.001)
10038
10039 5241: 1(FREE)
10040 GOTEST374B2:
10041 SETUP, RETURN/SCOPE374B, |GO EXEC SUBR THAT:
10042 NEXT, CALL(DINTORR=5) | PUTS D -> IR, BUT(INSTRS) TO TEST FOR ZERO
(5241) DCS(0.00,0.0,0.0) BM(0101..00.01..01.00..010..111...0.0.0..0...0.0000...0..0000,0...11,100...010,111.011)
10043
10044
10045
10046
10047 5242: 1(FREE)
10048 SCOPE374B:
10049 NEXT, AUTO(SCOPE), |NO ERROR: "EXPEC374C1" (+1. WORDS)
10050 J/EXPEC374C1 | ERROR: "ZERO374B1" (-8. WORDS)
(5242) DCS(0.00,0.1,0.0) BM(0000..00.00..00.00..000..000...0.0.0..0...0.0000...0..0000,0...11,000...101,101.111)
10051
10052
10053
```

```
10054
10055
10056
10057 *** TEST 374C ***
10058
10059 5557:
10060 EXPEC374C1:
10061 P3, CSPD(02)_EMIT, EMIT/010001, |EXPECTED "SERIAL" REPRESENTATION OF RESULT
10062 NFXT, J/ZERO374C1
(5557) DCS(0.00,0.0,0.0) BM(0001..10.00..00.00..000..001...0.0.0..0...0.1101...1..0000,0...11,000...101,110.110)
10063
10064 5566:
10065 ZPRO374C1:
10066 P0, RUMP-VERIFY, |COUNT
10067 SETUP, RETURN/DOWRITE374C1, |EXEC SUBR WHICH:
10068 NEXT, CALL(ZEROSF02DF04) | (1) (000204) -> IR,
10069 | (2) WRITES ZEROES TO A/B SP HI/LO SF/DF
(5566) DCS(0.00,0.0,0.1) BM(0101..00.01..01.00..011..111...0.0.0..0...0.0000...0..0000,0...11,100...000,011.110)
10070
10071 5243: 1(FREE)
10072 DOWRITE374C1:
10073 P2-T, D_CSPD(D01), D(C)0, |DATA WITH BIT<07> SET
10074 ASP-ADDRS-R(DF), |ADDRESS ASP WITH DF MODE
10075 BSP-ADDRS-R(SF), |ADDRESS BSP WITH SF MODE
10076 P3, WP(B,LO,A-ADDR), |SELECT THE PARTICULAR FUNCTION TO TEST,
10077 NEXT, J/GETTEM374C1 | USING A-ADDR FOR REWRITE
(5243) DCS(0.00,0.0,0.0) BM(1010..10.01..00.10..000..000...0.1.0..0...0.1110...0..0010,0...11,000...010,100.100)
10078
10079 5244: 1(FREE)
10080 GETTEM374C1:
10081 P3, CSPD(00)_EMIT, RETURN/ZERO374C2, |(SEE DESCRIPT OF SUBR FOR FUNCTION)
10082 NEXT, CALL(SFDFTOBR)
(5244) DCS(0.00,0.0,0.0) BM(0101..10.01..01.00..110..101...0.0.0..0...0.1111...1..0000,0...11,100...111,100.110)
10083
10084
10085
10086 5246: 1(FREE)
10087 ZERO374C2:
10088 SETUP, RETURN/DOWRITE374C2, |AGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
10089 NEXT, CALL(ZEROSF02DF04)
(5246) DCS(0.00,0.0,0.0) BM(0101..00.01..01.00..111..111...0.0.0..0...0.0000...0..0000,0...11,100...000,011.110)
10090
10091 5247: 1(FREE)
10092 DOWRITE374C2:
10093 P2-T, D_CSPD(D01), D(C)0, |DATA WITH BIT<07> SET
10094 ASP-ADDRS-R(DF), |ADDRESS ASP WITH DF MODE
10095 BSP-ADDRS-R(SF), |ADDRESS BSP WITH SF MODE
10096 P3, WP(R,LO,B-ADDR), |USE SAME FUNCTION AS ABOVE,
10097 NEXT, J/GETTEM374C2 | ONLY USE B-ADDR FOR REWRITE THIS TIME
(5247) DCS(0.00,0.0,0.0) BM(1010..10.01..00.10..000..000...0.1.0..0...0.1110...0..0110,0...11,000...010,101.000)
10098
10099 5250: 1(FREE)
```

```

10100 GETTEN374C2:
10101 P3, CSPD[00]_EMIT, RETURN/TEST374C2, 1(SEE DESCRIPT OF SUBR FOR FUNCTION)
10102 NEXT, CALL[SFDFTOGR]
(5250) DCS[0.00.0.0.0.0] BM[0101..10.10..10.10..100..101..0.0.0.0.0.0.0.1111..1.0000.0.0.11.100..111.100.110]
10103
10104 5251: 1(FREE)
10105 TEST374C2:
10106 P0, LOAD=ENUA(2TARGET434), INOV SETUP FOR IR=ZERO COMPARE
10107 LOAD=ERROR(TEST374C2), ERROR DIRECTORY KEY
10108 DCS=CTR(C6.), ICOMPARE AT TARGET
10109 BUMP=VERIFY, ICOUNT
10110 NEXT, J/GOTEST374C2
(5252) DCS[1.00.1.0.0.1] BM[1001..00.11..11.00..011..100..0.0.0.0.0.0.0.0000..0.0000.0.0.11.000..010.101.001]
10111
10112 5251: 1(FREE)
10113 GOTEST374C2:
10114 SETUP, RETURN/SCOPE374C, IGO EXEC SUBR THAT:
10115 NEXT, CALL[DINTOIR=5] I PUTS D -> IR, BUT(INSTRS) TO TEST FOR ZERO
(5251) DCS[0.00.0.0.0.0] BM[0101..00.01..01.01..010..111..0.0.0.0.0.0.0.0000..0.0000.0.0.11.100..010.111.011]
10116
10117
10118
10119
10120 5252: 1(FREE)
10121 SCOPE374C:
10122 NEXT, BUTD[SCOPE], INO ERROR: "EXPEC374D1" (+1, WORDS)
10123 J/EXPEC374D1 I ERROR: "ZERO374C1" (-9, WORDS)
(5252) DCS[0.00.0.1.0.0] BM[0000..00.00..00.00..000..000..0.0.0.0.0.0.0.0000..0.0000.0.0.11.000..101.110.111]
10124
10125
10126
10127
10128 I - - - - -
10129
10130 I*** TEST 374D ***
10131
10132 5567:
10133 EXPEC374D1:
10134 P3, CSPD[02]_EMIT, EMIT/001040, IEXPECTED "SERIAL" REPRESENTATION OF RESULT
10135 NEXT, J/ZERO374D1
(5567) DCS[0.00.0.0.0.0] BM[0000..10.00..10.00..100..000..0.0.0.0.0.0.0.1101..1.0000.0.0.11.000..101.001.100]
10136
10137 5514:
10138 ZERO374D1:
10139 P0, BUMP=VERIFY, ICOUNT
10140 SETUP, RETURN/DOWRITE374D1, IEXEC SUBR WHICH:
10141 I(1) (000204) -> IR,
10142 NEXT, CALL[ZEROSFO2DF04] I(2) WRITES ZEROS TO A/B SP HI/LO SP/DF
(5514) DCS[0.00.0.0.0.1] BM[0101..00.01..01.01..011..111..0.0.0.0.0.0.0.0000..0.0000.0.0.11.100..000.011.110]
10143
10144 5253: 1(FREE)
10145 DOWRITE374D1:

```

```

10146 P2=T, D_CSPD(D01), D[C]=0, I(ATA WITH BIT<07> SET
10147 ASP=ADDRS-R(SF), IADDRESS ASP WITH SF MODE
10148 BSP=ADDRS-R(DF), IADDRESS BSP WITH DF MODE
10149 P3, WR(B,HI,A=ADDR), ISELECT THE PARTICULAR FUNCTION TO TEST,
10150 NEXT, J/GETTEN374D1 I USING A=ADDR FOR REWRITE
(5253) DCS[0.00.0.0.0.0] BM[1010..10.00..00.11..000..000..0.1.0.0.0.0.0.1110..0.1110.0.0.11.000..010.101.100]
10151
10152 5254: 1(FREE)
10153 GETTEN374D1:
10154 P3, CSPD[00]_EMIT, RETURN/ZERO374D2, I(SEE DESCRIPT OF SUBR FOR FUNCTION)
10155 NEXT, CALL[SFDFTOGR]
(5254) DCS[0.00.0.0.0.0] BM[0101..10.01..01.01..101..101..0.0.0.0.0.0.0.1111..1.0000.0.0.11.100..111.100.110]
10156
10157
10158
10159 5255: 1(FREE)
10160 ZEP0374D2:
10161 SETUP, RETURN/DOWRITE374D2, IAGAIN GO WRITE ZEROS TO A/B-SP-HI/LO
10162 NEXT, CALL[ZEROSFO2DF04]
(5255) DCS[0.00.0.0.0.0] BM[0101..00.01..01.01..110..111..0.0.0.0.0.0.0.0000..0.0000.0.0.11.100..000.011.110]
10163
10164 5256: 1(FREE)
10165 DOWRITE374D2:
10166 P2=T, D_CSPD(D01), D[C]=0, I(ATA WITH BIT<07> SET
10167 ASP=ADDRS-R(SF), IADDRESS ASP WITH SF MODE
10168 BSP=ADDRS-R(DF), IADDRESS BSP WITH DF MODE
10169 P3, WR(B,HI,B=ADDR), IUSE SAME FUNCTION AS ABOVE,
10170 NEXT, J/GETTEN374D2 I ONLY USE B=ADDR FOR REWRITE THIS TIME
(5256) DCS[0.00.0.0.0.0] BM[1010..10.00..00.11..000..000..0.1.0.0.0.0.0.1110..0.1110.0.0.11.000..010.101.111]
10171
10172 5257: 1(FREE)
10173 GETTEN374D2:
10174 P3, CSPD[00]_EMIT, RETURN/TEST374D2, I(SEE DESCRIPT OF SUBR FOR FUNCTION)
10175 NEXT, CALL[SFDFTOGR]
(5257) DCS[0.00.0.0.0.0] BM[0101..10.10..10.00..000..101..0.0.0.0.0.0.0.1111..1.0000.0.0.11.100..111.100.110]
10176
10177 5500:
10178 TEST374D2:
10179 P0, LOAD=ENUA(2TARGET434), INOV SETUP FOR IR=ZERO COMPARE
10180 LOAD=ERROR(TEST374D2), ERROR DIRECTORY KEY
10181 DCS=CTR(C6.), ICOMPARE AT TARGET
10182 BUMP=VERIFY, ICOUNT
10183 NEXT, J/GOTEST374D2
(5500) DCS[1.00.1.0.0.1] BM[1001..00.11..11.00..011..100..0.0.0.0.0.0.0.0000..0.0000.0.0.11.000..010.110.000]
10184
10185 5260: 1(FREE)
10186 GOTEST374D2:
10187 SETUP, RETURN/SCOPE374D, IGO EXEC SUBR THAT:
10188 NEXT, CALL[DINTOIR=5] I PUTS D -> IR, BUT(INSTRS) TO TEST FOR ZERO
(5260) DCS[0.00.0.0.0.0] BM[0101..00.01..01.10..001..111..0.0.0.0.0.0.0.0000..0.0000.0.0.11.100..010.111.011]
10189
10190

```

```
10191
10192
10193 5261: 1(FREE)
10194 SCOPE374D1
10195 NEXT, BUTD[SCOPE], INO ERROR: "EXPEC374E1" (+1, WORDS)
10196 J/EXPEC374E1 ERROR: "ZERO374D1" (-8, WORDS)
(5261) DCS(0.00,0.1,0.0) BM(0000..00,00..00,00,000,000...0,0,0,0,0,0,0000...0,0000,0...11,000...101,001,101)

10197
10198
10199
10200
10201 | - - - - -
10202
10203 *** TEST 374E ***
10204
10205 5515:
10206 EXPEC374E1:
10207 P3, CSPD[02]_EMIT, EMIT/002520, EXPECTED "SERIAL" REPRESENTATION OF RESULT
10208 NEXT, J/ZERO374E1
(5515) DCS(0.00,0.0,0.0) BM(0000..10,01..01,01..010,000...0,0,0,0,0,0,0,1101...1,0000,0...11,000...101,010,110)

10209
10210 5526:
10211 ZERO374E1:
10212 P0, BUMP-VERIFY, COUNT
10213 SFTUP, RETURN/DOWRITE374E1, EXEC SUBR WHICH:
10214 NEXT, CALL[ZEROSF04DF02] (1) (800402) -> IR,
10215 (5526) DCS(0.00,0.0,0.0) BM(0101..00,01..01,10..010,011...0,0,0,0,0,0,0,0000...0,0000,0...11,100...000,100,001)
(2) WRITES ZEROES TO A/B SP HI/LO SF/DF

10216
10217 5262: 1(FREE)
10218 DOWRITE374E1:
10219 P2-T, D_CSPD(D01), D(C)_0, DATA WITH BIT<07> SET
10220 ASP-ADDRS-R(SF), ADDRESS ASP WITH SF MODE
10221 BSP-ADDRS-R(DF), ADDRESS BSP WITH DF MODE
10222 P3, WR(AB,LO,A-ADDR), SELECT THE PARTICULAR FUNCTION TO TEST,
10223 NEXT, J/GETTEM374E1 USING A-ADDR FOR REWRITE
(5262) DCS(0.00,0.0,0.0) BM(1010..10,00..00,11..000,000...0,1,0,0,0,0,0,1110...0,0011,0...11,000...010,110,011)

10224
10225 5263: 1(FREE)
10226 GETTEM374E1:
10227 P3, CSPD[00]_EMIT, RETURN/ZERO374E2, (SEE DESCRIPT OF SUBR FOR FUNCTION)
10228 NEXT, CALL[SPDFT08R]
(5263) DCS(0.00,0.0,0.0) BM(0101..10,01..01,10..100,101...0,0,0,0,0,0,0,1111...1,0000,0...11,100...111,100,110)

10229
10230
10231
10232 5264: 1(FREE)
10233 ZERO374E2:
10234 SFTUP, RETURN/DOWRITE374E2, AGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
10235 NFXT, CALL[ZEROSF04DF02]
(5264) DCS(0.00,0.0,0.0) BM(0101..00,01..01,10..101,111...0,0,0,0,0,0,0,0000...0,0000,0...11,100...000,100,001)

10236
```

```
10237 5265: 1(FREE)
10238 DOWRITE374F2:
10239 P2-T, D_CSPD(D01), D(C)_0, DATA WITH BIT<07> SET
10240 ASP-ADDRS-R(SF), ADDRESS ASP WITH SF MODE
10241 BSP-ADDRS-R(DF), ADDRESS BSP WITH DF MODE
10242 P3, WR(AB,LO,B-ADDR), USE SAME FUNCTION AS ABOVE,
10243 NEXT, J/GETTEM374E2 ONLY USE B-ADDR FOR REWRITE THIS TIME
(5265) DCS(0.00,0.0,0.0) BM(1010..10,00..00,11..000,000...0,1,0,0,0,0,0,1110...0,0111,0...11,000...010,110,110)

10244
10245 5266: 1(FREE)
10246 GETTEM374F2:
10247 P3, CSPD[00]_EMIT, RETURN/TEST374E2, (SEE DESCRIPT OF SUBR FOR FUNCTION)
10248 NEXT, CALL[SPDFT08R]
(5266) DCS(0.00,0.0,0.0) BM(0101..10,10..10,00..010,101...0,0,0,0,0,0,0,1111...1,0000,0...11,100...111,100,110)

10249
10250 5502:
10251 TEST374E2:
10252 P0, LOAD-FNUA(ZTARGET434), INOW SETUP FOR IN=ZERO COMPARE
10253 LOAD-ERROR(TEST374E2), ERROR DIRECTORY KEY
10254 DCS-CTR(C6), COMPARE AT TARGET
10255 BUMP-VERIFY, COUNT
10256 NFXT, J/GOTEST374E2
(5502) DCS(1.00,1.0,0.1) BM(1001..00,11..11,00..011,100...0,0,0,0,0,0,0,0000...0,0000,0...11,000...010,110,111)

10257
10258 5267: 1(FREE)
10259 GOTEST374F2:
10260 SETUP, RETURN/SCOPE374E, GO EXEC SUBR THAT:
10261 NEXT, CALL[DINTOIR=5] PUTS D -> IR, BUT(INSTR5) TO TEST FOR ZERO
(5267) DCS(0.00,0.0,0.0) BM(0101..00,01..01,11..000,011...0,0,0,0,0,0,0,0000...0,0000,0...11,100...010,111,011)

10262
10263
10264
10265
10266 5270: 1(FREE)
10267 SCOPE374E:
10268 NEXT, BUTD[SCOPE], INO ERROR: "EXPEC374F1" (+1, WORDS)
10269 J/EXPEC374F1 ERROR: "ZERO374E1" (-8, WORDS)
(5270) DCS(0.00,0.1,0.0) BM(0000..00,00..00,00,000,000...0,0,0,0,0,0,0,0000...0,0000,0...11,000...101,010,111)

10270
10271
10272
10273
10274 | - - - - -
10275
10276 *** TEST 374F ***
10277
10278 5527:
10279 EXPEC374F1:
10280 P3, CSPD[02]_EMIT, EMIT/120012, EXPECTED "SERIAL" REPRESENTATION OF RESULT
10281 NFXT, J/ZERO374F1
(5527) DCS(0.00,0.0,0.0) BM(1010..10,00..00,00,001,010...0,0,0,0,0,0,0,1101...1,0000,0...11,000...101,001,110)

10282
10283 5516:
```

```

10284 ZEP0374F1:
10285 P0, BUMP-VERIFY, ICOUNT
10286 SETUP, RETURN/DOWRITE374F1, IEXEC SUBR WHICH:
10287 NEXT, CALL(ZEROSF02DF04) I(1) (000204) -> IR,
10288 (5516) DCS[0.00,0.0,0.0,1] BM10101..00.01..01.11..001..111..0.0.0.0..0.0000..0.0000.0...11.100...000.011.110]
I(2) WRITES ZEROES TO A/B SP HI/LO SP/DF

10289 5271: 1(FREE)
10290 DOWRITE374F1:
10291 P2-T, D_CSPD(D01), D[C]-0, I DATA WITH BIT<07> SET
10292 ASP-ADDRS-R[DF], I ADDRESS ASP WITH DF MODE
10293 BSP-ADDRS-R[SF], I ADDRESS BSP WITH SF MODE
10294 P3, WR(AB,HI,A-ADDR), I SELECT THE PARTICULAR FUNCTION TO TEST,
10295 NEXT, J/GETTEM374F1 I USING A-ADDR FOR REWRITE
10296 (5271) DCS[0.00,0.0,0.0,0] BM1010..10.01..00.10..000..000..0.1.0.0.0.0..0.1110..0..1011.0...11.000...010.111.010]

10297 5272: 1(FREE)
10298 GETTEM374F1: I(SEE DESCRIPT OF SUBR FOR FUNCTION)
10299 P3, CSPD[00]_EMIT, RETURN/ZERO374F2,
10300 NEXT, CALL(SPDFTOBR) I
10301 (5272) DCS[0.00,0.0,0.0,0] BM10101..10.01..01.11..011..101..0.0.0.0..0.0.0.0.1111...1.0000.0...11.100...111.100.110]

10302
10303
10304 5273: 1(FREE)
10305 ZERO374F2: I AGAIN GO WRITE ZEROES TO A/B-SP-HI/LO
10306 SETUP, RETURN/DOWRITE374F2,
10307 NEXT, CALL(ZEROSF02DF04) I
10308 (5273) DCS[0.00,0.0,0.0,0] BM10101..00.01..01.11..100..111..0.0.0.0..0.0.0.0.0000..0.0000.0...11.100...000.011.110]

10309 5274: 1(FREE)
10310 DOWRITE374F2: I DATA WITH BIT<07> SET
10311 P2-T, D_CSPD(D01), D[C]-0, I ADDRESS ASP WITH DF MODE
10312 ASP-ADDRS-R[DF], I ADDRESS BSP WITH SF MODE
10313 BSP-ADDRS-R[SF], I ADDRESS BSP WITH SF MODE
10314 P3, WR(AB,HI,B-ADDR), I USE SAME FUNCTION AS ABOVE,
10315 NEXT, J/GETTEM374F2 I ONLY USE B-ADDR FOR REWRITE THIS TIME
10316 (5274) DCS[0.00,0.0,0.0,0] BM1010..10.01..00.10..000..000..0.1.0.0.0.0..0.1110..0..1111.0...11.000...010.111.101]

10317 5275: 1(FREE)
10318 GETTEM374F2: I(SEE DESCRIPT OF SUBR FOR FUNCTION)
10319 P3, CSPD[00]_EMIT, RETURN/TEST374F2,
10320 NEXT, CALL(SPDFTOBR) I
10321 (5275) DCS[0.00,0.0,0.0,0] BM10101..10.10..10.00..100..101..0.0.0.0..0.0.0.0.1111...1.0000.0...11.100...111.100.110]

10322 5504:
10323 TEST374F2:
10324 P0, LOAD-ENUA(ZTARGET434), I NOW SETUP FOR IR=ZERO COMPARE
10325 P0, LOAD-ERROR(TEST374F2), I ERROR DIRECTORY KEY
10326 DCS-CTR(C6), I COMPARE AT TARGET
10327 BUMP-VERIFY, I COUNT
10328 NEXT, J/GOTEST374F2 I
10329

```

```

(5504) DCS[1.00,1.0,0.0,1] BM1001..00.11..11.00..011..100...0.0.0.0..0.0.0.0.0000..0.0000.0...11.000...010.111.110]

10330 5276: 1(FREE)
10331 GOTEST374F2: I GO EXEC SUBR THAT:
10332 SETUP, RETURN/SCOPE374F, I PUTS D -> IR, BUT(INSTR5) TO TEST FOR ZERO
10333 NEXT, CALL(DINTOIR-5) I
10334 (5276) DCS[0.00,0.0,0.0,0] BM10101..00.01..01.11..111..111..0.0.0.0..0.0.0.0.0000..0.0000.0...11.100...010.111.011]

10335
10336
10337 5277: 1(FREE)
10338 SCOPE374F1: I RESET RES TO SR-LOAD/GUARD-DISABLED
10339 P2, RES_CSPD(C000000), I NO ERROR; "TEST375A" (+17, WORDS)
10340 NEXT, PUTD(SCOPE), I ERROR; "ZERO374F1" (-8, WORDS)
10341 J/TEST375A I
10342 (5277) DCS[0.00,0.0,0.0,0] BM10000..10.00..00.00..000..000..0.0.0.0..0.0.0.0.0100..0.1000.1...11.000...101.001.111]

10343
10344
10345
10346
10347
10348
10349
10350 THE FOLLOWING SUBROUTINES ARE USED IN THE ABOVE TESTS (374 A1-F2):
10351
10352 THIS FIRST SUBROUTINE LOADS THE IR WITH THE APPROPRIATELY
10353 SELECTED SF AND DF FIELD VALUES, THEN PROCEEDS TO
10354 ZERO OUT THE ASP AND BSP LOCATIONS CORRESPONDING TO
10355 THESE VALUES, IN BOTH ASP/BSP HI/LO SP'S.
10356
10357 NOTE: THIS SUBR MUST BE ENTERED WITH BUSDIN_EMIT AND
10358 EN CLK IR PROC MCONS ENABLED
10359
10360 ALSO: THE WRITE FUNCTION USED IS: WR(AB,M,A) OR WR(AB,L,A)
10361 WITH A-ADDR = SF OR DF, RESPECTIVELY
10362
10363
10364 7036: 1(FREE)
10365 ZFROSFO2DF04: I SF=2, DF=4
10366 P2-U, IR_EMIT, EMIT/000204, I GO ZERO
10367 NEXT, J/ZEP06FDF I
10368 (7036) DCS[0.00,0.0,0.0,0] BM10000..00.00..00.10..000..100...0.0.0.0..0.0.0.0.1.1010...0.0000.0...11.000...000.100.010]

10369 7041: 1(FREE)
10370 ZFROSFO4DF02: I SF=4, DF=2
10371 P2-U, IR_EMIT, EMIT/000402, I GO ZERO
10372 NEXT, J/ZEP06FDF I
10373 (7041) DCS[0.00,0.0,0.0,0] BM10000..00.00..01.00..000..010...0.0.0.0..0.0.0.0.1.1010...0.0000.0...11.000...000.100.010]

10374 7042: 1(FREE)
10375 ZEROSFDF:
10376 P2-T, D_ZERO, SAVE-D[C], I ZEROS
10377 P3, A=RSPI[DF]_D, I

```

```

10378      NEXT,      J/ZEROSDFA
(7042) DCS(0,00,0,0,0,0) BM(0011,00,00,00,10,000,011,0,0,0,0,0,0000,0,0,1011,0,0,11,000,0,000,100,101)
10379
10380      7043: I(FREE)
10381      ZEROSDFA:
10382      P3,      A=BSPL0[DF]_D,
10383      NEXT,      J/ZEROSF
(7043) DCS(0,00,0,0,0,0) BM(0000,00,00,00,10,000,000,0,0,0,0,0,0,0000,0,0,0011,0,0,11,000,0,000,100,100)
10384
10385      7044: I(FREE)
10386      ZEROSF:
10387      P2-T,      D_ZERO, SAVE-DIC),
10388      P3,      A=BSPHI(SF)_D,
10389      NEXT,      J/ZEROSFA
(7044) DCS(0,00,0,0,0,0) BM(0011,00,01,00,11,000,011,0,0,0,0,0,0,0000,0,0,1011,0,0,11,000,0,000,100,101)
10390
10391      7045: I(FREE)
10392      ZEROSFA:
10393      P3,      A=BSPL0(SF)_D,
10394      NEXT,      BUTA(RETURN),
10395      J/BUTERROR7
(7045) DCS(0,00,0,0,0,0) BM(0000,00,01,00,11,000,000,0,0,0,0,0,0,0,0000,0,0,0011,0,0,11,111,0,011,111,110)
10396
10397
10398
10399
10400
10401      THIS SECOND SUBROUTINE NOW READS BACK ALL THE REGISTERS IN ASP/BSF,
10402      HI/LO, SF/DF THAT THE WRITE COULD HAVE REFERENCED,
10403      IF LOCATION WAS WRITTEN, BIT<07> SHOULD BE SET,
10404      ELSE IT SHOULD REMAIN CLEAR (FROM ZEROSFIXDFIX ROUTINE)
10405      THE BITS ARE SHIFTED SERIALY INTO THE SR, AND AFTER
10406      TWO WRITES (WHICH GIVES 16. BITS, THE SR JUST FILLED),
10407      THE COMPARE IS MADE BETWEEN EXPECTED/RECEIVED ANSWERS.
10408
10409      NOTE: THIS SUBR REQUIRES SR-LEFT/GUARD-DISABLED SETUP,
10410      AND CSP(02) CONTAINS EXPECTED ANSWER,
10411      CSP(00) CONTAINS RETURN ADDRESS, IN BITS<14:03>.
10412
10413
10414
10415      5746:
10416      SFDFOTOSR:
10417      PO,      LOAD-ENUA(ZTARGET777),
10418      LOAD-ERROR(SFDFOTOSR),
10419      DCS-CTR(C11,),
10420      NEXT,      GOTO-PAGE(77),
10421      J/SFDFOTOSRAA
(5746) DCS(1,00,1,0,0,0) BM(0100,00,11,11,11,11,11,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0,000,100,000)
10422
10423      7040: I(FREE)
10424      SFDFOTOSRA:
10425      P2-T,      D=ASPHI[DF], DIC=ALU07,
10426      NEXT,      J/SFDFOTOSRA
10427
10428
10429
10430
10431
10432
10433
10434
10435
10436
10437
10438
10439
10440
10441
10442
10443
10444
10445
10446
10447
10448
10449
10450
10451
10452
10453
10454
10455
10456
10457
10458
10459
10460
10461
10462
10463
10464
10465
10466
10467
10468
10469
10470
10471
10472
10473
10474
10475
10476
10477
10478
10479
10480
10481
10482
10483
10484
10485
10486
10487
10488
10489
10490
10491
10492
10493
10494
10495
10496
10497
10498
10499
10500
10501
10502
10503
10504
10505
10506
10507
10508
10509
10510
10511
10512
10513
10514
10515
10516
10517
10518
10519
10520
10521
10522
10523
10524
10525
10526
10527
10528
10529
10530
10531
10532
10533
10534
10535
10536
10537
10538
10539
10540
10541
10542
10543
10544
10545
10546
10547
10548
10549
10550
10551
10552
10553
10554
10555
10556
10557
10558
10559
10560
10561
10562
10563
10564
10565
10566
10567
10568
10569
10570
10571
10572
10573
10574
10575
10576
10577
10578
10579
10580
10581
10582
10583
10584
10585
10586
10587
10588
10589
10590
10591
10592
10593
10594
10595
10596
10597
10598
10599
10600
10601
10602
10603
10604
10605
10606
10607
10608
10609
10610
10611
10612
10613
10614
10615
10616
10617
10618
10619
10620
10621
10622
10623
10624
10625
10626
10627
10628
10629
10630
10631
10632
10633
10634
10635
10636
10637
10638
10639
10640
10641
10642
10643
10644
10645
10646
10647
10648
10649
10650
10651
10652
10653
10654
10655
10656
10657
10658
10659
10660
10661
10662
10663
10664
10665
10666
10667
10668
10669
10670
10671
10672
10673
10674
10675
10676
10677
10678
10679
10680
10681
10682
10683
10684
10685
10686
10687
10688
10689
10690
10691
10692
10693
10694
10695
10696
10697
10698
10699
10700
10701
10702
10703
10704
10705
10706
10707
10708
10709
10710
10711
10712
10713
10714
10715
10716
10717
10718
10719
10720
10721
10722
10723
10724
10725
10726
10727
10728
10729
10730
10731
10732
10733
10734
10735
10736
10737
10738
10739
10740
10741
10742
10743
10744
10745
10746
10747
10748
10749
10750
10751
10752
10753
10754
10755
10756
10757
10758
10759
10760
10761
10762
10763
10764
10765
10766
10767
10768
10769
10770
10771
10772
10773
10774
10775
10776
10777
10778
10779
10780
10781
10782
10783
10784
10785
10786
10787
10788
10789
10790
10791
10792
10793
10794
10795
10796
10797
10798
10799
10800
10801
10802
10803
10804
10805
10806
10807
10808
10809
10810
10811
10812
10813
10814
10815
10816
10817
10818
10819
10820
10821
10822
10823
10824
10825
10826
10827
10828
10829
10830
10831
10832
10833
10834
10835
10836
10837
10838
10839
10840
10841
10842
10843
10844
10845
10846
10847
10848
10849
10850
10851
10852
10853
10854
10855
10856
10857
10858
10859
10860
10861
10862
10863
10864
10865
10866
10867
10868
10869
10870
10871
10872
10873
10874
10875
10876
10877
10878
10879
10880
10881
10882
10883
10884
10885
10886
10887
10888
10889
10890
10891
10892
10893
10894
10895
10896
10897
10898
10899
10900
```

```

10426 (7046) DCS(0,00,0,0,0,0) BM(1111,00,00,11,10,000,011,0,0,1,0,0,0,0,0000,0,0,0000,0,000,100,111)
10427 7047: 1(FREE)
10428 SFDFTOSR4:
10429 P2-T, SR_SR-LEFT-1, ISTORE PAST
10430 D_ASPLO(DF), D[C]_ALU07, ISR<14,06> = ASPLO/DF
10431 NEXT, J/SFDFTOSR4
10432 (7047) DCS(0,00,0,0,0,0) BM(1111,00,00,10,10,000,011,0,0,1,1,0,0,0,0,0000,0,0,0000,0,000,101,000)
10433 7050: 1(FREE)
10434 SFDFTOSR4:
10435 P2-T, SR_SR-LEFT-1, ISTORE PAST
10436 D_BSPHI(DF), D[C]_ALU07, ISR<13,05> = BSPHI/DF
10437 NEXT, J/SFDFTOSR4
10438 (7050) DCS(0,00,0,0,0,0) BM(1010,01,00,00,00,000,011,0,0,1,1,0,0,0,0,0000,0,0,0000,0,000,101,001)
10439 7051: 1(FREE)
10440 SFDFTOSR4:
10441 P2-T, SR_SR-LEFT-1, ISTORE PAST
10442 D_BSPHI(DF), D[C]_ALU07, ISR<12,04> = BSPHI/DF
10443 NEXT, J/SFDFTOSR4
10444 (7051) DCS(0,00,0,0,0,0) BM(1010,00,00,00,00,000,011,0,0,1,1,0,0,0,0,0000,0,0,0000,0,000,101,010)
10445 7052: 1(FREE)
10446 SFDFTOSR4:
10447 P2-T, SR_SR-LEFT-1, ISTORE PAST
10448 D_ASPHI(SF), D[C]_ALU07, ISR<11,03> = ASPHI/SF
10449 NEXT, J/SFDFTOSR4
10450 (7052) DCS(0,00,0,0,0,0) BM(1111,00,00,11,11,000,011,0,0,1,1,0,0,0,0,0000,0,0,0000,0,000,101,011)
10451 7053: 1(FREE)
10452 SFDFTOSR4:
10453 P2-T, SR_SR-LEFT-1, ISTORE PAST
10454 D_ASPLO(SF), D[C]_ALU07, ISR<10,02> = ASPLO/SF
10455 NEXT, J/SFDFTOSR4
10456 (7053) DCS(0,00,0,0,0,0) BM(1111,00,00,10,11,000,011,0,0,1,1,0,0,0,0,0000,0,0,0000,0,000,101,100)
10457 7054: 1(FREE)
10458 SFDFTOSR4:
10459 P2-T, SR_SR-LEFT-1, ISTORE PAST
10460 D_BSPHI(SF), D[C]_ALU07, ISR<09,01> = BSPHI/SF
10461 NEXT, J/SFDFTOSR4
10462 (7054) DCS(0,00,0,0,0,0) BM(1010,01,01,00,00,000,011,0,0,1,1,0,0,0,0,0000,0,0,0000,0,000,101,101)
10463 7055: 1(FREE)
10464 SFDFTOSR4:
10465 P2-T, SR_SR-LEFT-1, ISTORE PAST
10466 D_BSPLO(SF), D[C]_ALU07, ISR<08,00> = BSPLO/SF
10467 NEXT, J/SFDFTOSR4
10468 (7055) DCS(0,00,0,0,0,0) BM(1010,00,01,00,00,000,011,0,0,1,1,0,0,0,0,0000,0,0,0000,0,000,101,110)
10469 7056: 1(FREE)
10470 SFDFTOSR4:
10471 P2-T, SR_SR-LEFT-1, ISTORE PAST

```

```
10472      D_CSPD(D00), D(C)_D,      RETRIEVE RETURN ADDRESS
10473      NEXT, J/SFDFOTOSRI
(7056) DCS[0.00.0.0.0.0] BM[1010..10.00..00.00..0000..0000..0.1.1.0.0..0.1111..0.0000.0...11.000..000.101.111]
10474
10475      7057: I(FREE)
10476      SFDFOTOSRI
10477      P0,      RETURN_D[14-03], PAGE(7),      INPUT RETURN ADDRESS INTO RETURN REGISTER,
10478      P2-T, D_SP-XOR-CSPD(D02), SAVE-D(C),      ICOMPARE RECEIVED;EXPECTED
10479      NEXT, J/STARGET777      IGO DO A BUTA(RETURN)
(7057) DCS[0.00.0.0.0.0] BM[0110..10.00..00.00..0000..111..0.1.0.0..0.1101..0.0000.0...11.101...111.111.111]
10480
10481
10482
10483
10484
10485      1.PAGE=====
10486
10487
10488      .TOC * TEST375-376: BYTE WRITE TO ASP/BSP LO, SP ADDR R-OR-1/FLTPT-INHIBIT
10489
10490
10491      I=====
10492      I*
10493      I*      TESTS: 375A - 376      UNWORDS: 020 + 017
10494      I*
10495      I*      FUNCTIONS:
10496      I*
10497      I*      THE FOLLOWING SET OF THREE CHECKS TESTS THE FOLLOWING FEATURES OF THE ASP/BSP:
10498      I*
10499      I*      TEST 375 A/B TEST THAT BYTE WRITES CAN BE PERFORMED TO ASP/BSP LO, USING THE
10500      I*      "DAD" EXTENSION BIT COMBINATION.
10501      I*
10502      I*      TEST 376 VERIFIES THAT A FLOATING POINT INSTRUCTION IN THE IR FORCES SF-ADDRS
10503      I*      BIT<02> TO A (0), AND THAT BUTA(R-OR-1) IN THE REFERENCING MICROWORD FORCES
10504      I*      BIT<00> TO A (1).
10505      I*
10506      I=====
10507
10508
10509
10510      ITEST 375 A CHECKS THAT THE DAD BIT COMBINATION /11 DOES A BYTE WRITE (IE, LO BYTE ONLY)
10511      5517:
10512      TEST375A:
10513      P0,      LOAD=ENUA(STARGET432),      ISETUP FOR INSTRS IR=(000125)/E78/432
10514      LOAD=ERROR(TEST375A),      IERROR DIRECTORY KEY
10515      DCS=CTR(C11.),      ICOMPARE AT TARGET
10516      NEXT, J/SETSP375A
(5517) DCS[1.00.1.0.0.0] BM[0100..00.11..11.00..011..010...0.0.0.0..0.0000...0.0000.0...11.000...101.000.110]
10517
10518      5506:
10519      SETSP375A:
10520      P2-T, D_CSPD(C125252),      IFIRST SETUP SP'S WITH JUNK
10521      P3, A$BSPLO[06]_D=[A],      I
10522      NEXT, GOTO=PAGE(7),      IXFER FOR DCS-DAD BITS
```

```
10523      J/FIRST375A      I
(5506) DCS[0.00.0.0.0.0] BM[1010..10.00..00.00..111..111...0.1.0.0..0.0110...0.0011.0...11.100...000.100.110]
10524
10525      7046: I(FREE)
10526      FIRST375A:
10527      SETUP, FIRST-1-OR-2,      IDAD/01, SHOULDNT CAUSE BYTE WRITE
10528      P2-T, D_ZERO,      I SHOULD WRITE ALL ZEROS
10529      P3, ASPLO[06]_D,      I
10530      NEXT, J/SECOND375A
(7046) DCS[0.01.0.0.0.0] BM[0011..00.00..00.00..111..000...0.1.0.0..0.0000...0.0001.0...11.000...000.110.001]
10531
10532      7061: I(FREE)
10533      SECOND375A:
10534      SETUP, SECOND-1-OR-2,      IDAD/10, SHOULDNT CAUSE BYTE WRITE
10535      P2-T, D_ZERO,      I SHOULD WRITE ALL ZEROS
10536      P3, BSPLO[06]_D,      I
10537      NEXT, J/BYTE375A
(7061) DCS[0.10.0.0.0.0] BM[0011..00.10..00.00..111..000...0.1.0.0..0.0000...0.0110.0...11.000...000.110.010]
10538
10539      7062: I(FREE)
10540      BYTE375A:
10541      SETUP, BYTE-WRITE,      IDAD/11, SHOULD CAUSE A BYTE WRITE
10542      P2-T, D_ASPI(C052525),      IWRITE THE (125) IN THE LOW BYTE
10543      P3, A$BSPLO[06]_D=[B],      IUPPER BYTE SHOULD BE (000) FROM ABOVE
10544      NEXT, J/CHECK375A
(7062) DCS[0.11.0.0.0.0] BM[1111..00.10..11.01..111..000...0.1.0.0..0.0000...0.0111.0...11.000...000.110.011]
10545
10546      7063: I(FREE)
10547      CHECK375A:
10548      P2-T, D_ASPI(06),      IGET THE A SIDE SP
10549      NEXT, J/GOBUT375A
(7063) DCS[0.00.0.0.0.0] BM[1111..00.00..10.00..111..000...0.1.0.0..0.0000...0.0000.0...11.000...000.110.100]
10550
10551      7064: I(FREE)
10552      GOBUT375A:
10553      SETUP, RETURN/TEST375B,      IRETURN TO START OF NEXT SUBTEST
10554      NEXT, CALL(DINTOIR=5)      IGO CHECK (000125) OBTAINED
(7064) DCS[0.00.0.0.0.0] BM[0101..00.11..00.10..101..111...0.0.0.0..0.0000...0.0000.0...11.100...010.111.011]
10555
10556
10557
10558
10559
10560
10561      ITEST 375 B NOW CHECKS THE SAME THING (000125) IS ON THE B SIDE
10562      5625:
10563      TEST375B:
10564      P0,      LOAD=ENUA(STARGET432),      ISETUP FOR INSTRS IR=(000125)/E78/432
10565      LOAD=ERROR(TEST375B),      IERROR DIRECTORY KEY
10566      DCS=CTR(C7.),      ICOMPARE AT TARGET
10567      NEXT, J/CHECK375B
(5625) DCS[1.00.1.0.0.0] BM[1000..00.11..11.00..011..010...0.0.0.0..0.0000...0.0000.0...11.000...011.000.000]
10568
```

```

10569 5300: 1(FREE)
10570 CHECK375B:
10571 P2-T, D_BSPLO(R06), |GET THE B SIDE SP
10572 NEXT, J/GOBUT375B |
(5300) DCS[0.00,0.0,0.0] BM[1010.00,10.00,00.111.000...0.1,0.0.0...0.0000...0.0000,0...11.000...011.000,001]
10573
10574 5301: 1(FREE)
10575 GOBUT375B:
10576 SETUP, RETURN/TEST376A, |RETURN TO START OF NEXT SUBTEST
10577 NEXT, CALL[DINTOIR-5] |GO CHECK (000125) OBTAINED
(5301) DCS[0.00,0.0,0.0] BM[0101.00,10.01...011.111...0.0,0.0.0...0.0000...0.0000,0...11.100...010.111,011]
10578
10579
10580
10581
10582
10583 | - - - - -
10584
10585
10586 |TEST 376 A NOW DOES THE SF ADDRESS MODE, W/ FLTP-T-INHIBIT AND BUTA(R-OR-1) ACTIVE
10587 |NOTE:
10588 | THE ASPHI CONTAINS THE FOLLOWING VALUES IN THESE LOCATIONS:
10589 |
10590 | ASPHI(02) ASPHI(03) ASPHI(06) ASPHI(07)
10591 | (052522) (177777) (056166) (052828)
10592 |
10593
10594 5513:
10595 TEST376A:
10596 PO, LOAD-ENUA(DOIT376A), |MAKE SURE BUTA(R-OR-1) DOESN'T CAUSE A BRANCH
10597 LOAD-ERRDIR(TEST376A), |ERROR DIRECTORY KEY
10598 DCS-CTR(C2), |COMPARE AT TARGET
10599 NEXT, J/SETIR376A |
(5513) DCS[1.00,1.0,0.0] BM[1101.00,10.11.00...100.000...0.0,0.0.0...0.0000...0.0000,0...11.000...011.000,010]
10600
10601 5302: 1(FREE)
10602 SETIR376A:
10603 SETUP, BUTA(R-OR-1), |SETUP ACTIVE BUT MODIFICATION OF SF BIT<00> ADDRESS
10604 P2-U, IR_EMIT, EMIT/170600, |IR=FLTP-T INSTR, SF=(6)
10605 NEXT, J/DOIT376A |
(5302) DCS[0.00,0.0,0.0] BM[1111.00,00.01.10...000.000...0.0,0.0.0...1.1010...0.0000,0...10.010...100.100,000]
10606
10607 5440:
10608 DOIT376A:
10609 P2-T, D_A-XOR-B, |COMPARE SF/OBTAINED; REGISTER EXPECTED
10610 BUS-A_ASPHI(SF), |SF ON A GOES FROM (6) -> (2) FROM FLTP-T,
10611 |AND FROM (2) -> (3) FOR (R-OR-1)
10612 BUS-B_ASPHI(R03), |THIS WE EXPECT
10613 NEXT, J/TEST376A |
(5440) DCS[0.00,0.0,0.0] BM[0110.01.11.11...101.000...0.1,0.0.0...0.0000...0.0000,0...11.000...110.001,100]
10614
10615 | - - - - -
10616

```

```

10617 |NOW CHECK THE RIGHT RESULT WAS OBTAINED
10618 5614:
10619 TEST376A1:
10620 PO, LOAD-ENUA(ZTARGET434), |SETUP FOR IR=(000000)/INSTRS COMPARE
10621 LOAD-ERRDIR(TEST376A1), |ERROR DIRECTORY KEY
10622 DCS-CTR(C6), |COMPARE AT TARGET
10623 NEXT, J/GOBUT376A |
(5614) DCS[1.00,1.0,0.0] BM[1001.00,11.11.00...011.100...0.0,0.0.0...0.0000...0.0000,0...11.000...011.000,011]
10624
10625 5303: 1(FREE)
10626 GOBUT376A:
10627 SETUP, RETURN/SCOPE376, |RETURN TO SCOPE LOOP TEST WORD
10628 NEXT, CALL[DINTOIR-5] |CHECK FOR (000000)
(5303) DCS[0.00,0.0,0.0] BM[0101.00,01.10.00...100.111...0.0,0.0.0...0.0000...0.0000,0...11.100...010.111,011]
10629
10630
10631
10632 5304: 1(FREE)
10633 SCOPE376:
10634 PO, BUMP-VERIFY, |COUNT
10635 NEXT, BUTD[SCOPE], |NO ERROR: "TEST410" (+1, WORDS)
10636 J/TEST410 |ERROR: "SETSP376A" (-12, WORDS)
(5304) DCS[0.00,0.1,0.1] BM[0000.00,00.00.00...000.000...0.0,0.0.0...0.0000...0.0000,0...11.000...101.000,111]
10637
10638
10639
10640
10641
10642 |.PAGE=====
10643
10644
10645 |.TNC * TEST410: BYTE/BYTE CONSTANT/D=ZERO
10646
10647
10648 |*****
10649 |*
10650 |* TESTS: 410 A - E UNWORDS: 044 + 020
10651 |*
10652 |* FUNCTIONS:
10653 |*
10654 |* THE FOLLOWING TESTS RUN A COUNT PATTERN THRU THE IR, MAINTAINING TOTALS OF
10655 |* THE NUMBER OF TIMES:
10656 |*
10657 |* BYTE-H=LOW, BYTE/1-OR-2-FIRST=HIGH, BYTE/1-OR-2-SECOND=HIGH,
10658 |* (D=ZERO)-H=HIGH, AND (D=ZERO)-H=LOW.
10659 |*
10660 |* AT THE END, THE TESTS COMPARE THE EXPECTED COUNTS TO THE RECEIVED COUNTS.
10661 |*
10662 |*****
10663
10664
10665 5507:
10666 TEST410:

```

```
10667      PO,      LOAD-ERROR(TEST410),      |ERROR DIRECTORY KEY
10668      DCS-CTR(C6.),      |COMPARE BELOW
10669      NEXT,      GOTO-PAGE(6),      |XFER
10670      J/SETBYTE410
(5507) DCS(1.00,1.0,0,0) BM(1001.00.00.00.00.000.110...0,0,0,0,0,0,0.0000...0.0000,0...11,100...101,110,000)
10671
10672      6560:
10673      SETBYTE410:
10674      P2-T,      D_ZERO,      |ZEROS INTO:
10675      P3,      A$BSPHI(10)_D-(A),      |ASPHI(10) = BYTE-FIRST
10676      NEXT,      GOTO-PAGE(7),      |
10677      J/SETBYTE410      |B$PHI(10) = WORD (n-BYTE)
(6560) DCS(0.00,0,0,0,0) BM(0011.00.00.00.00.000.111...0,1,0,0,0,0,0.0000...0.1011,0...11,100...000,110,000)
10678
10679      7060: 1(FREE)
10680      SETBYTE410:
10681      P3,      A$BSPLO(10)_D-(B),      |ASPL(10) = BYTE-SECOND
10682      NEXT,      J/SETBYTE410      |B$PLO(10) = IR-DATA
(7060) DCS(0.00,0,0,0,0) BM(0000.00.10.00.00.000.000...0,0,0,0,0,0,0.0000...0.0111,0...11,000...000,110,110)
10683
10684      7066: 1(FREE)
10685      SETBYTE410:
10686      P3,      A$BSPLO(11)_D-(B),      |ASPL(11) = D-NONZERO
10687      NEXT,      GOTO-PAGE(6),      |B$PLO(11) = D-ZERO
10688      J/SETBYTE410      |
(7066) DCS(0.00,0,0,0,0) BM(0000.00.11.00.00.000.110...0,0,0,0,0,0,0.0000...0.0111,0...11,100...000,000,000)
10689
10690      6000: 1(FREE)
10691      SETBYTE410:
10692      PO,      BUMP-VERIFY,      |COUNT
10693      P3,      CSPD(17)_EMIT, EMIT/000001,      |A (1) IN BYTE-CONSTANT
10694      NEXT,      J/SETBYTE410      |LOCATION IN CSP
(6000) DCS(0.00,0,0,0,1) BM(0000.10.00.00.00.000.001...0,0,0,0,0,0,0.0000...1.0000,0...11,000...001,000,101)
10695
10696      6105: 1(FREE)
10697      SETBYTE410:
10698      P2-U,      IR_DBUF-[I],      |SETUP UCONS FOR D --> IR
10699      P3,      DBUF_D-[I],      |
10700      NEXT,      J/TESTD410      |
(6105) DCS(0.00,0,0,0,0) BM(0100.00.00.00.01.000.100...0,0,0,0,0,0,1.1011...0.0000,0...11,000...111,111,011)
10701
10702
10703
10704      *** LOOP BACK ENTRY POINT ***
10705
10706      6773:
10707      TESTD410:
10708      PO,      LOAD-ENUA(GOFOR410),      |COMPARE AT END OF LOOP
10709      LOAD-ERROR(TESTD410),      |ERROR DIRECTORY KEY
10710      DCS-CTR(C15.),      |RELOAD EACH TIME THROUGH
10711      P2-U,      IR_DBUF,      |DON'T CARE HERE
```

```
10712      P3,      DBUF_D,      |COPY IR-DATA FROM D --> DBUF
10713      NEXT,      BUTR(D14-00-EG-0),      |TEST D<14:00>H:
10714      J/DNONZERO410      |ZERO --> DZERO410, NONZERO --> DNONZERO410
(6773) DCS(1.00,1,0,0,0) BM(0000.00.11.10.11.110.001...0,0,0,0,0,0,1.1010...0.0000,0...01,101...101,011,001)
10715
10716      ENTER HERE IF D<14:00> WAS DETECTED AS NON-ZERO
10717      6531:
10718      DNONZERO410:
10719      P2-T,      D_ASPL(DNONZERO)-PLUS-1,      |BUMP NON ZERO COUNTER
10720      P2-U,      IR_DBUF,      |COPY IR-DATA FROM DBUF --> IR
10721      P3,      ASPL(11)_D,      |SAVE NON ZERO COUNTER
10722      DBUF_D,      |DON'T CARE HERE)
10723      NEXT,      GOTO-PAGE(7),      |XFR TO 7 FOR DAD
10724      J/NEXTPAT410      |
(6531) DCS(0.00,0,0,0,0) BM(1001.01.11.10.01.000.111...0,1,0,0,0,0,1.1010...0.0001,0...11,100...000,110,101)
10725
10726      ENTER HERE IF D<14:00> WAS DETECTED AS ZERO
10727      6533:
10728      DZERO410:
10729      P2-T,      D_BSPLO(DZERO)-PLUS-1,      |BUMP ZERO COUNTER
10730      P2-U,      IR_DBUF,      |COPY IR-DATA FROM DBUF --> IR
10731      P3,      BSPLO(11)_D,      |SAVE ZERO COUNTER
10732      DBUF_D,      |DON'T CARE NOW)
10733      NEXT,      GOTO-PAGE(7),      |XFR TO 7 FOR DAD
10734      J/NEXTPAT410      |
(6533) DCS(0.00,0,0,0,0) BM(1001.00.11.11.01.000.111...0,1,0,0,0,0,1.1010...0.0110,0...11,100...000,110,101)
10735
10736      7065: 1(FREE)
10737      NEXTPAT410:
10738      SETUP,      FIRST-1-OR-2,      |SELECT DAD BITS FOR "BYTEFIRST410"
10739      P2-T,      D_BSPLO(IR-DATA)-PLUS-2,      |INCRE DATA FOR -NEXT- TIME THRU)
10740      D(C)_COUNT,      |WHEN THIS SETS WE'RE DONE
10741      P3,      BSPLO(10)_D,      |SAVE NEXT
10742      NEXT,      J/BYTEFIRST410      |
(7065) DCS(0.01,0,0,0,0) BM(1100.00.10.11.01.000.110...0,1,0,0,0,0,0.0000...0.0110,0...11,000...000,111,000)
10743
10744      7070: 1(FREE)
10745      BYTEFIRST410:
10746      SETUP,      SECOND-1-OR-2,      |SELECT DAD BITS FOR "BYTESECOND410"
10747      P2-T,      D_ASPL(BYTE-FIRST)-PLUS-CSP(1-0),      |BYTE-FIRST SELECTS EITHER
10748      SAVE-D(C),      |CSP(17)=1 OR CSP(13)=0
10749      P3,      ASPHI(10)_D,      |WRITE BACK
10750      NEXT,      BUTR(BYTE),      |BYTE --> "BYTESECOND410"
10751      J/WORD410      |BYTE --> "WORD410"
(7070) DCS(0.10,0,0,0,0) BM(1001.10.00.11.00.000.111...0,1,0,0,0,0,0.0100...0.1001,0...01,001...011,110,110)
10752
10753      ENTER HERE IF BYTE-N NOT ASSERTED, IE IR=(WORD)
10754      7366:
10755      WORD410:
10756      SETUP,      SECOND-1-OR-2,      |SELECT DAD BITS FOR "BYTESECOND410"
10757      P2-T,      D_ASPL(WORD)-PLUS-1,      |BUMP WORD=BYTE COUNTER
10758      SAVE-D(C),      |SAVE PAST CARRYOUT STATUS
```

```
10759      P3,      BSPHI(10)D,      IWRITE BACK
10760      NEXT,      J/BYTESECOND410      INOW GO TRY BYTE-SECOND
(7366) DCS(0.10.0.0.0.0) BM(1001.01.10.11.01.000.111...0.1.0.0.0.0.0.0000...0.1110.0...11.000...011.110.111)
10761      IENTER HERE IF BYTE-H WAS ASSERTED, IE IR=(BYTE)
10762      7367:
10763      BYTESECOND410:
10764      SETUP,      NO-DAD,      IKEEP FOR NOISE
10765      P2=T,      D_ASPLD(BYTE-SECOND)+PLUS-CSP(1-0),      IBYTE-SECOND SELECTS EITHER
10766      SAVE-D(C),      ICP(17)=1 OR CSP(13)=0
10767      P3,      ASPLD(10)D,      IWRITE BACK
10768      NEXT,      BUTR(D(C)-B),      IIF SET, SKIP OUT TO TEST-410A
10770      J/GOPOR410      IIF CLEAR, FALL THRU TO NEXT
(7367) DCS(0.00.0.0.0.0) BM(1001.10.00.10.00.000.111...0.1.0.0.0.0.0.0100...0.0001.0...10.011...011.110.001)
10771      7361:
10772      GOPOR410:
10773      P2=T,      D_BSPLO(IR=DATA),      IGET DATA FOR IR INTO D
10774      NEXT,      GOTO=PAGE(6),      I
10775      J/TESTD410      ILOOP BACK FOR NEXT
(7361) DCS(0.00.0.0.0.0) BM(1010.00.10.00.00.000.110...0.1.0.0.0.0.0.0000...0.0000.0...11.100...111.111.011)
10777
10778
10779
10780
10781      I - - - - -
10782
10783      ITEST 410A CHECKS THAT D<14100>=ZERO WAS ONLY ASSERTED TWICE
10784      7363:
10785      EXPEC410A:
10786      NEXT,      GOTO=PAGE(6),      IFOR LOADING DCS=CTR
10787      J/TEST410A      I
(7363) DCS(0.00.0.0.0.0) BM(0000.00.00.00.00.000.110...0.0.0.0.0.0.0.0000...0.0000.0...11.100...111.001.001)
10788      4711:
10789      TEST410A:
10790      PO,      LOAD-ENUA(ZTARGET422),      IFOR IR=(000002) W/INSTR5
10791      LOAD-ERROR(TEST410A),      IERROR DIRECTORY KEY
10792      DCS-CTR(C7),      ICOMPARE AT TARGET
10793      NEXT,      J/COMP410A      I
10794      (6711) DCS(1.00.1.0.0.0) BM(1000.00.11.11.00.010.010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.000.110)
10795      6106: I(FREE)
10796      COMP410A:
10797      P2=T,      D_BSPLO(DZERO),      IGET DATA
10798      NEXT,      J/INTOTR410A      I
(6106) DCS(0.00.0.0.0.0) BM(1010.00.11.00.00.000.000...0.1.0.0.0.0.0.0000...0.0000.0...11.000...001.000.111)
10800      6107: I(FREE)
10801      INTOTR410A:
10802      SETUP,      RETURN/TEST410B,      ICOPY D --> IR, RESET BUSDIN_EMIT
10803      NEXT,      CALL(DINTOTR=5)      I AND CHECK ITS ALL ZERO
10804
```

```
(6107) DCS(0.00.0.0.0.0) BM(0110.00.11.10.00.100.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.111.011)
10805
10806
10807
10808
10809      I - - - - -
10810
10811      ITEST 410B CHECKS THAT D<14100>=ZERO WAS NOT ASSERTED
10812      I 32768.-2, = 32766, (077776) TIMES
10813      6704:
10814      TEST410B:
10815      PO,      LOAD-ENUA(ZTARGET434),      IFOR IR=(000000) W/INSTR5
10816      LOAD-ERROR(TEST410B),      IERROR DIRECTORY KEY
10817      DCS-CTR(C8),      ICOMPARE AT TARGET
10818      BUMP-VERIFY,      ICOUNT
10819      NEXT,      J/EXPEC410B      I
(6704) DCS(1.00.1.0.0.0) BM(0111.00.11.11.00.011.100...0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.001.000)
10820      6110: I(FREE)
10821      EXPEC410B:
10822      P3,      CSPD(17)_EMIT, EMIT/077776,      IEXPECTED NUMBER OF TIMES
10823      NEXT,      J/COMP410B      ID<14100> WAS NON ZERO
(6110) DCS(0.00.0.0.0.0) BM(0111.10.11.11.11.111.110...0.0.0.0.0.0.0.0000...1.0000.0...11.000...001.001.001)
10824
10825      6111: I(FREE)
10826      COMP410B:
10827      P3=T,      D_ASPLD(DNONZERO)-MINUS-CSPD(17),      ICOMPARE RECEIVED:EXPECTED
10828      NEXT,      J/GOPUT410B      I
(6111) DCS(0.00.0.0.0.0) BM(1101.10.00.10.01.000.000...1.1.0.0.0.0.0.0000...0.0000.0...11.000...001.001.010)
10830      6112: I(FREE)
10831      GOPUT410B:
10832      SETUP,      RETURN/TEST410C,      IGO PUT D --> IR
10833      NEXT,      CALL(DINTOTR=5)      I AND CHECK IT'S ALL ZERO
(6112) DCS(0.00.0.0.0.0) BM(0110.00.11.10.00.111.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.111.011)
10835
10836
10837
10838
10839      I - - - - -
10840
10841      ITEST 410C CHECKS THAT BYTE-H WAS NOT ASSERTED 21696,,
10842      IOP WAS ASSERTED 11072, TIMES IN 32768, ITERATIONS
10843      6707:
10844      TEST410C:
10845      PO,      LOAD-ENUA(ZTARGET402),      ISETUP FOR D=ZERO TEST
10846      LOAD-ERROR(TEST410C),      IERROR DIRECTORY KEY
10847      DCS-CTR(C6),      ICOMPARE AT TARGET
10848      NEXT,      J/EXPEC410C      I
(6707) DCS(1.00.1.0.0.0) BM(1001.00.11.11.00.000.010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.001.011)
10849      6113: I(FREE)
10850
```

```
10851 EXPEC410C:
10852 P3, CSPD[17]_EMIT, EMIT/52300, 132768,-9TIMES BYTE-M ASSERTED
10853 NEXT, J/ASIDE410C 1 21896,-9(52300)
(6113) DCS[0.00,0.0,0.0] BM[0101..10.01..00.11..000..000..0.0.0..0.0..0.0000...1..0000,0...11.000...001.001,100]
10854 6114: 1(FREE)
10855 ASIDE410C:
10856 P2-T, SR_CSPD(D17), 1GET ONTO A-SIDE
10857 NEXT, J/COMP410C 1
10858 (6114) DCS[0.00,0.0,0.0] BM[1010..10.00..00.00..000..000..0.0.1..0.0..0.0000...0..0000,0...11.000...001.001,101]
10859 6115: 1(FREE)
10860 COMP410C:
10861 P3-T, D_SR-MINUS-BSPHI(WORD), 1COMPARE EXPEC:RECEIVED
10862 NEXT, J/GOBUT410C 1
10863 (6115) DCS[0.00,0.0,0.0] BM[1101..01.10..00.00..000..000...1.1.0..0.0..0.0000...0..0000,0...11.000...001.001,110]
10864 6116: 1(FREE)
10865 GORUT410C:
10866 SETUP, RETURN/TEST410D, 1RETURN TO START OF NEXT SUBTEST
10867 NEXT, GOTO-PAGE(7), 1BUT TABLE
10868 J/BUTD-18-ZERO 1CHECK EQUALITY
10869 (6116) DCS[0.00,0.0,0.0] BM[0110..00.11..10.10..111..111...0.0.0..0.0..0.0000...0..0000,0...11.100...011.100,001]
10870
10871
10872
10873
10874
10875
10876 1TEST 410D CHECKS THAT BYTE-CONSTANT WAS ASSERTED (4270)=2232.
10877 1TIMES UNDER "FIRST-1-OR-2"
10878 6727:
10879 TEST410D:
10880 P0, LOAD-ENUA(ZTARGET402), 1SETUP FOR D=ZERO TEST
10881 LOAD-ERROR(TEST410D), 1ERROR DIRECTORY KEY
10882 DCS-CTR(CS,), 1COMPARE AT TARGET
10883 NEXT, J/EXPEC410D 1
(6727) DCS[1.00,1.0,0.0] BM[1010..00.11..11.00..000..010...0.0.0..0.0..0.0000...0..0000,0...11.000...001.001,111]
10884 6117: 1(FREE)
10885 EXPEC410D:
10886 P0, RUMP-VERIFY, 1COUNT
10887 P3, CSPD[17]_EMIT, 1
10888 EMIT/004270, 1(4270)=2232, TIMES FOR FIRST
10889 NEXT, J/COMP410D 1
(6117) DCS[0.00,0.0,0.1] BM[0000..10.10..00.10..111..000...0.0.0..0.0..0.0000...1..0000,0...11.000...001.010,000]
10890 6120: 1(FREE)
10891 COMP410D:
10892 P3-T, D_ASPhi[BYTE-FIRST]-MINUS-CSPD[17], 1COMPARE RECEIVED:EXPEC
10893 NEXT, J/GOBUT410D 1
(6120) DCS[0.00,0.0,0.0] BM[1101..10.00..11.00..000..000...1.1.0..0.0..0.0000...0..0000,0...11.000...001.010,001]
10894
10895
```

```
10897 6121: 1(FREE)
10898 GORUT410D:
10899 SETUP, RETURN/TEST410E, 1RETURN TO START OF NEXT SUBTEST
10900 NEXT, GOTO-PAGE(7), 1BUT TABLE
10901 J/BUTD-18-ZERO 1CHECK EQUALITY
(6121) DCS[0.00,0.0,0.0] BM[0110..00.11..10.00..101..111...0.0.0..0.0..0.0000...0..0000,0...11.100...011.100,001]
10902
10903
10904
10905
10906
10907
10908 1TEST 410E CHECKS THAT BYTE-CONSTANT WAS ASSERTED (3600)=1920.
10909 1TIMES UNDER "SECOND-1-OR-2"
10910 6705:
10911 TEST410E:
10912 P0, LOAD-ENUA(ZTARGET402), 1SETUP FOR D=ZERO TEST
10913 LOAD-ERROR(TEST410E), 1ERROR DIRECTORY KEY
10914 DCS-CTR(CS,), 1COMPARE AT TARGET
10915 NEXT, J/EXPEC410E 1
(6705) DCS[1.00,1.0,0.0] BM[1010..00.11..11.00..000..010...0.0.0..0.0..0.0000...0..0000,0...11.000...001.010,010]
10916 6122: 1(FREE)
10917 EXPEC410E:
10918 P0, BUMP-VERIFY, 1COUNT
10919 P3, CSPD[17]_EMIT, 1
10920 EMIT/003600, 1(3600)=1920, TIMES FOR SECOND
10921 NEXT, J/COMP410E 1
(6122) DCS[0.00,0.0,0.1] BM[0000..10.01..11.10..000..000...0.0.0..0.0..0.0000...1..0000,0...11.000...001.010,011]
10922 6123: 1(FREE)
10923 COMP410E:
10924 P3-T, D_ASPhi[BYTE-SECOND]-MINUS-CSPD[17], 1COMPARE RECEIVED:EXPEC
10925 NEXT, J/GOBUT410E 1
(6123) DCS[0.00,0.0,0.0] BM[1101..10.00..10.00..000..000...1.1.0..0.0..0.0000...0..0000,0...11.000...001.010,100]
10926 6124: 1(FREE)
10927 GORUT410E:
10928 SETUP, RETURN/SCOPE410, 1RETURN TO SCOPE LOOP TEST WORD
10929 NEXT, GOTO-PAGE(7), 1BUT TABLE
10930 J/BUTD-18-ZERO 1CHECK EQUALITY
(6124) DCS[0.00,0.0,0.0] BM[0110..00.00..10.10..101..111...0.0.0..0.0..0.0000...0..0000,0...11.100...011.100,001]
10931
10932
10933
10934
10935
10936 6125: 1(FREE)
10937 SCOPE410:
10938 P0, BUSPIN_EMIT-[1], 1RESET PROC UCON
10939 EN-CLK-TR[15-00], 1
10940 NEXT, BUTD[SCOPE], 1NO ERROR: "TEST500" (+1, WORDS)
10941 J/TEST500 1ERROR: "SETBYTE410" (-34, WORDS)
(6125) DCS[0.00,0.1,0.0] BM[0000..00.00..00.01..000..100...0.0.0..0.0..1.1001...0..0000,0...11.000...101.110,001]
10942
```

```

10943      .
10944      .
10945      .
10946      .
10947      .
10948      .
10949      I,PAGE=====
10950
10951    |*** VERSION /V101A0/ ***
10952
10953    ,TDC * TEST500; PREFETCH/OVERLAP/SP DEFEAT
10954
10955    |*****
10956    |+
10957    |+   TESTS: 500C = 500F                               UWORDS; 034 + 013
10958    |+
10959    |+   FUNCTIONS:     TESTS 500C = 500F RUN PATTERNS THRU THE IR TO TEST
10960    |+                   THE PREFETCH/OVERLAP ROM & THE SP WRITE DEFEAT LOGIC.
10961    |+
10962    |+ *****
10963
10964
10965
10966
10967    6561:
10968    TEST500:
10969        PN,          LOAD=ENUA(SETFETCHG500),             INTERMEDIATE COMPARE AT START OF LOOP
10970                    LOAD=ERROR(TEST500).                  ERROR DIRECTORY KEY
10971                    DCS-CTR(C5.),                          !COMPARE AT TARGET
10972                    NEXT, J/SFTTFCHB500                     !
10973    (6561) DCS[1,00,1,0,0,0] BM[1010,,00,11,,00,01,,011,,001,,..0,0,0,..0,000...0.,0000,0...,11,000...,110,111,110]
10974
10975    6676:
10976    SETTFCHB500:
10977        P2-T,         D_ZERO, DIC!ALU15,                 !ZERO
10978                    SR_ZERO,                                !TO FLAG JANUPPS AS ILLEGAL -- KEEP ZERO THRU "TEST505"
10979                    P3, A#BSPLO(OVERLAP)_D,                !ZERO OVERLAP -A, -B COUNTERS
10980                    NEXT, J/SFTTFCHD500                      !
10981    (6676) DCS[0,00,0,0,0,0] BM[0011,,00,00,,10,00,,000,,100,...0,1,1,0,0,..0,000...0.,0011,0...,11,000...,001,010,110]
10982
10983    6126: !FREE)
10984    SETTFCHD500:
10985        P3,           A#BSPHI(PATTERN)_D,                 !ZERO PATTERN (A AND B)
10986                    NEXT, J/SEIFETCHES00                   !
10987    (6126) DCS[0,00,0,0,0,0] BM[0000,,00,00,,11,01,,011,,000,...0,0,0,..0,0...0,0000...0.,1011,0...,11,000...,001,010,111]
10988
10989    6127: !FREE)
10990    SETTFCHF500:
10991        P3,           A#BSPI[PREFETCH>_D,                 !ZERO PREFETCH=A COUNTER, ALSO B SIDE
10992                    NEXT, J/SETTFCHF500                     !
10993    (6127) DCS[0,00,0,0,0,0] BM[0000,,00,00,,11,00,,000,,000,...0,0,0,..0,0...0,0000...0.,1011,0...,11,000...,001,011,000]
10994
10995    6130: !FREE)
10996    SFTTFCHF500:

```

```

10993      P0,      BUMP-VERIFY,          !COUNT
10994      P1,      CSPD[17]_020010,      !
10995      NEXT,     J/SETFETCHG500        !MASK CONSTANT FOR DATA PATTERN
(6130) DCS(0,0,0,0,0,1) BM(0010,10,00,00,00,001,000,0,0,0,0,0,0,0,0000,1,0000,0,11,000,0,001,011,001)

10996
10997      6131: 1(FREE)
10998      SETFETCHG500:
10999      P0,      BUMP-VERIFY,          !COUNT
11000      P2=U,    IR_DBUF=DI,          !SETUP THESE UCON'S,
11001      P3,      DBUF_D=DI,          !DON'T WORRY ABOUT EFFECT HERE
11002      NEXT,     J/TESTD500          !
(6131) DCS(0,0,0,0,0,1) BM(0100,00,00,00,01,000,100,0,0,0,0,0,0,1,1011,0,0,0000,0,11,000,0,110,111,001)

11003
11004
11005      !*** LOOP-BACK ENTRY POINT ***
11006
11007      5671:
11008      TESTD500:
11009      P0,      LOAD=ENVA(NEXTPAT500), !COMPARE POINT IN LOOP
11010      LOAD=ERROR(TESTD500),         !ERROR DIRECTORY KEY
11011      DCS=CTR(C4,),                 !COMPARE AT TARGET
11012      P2=U,    IR_DBUF,             !JUST HAPPENS, DON'T CARE NOW
11013      P3,      DBUF_D,             !PUT PATTERN LEFT IN D INTO DBUF
11014      NEXT,     J/LOADIP500         !
(6671) DCS(1,00,1,0,0,0) BM(1011,00,11,00,11,11,11,0,0,0,0,0,0,1,1010,0,0,0000,0,11,000,0,001,011,010)

11015
11016      6132: 1(FREE)
11017      LOADIP500:
11018      P2=U,    IR_DBUF,             !PUT PATTERN NOW IN DBUF TO IR
11019      P3,      DBUF_D,             !JUST HAPPENS, DON'T CARE NOW
11020      NEXT,     J/TESTINHSP500      !
(6132) DCS(0,00,0,0,0,0) BM(0000,00,00,00,00,000,000,0,0,0,0,0,0,1,1010,0,0,0000,0,11,000,0,001,011,011)

11021
11022      6133: 1(FREE)
11023      TESTINHSP500:
11024      P2=T,    D_ASPL0[OVERLAP]=PLUS-1, !GET BUMPED OVERLAP COUNT, B-SIDE
11025      DICI_0, !
11026      P3,      BSPLO[OVERLAP]_D,     !ONLY WRITTEN BACK IF
11027      BUTA(INSTR=1),                 !"OVERLAP L" ASSERTED
11028      NEXT,     J/TESTINHSP500      !"BRANCH EFFECT OF BUT MASKED)
(6133) DCS(0,00,0,0,0,0) BM(1001,00,10,11,01,000,000,0,0,1,0,0,0,0,0000,0,0,0110,0,00,110,0,111,111,111)

11029
11030      6777:
11031      TESTINHSP500:
11032      P2=T,    D_ASPL0[OVERLAP]=PLUS-1, !GET BUMPED OVERLAP COUNT, A-SIDE
11033      SAVE-DIC], !
11034      P3,      ASPLO[OVERLAP]_D,     !ONLY WRITTEN BACK IF
11035      BUTA(INSTR=1),                 !"OVERLAP L" ASSERTED
11036      NEXT,     J/NEXTPAT500        !"BRANCH EFFECT OF BUT MASKED)
(6777) DCS(0,00,0,0,0,0) BM(1001,00,11,10,00,000,111,0,0,1,0,0,0,0,0000,0,0,0001,0,00,110,0,011,111,111)

11037
11038      6777:
11039      NEXTPAT500:
11040      P2=T,    D_ASPL0[PATTERN]=PLUS-020010-PLUS-1, !GET NEXT DATA PATTERN IN D

```

```

11041      DIC]_COUNT,      [GET DIC]OVERFLOW OUT OF BIT<15>
11042      P3,      A#BSPHI[PATTERN]_D,      [WRITE BACK UNMASKED NEXT PATTERN
11043      NEXT,      BUTR(PREFETCH-L),      [IF ASSERTED, "ASSERTFOVS00"
11044      J/ASSERTFOVS00      [OTHERWISE, "FIXPAT500"
(6377) DCS[0.00,0.0,0,0] BM[1100..11.00..11.01..011..110...0.1,0,0,0...0,0000...0..1011,0...10,000...101,010,101]
11045
11046      ENTER HERE IF "PREFETCH-L" WAS ASSERTED, LOW
11047      6525:
11048      ASSERTFOVS00:
11049      P2-T,      D_ASPHI[PREFETCH]-PLUS-1,      [PREFETCH(0)H" ASSERTED, BUMP COUNT
11050      SAVE-DIC],      [SAVE CARRYOUT]
11051      P3,      ASPHI[PREFETCH]_D,
11052      NEXT,      J/FIXPAT500
(6525) DCS[0.00,0.0,0,0] BM[1001..01.11..11.00..000...111...0.1,0,0,0...0,0000...0..1001,0...11,000...101,010,111]
11053
11054      ENTER HERE IF "PREFETCH-L" WAS NOT ASSERTED, HIGH
11055      6527:
11056      FIXPAT500:
11057      P2-T,      D_ASPHI[PATTERN]-AND-NOT-020010,      [CLEAR UNNEEDED BITS
11058      SAVE-DIC],      [KEEP DIC] CONSTANT FOR "BUT"
11059      P3,      A#BSPHI[PATTERN]_D,      [RESTORE PATTERN, ALSO LEAVE IN D
11060      NEXT,      BUTR(DIC)-B),      [TEST CARRYOUT]
11061      J/TESTD500      [IF CLEAR, NEXT PATTERN AT: "TESTD500" (-6, WORDS)
(6527) DCS[0.00,0.0,0,0] BM[0111..11.00..11.01..011..111...0.1,0,0,0...0,0000...0..1011,0...10,011...110,111,001]
11062      [IF SET, EXIT LOOP TO: "TEST500A" (+1, WORD)
11063
11064
11065      | - - - - -
11066
11067      *** TEST 500A ***
11068      [CHECK THAT REGISTER OPPOSITE ASPHI[PREFETCH], IN BSPHI, STILL ZERO
11069      6673:
11070      TEST500A:
11071      P0,      LOAD-ENUA(ZTARGET434),      [SETUP FOR IN(000000)/INSTR5 TEST
11072      LOAD-ERROR(TEST500A),      [ERROR DIRECTORY KEY
11073      DCS-CTR(C7),      [COMPARE AT TARGET
11074      BUMP-VERIFY,      [COUNT
11075      NEXT,      J/COMP500A
(6673) DCS[1.00,1.0,0,1] BM[1000..00.11..11.00..011..100...0.0,0,0,0...0,0000...0..0000,0...11,000...001,011,100]
11076
11077      6134: [FREE]
11078      COMP500A:
11079      P2-T,      D_BSPHI[PREFETCH],      [GET REGISTER
11080      SAVE-DIC],
11081      NEXT,      J/GOBUT500A
(6134) DCS[0.00,0.0,0,0] BM[1010..01.10..00.00..000...111...0.1,0,0,0...0,0000...0..0000,0...11,000...001,011,101]
11082
11083      6135: [FREE]
11084      GOBUT500A:
11085      P0,      BUMP-VERIFY,      [COUNT
11086      SETUP,      RETURN/TEST500C,      [RETURN TO START OF NEXT SUBTEST
11087      NEXT,      CALL(DINTOR-5)      [PUT D -> DBUF -> IR, DO INSTR5 BUT FOR (000000)
(6135) DCS[0.00,0.0,0,1] BM[0110..00.11..10.10..110..111...0.0,0,0,0...0,0000...0..0000,0...11,100...010,111,011]

```

```

11088      |
11089      *** ALSO SETS UP BUSDIN <- EMIT UPON EXIT ***
11090
11091
11092
11093
11094      | - - - - -
11095
11096      *** TEST 500C ***
11097      [CHECK THAT ASSERTING "OVERLAP L" ONLY ALLOWED A SP REWRITE (DURING
11098      1A BUT(INSTR-1)) IN ASPLO 4005, TIMES / 16384, DATA PATTERNS
11099      6776:
11100      TEST500C:
11101      P0,      LOAD-ENUA(ZTARGET402),      [SETUP FOR D=0 TEST
11102      LOAD-ERROR(TEST500C),      [ERROR DIRECTORY KEY
11103      DCS-CTR(C5),      [COMPARE AT TARGET
11104      BUMP-VERIFY,      [COUNT
11105      NEXT,      J/EXPEC500C
(6776) DCS[1.00,1.0,0,1] BM[1010..00.11..11.00..000..010...0.0,0,0,0...0,0000...0..0000,0...11,000...001,011,110]
11106
11107      6136: [FREE]
11108      EXPEC500C:
11109      P3,      CSPD[EXPEC]_EMIT,      [EXPECTED COUNT =
11110      EMIT/7645,      [ (7645) = 4005.
11111      NEXT,      J/COMP500C
(6136) DCS[0.00,0.0,0,0] BM[0000..10.11..11.10..100..101...0.0,0,0,0...0,0000...0..0000,0...11,000...001,011,111]
11112
11113      6137: [FREE]
11114      COMP500C:
11115      P0,      BUMP-VERIFY,      [COUNT
11116      P3-T,      D_ASPLO[OVERLAP]-MINUS-CSPB[EXPEC],      [COMPARE RECEIVED:EXPECTED
11117      SAVE-DIC],
11118      NEXT,      J/GOBUT500C
(6137) DCS[0.00,0.0,0,1] BM[1101..11.00..10.00..000..111...1.1,0,0,0...0,0000...0..0000,0...11,000...001,100,000]
11119
11120      6140: [FREE]
11121      GOBUT500C:
11122      SETUP,      RETURN/TEST500D,      [RETURN TO START OF NEXT SUBTEST
11123      NEXT,      GOTO-PAGE(7),      [BUT TABLE
11124      J/BUTD-IS-ZERO      [CHECK EQUALITY
(6140) DCS[0.00,0.0,0,0] BM[0110..00.11..11.00..110..111...0.0,0,0,0...0,0000...0..0000,0...11,100...011,100,001]
11125
11126
11127
11128
11129
11130      | - - - - -
11131
11132      *** TEST 500D ***
11133
11134      [CHECK THAT ASSERTING "OVERLAP L" ALLOWED THE SAME NUMBER OF SP REWRITES
11135      1TO ASPLO (CHECKED ABOVE) AS TO BSPLO (CHECKED HERE)
11136      6746:
11137      TEST500D:

```

```
11137      PO,      LOAD=ENUA(ZTARGET402),      |SETUP FOR D=0 TEST
11138      LOAD=ERROR(TEST500D),      |ERROR DIRECTORY KEY
11139      DCS=CTR(C4.),      |COMPARE AT TARGET
11140      NEXT,      J/COMP500D      |
(6746) DCS[1.00,1.0,0,0] BM[1011..00,11..11,00..000..010...0,0,0..0..0..0.0000...0..0000,0...11,000...001,100,001]
11141
11142      6141: 1(FREE)
11143      COMP500D:
11144      P3=T,      D=ASPLO[OVERLAP]-MINUS-BSPLO[OVERLAP],      |COMPARE A-COUNT;B-COUNT
11145      SAVE=D[C],      |
11146      NEXT,      J/GOBUT500D      |
(6141) DCS[0.00,0.0,0,0] BM[1101..00,10..10,00..000..111...1,1,0..0..0..0.0000...0..0000,0...11,000...001,100,010]
11147
11148      6142: 1(FREE)
11149      GOBUT500D:
11150      PO,      BUMP-VERIFY,      |COUNT
11151      SETUP,      RETURN/TEST500E,      |RETURN TO START OF NEXT SUBTEST
11152      NEXT,      GOTO=PAGE(7),      |BUT TABLE
11153      J/BUTD-IS-ZERO      |CHECK EQUALITY
(6142) DCS[0.00,0.0,0,0] BM[0110..00,11..10,11..110..111...0,0,0..0..0..0.0000...0..0000,0...11,100...011,100,001]
11154
11155
11156
11157
11158
11159      | - - - - -
11160
11161      |*** TEST 500E ***
11162      |CHECK THAT "PREFETCH(0)H" WAS ONLY ASSERTED 357, TIMES/16384, DATA PATTERNS
11163      4736:
11164      TEST500E:
11165      PO,      LOAD=ENUA(ZTARGET402),      |SETUP FOR D=0 TEST
11166      LOAD=ERROR(TEST500E),      |ERROR DIRECTORY KEY
11167      DCS=CTR(C5.),      |COMPARE AT TARGET
11168      NEXT,      J/EXPEC500E      |
(6736) DCS[1.00,1.0,0,0] BM[1010..00,11..11,00..000..010...0,0,0..0..0..0.0000...0..0000,0...11,000...001,100,011]
11169
11170      6143: 1(FREE)
11171      EXPEC500E:
11172      PO,      BUMP-VERIFY,      |
11173      P3,      CSPD[EXPEC]_EMIT,      |EXPECTED COUNT=
11174      EMIT/545,      | (545)MS57.
11175      NEXT,      J/COMP500E      |
(6143) DCS[0.00,0.0,0,0] BM[0000..10,00..01,01..100..101...0,0,0..0..0..0.0000...1..0000,0...11,000...001,100,100]
11176
11177      6144: 1(FREE)
11178      COMP500E:
11179      P3=T,      D=ASPH[PREFETCH]-MINUS-CSPB[EXPEC],      |COMPARE RECEIVED;EXPECTED
11180      SAVE=D[C],      |
11181      NEXT,      J/GOBUT500E      |
(6144) DCS[0.00,0.0,0,0] BM[1101..11,00..11,00..000..111...1,1,0..0..0..0.0000...0..0000,0...11,000...001,100,101]
11182
11183      6145: 1(FREE)
```

```
11184      GOBUT500E:
11185      SETUP,      RETURN/SCOPE500F,      |RETURN TO SCOPE LOOP TEST WORD
11186      NEXT,      GOTO=PAGE(7),      |BUT TABLE
11187      J/BUTD-IS-ZERO      |CHECK EQUALITY
(6145) DCS[0.00,0.0,0,0] BM[0110..00,00..11,00..110..111...0,0,0..0..0..0.0000...0..0000,0...11,100...011,100,001]
11188
11189
11190
11191      6146: 1(FREE)
11192      SCOPE500F:
11193      PO,      BUMP-VERIFY,      |COUNT
11194      BUSDIN_EMIT=[1],      |RESET PROC UCON
11195      EN=CLK-IR[15=00],      |
11196      NEXT,      RUTD[SCOPE],      |NO ERROR: "TEST503A" (+1, WORDS)
11197      J/TEST503A      |ERROR: "GETFETCH500" (-27, WORDS)
(6146) DCS[0.00,0.1,0,1] BM[0000..00,00..00,01..000..100...0,0,0..0..0..1.1001...0..0000,0...11,000...110,111,111]
11198
11199
11200
11201
11202
11203      |.PAGE=====
11204
11205
11206      .TNC * TEST503-510: PROCESSOR UCON TESTS (FLAGS, FPS, PS, BUTM) & ASSOC LOGIC
11207
11208      |*****
11209      |*
11210      |* TESTS: 503A - 510F      UWORDS: 214 + 245
11211      |*
11212      |* FUNCTIONS:      TESTS 503A - 510F MANIPULATE THE VARIOUS PROCESSOR UCON
11213      |*      FUNCTIONS (FLAGS, EXFLAGS, FPS, PS, CUA) AND RELATED "BUT"
11214      |*      TESTS TO SEE THAT ALL ARE FUNCTIONAL.
11215      |*
11216      |*****
11217
11218
11219
11220
11221
11222
11223
11224      | - - - - -
11225
11226      |*** TEST 503 ***
11227
11228      |TESTS 503 A-I USE DATA PATTERNS OF:
11229      |      FLAG<R14,210>H = "10101010", EXFLAG<211>H = "01",
11230      |      FPS<710>H = "1010 1010"
11231
11232      | - - - - -
11233
11234      |*** TEST 503A ***
```



```
11329 (6156) DCS[0.00.0.0.0.0] BM[0110..00.11..00.11..010..111..0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.010.101]
11330
11331
11332
11333 1 - - - - -
11334
11335 1*** TEST 503C ***
11336 1DO THE "BUT" ON "FLTPT-PROC-H"="FLAGS-H=EXFLAG1=L" TO CHECK IT'S CLEAR
11337 6632:
11338 TEST503C:
11339     PO,          LOAD-ENUA(ZTARGET402),          |BIT<00> CLEAR
11340     LOAD-ERROR(TEST503C),          |ERROR DIRECTORY KEY
11341     DCS-CTR(C3.),          |COMPARE AT TARGET
11342     BUMP-VERIFY,          |COUNT
11343     NEXT,          J/GOBUT503C
11344 (6632) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..000..010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.101.111]
11345 6157: 1(FREE)
11346 GORUT503C:
11347     SETUP,          RETURN/TEST503DA,          |RETURN TO START OF NEXT SUBTEST
11348     NEXT,          GOTO-PAGE(7),          |BUT TABLE
11349     J/BUTFPFPROC          |FLAGS-H=EXFLAG1=L IN BIT<00>
11350 (6157) DCS[0.00.0.0.0.0] BM[0110..00.11..00.00..001..111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.101.100]
11351
11352
11353
11354 1 - - - - -
11355
11356 1*** TEST 503DA ***
11357 1DO THE "MULTIPLE BUT" ON "FLTPT" TO CHECK IT'S CLEAR, IR=(125122), NOT-FLTPT-H INSTR
11358 6601:
11359 TEST503DA:
11360     PO,          LOAD-ENUA(ZTARGET406),          |BIT<00> CLEAR
11361     LOAD-ERROR(TEST503DA),          |ERROR DIRECTORY KEY
11362     DCS-CTR(C4.),          |COMPARE AT TARGET
11363     NEXT,          J/GOBUT503DA
11364 (6601) DCS[1.00.1.0.0.0] BM[1011..00.11..11.00..000..110...0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.110.000]
11365 6160: 1(FREE)
11366 GORUT503DA:
11367     SETUP,          RETURN/TEST503D,          |RETURN TO START OF NEXT SUBTEST
11368     NEXT,          GOTO-PAGE(7),          |BUT TABLE
11369     J/BUTMFLTPTS          |FLTPT-H IN BIT<00>
11370 (6160) DCS[0.00.0.0.0.0] BM[0110..00.11..01.11..000..111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.011.001]
11371
11372
11373
11374 1 - - - - -
```

```
11375
11376 1*** TEST 503D ***
11377 1DO AN INSTR-1 FLOATING POINT DECODE, TO CHECK THAT FLAG<4:5> -> BIT<1:0>
11378 6670:
11379 TEST503D:
11380     PO,          LOAD-ENUA(ZTARGET476),          |INSTR-1 FLTPT, BIT<1:0>="10"
11381     LOAD-ERROR(TEST503D),          |ERROR DIRECTORY KEY
11382     DCS-CTR(C4.),          |COMPARE AT TARGET
11383     NEXT,          J/LOADIR503D
11384 (6670) DCS[1.00.1.0.0.0] BM[1011..00.11..11.00..111..110...0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.110.001]
11385 6161: 1(FREE)
11386 LOADIR503D:
11387     PO,          BUMP-VERIFY,          |COUNT
11388     P2-U,          IR-EMIT,          |IR <- FLTPT INSTR
11389     EMIT/175252,          |
11390     NEXT,          J/GOBUT503D
11391 (6161) DCS[0.00.0.0.0.0] BM[1111..00.10..10.10..101..010...0.0.0.0.0.0.0.1010...0.0000.0...11.000...001.110.010]
11392 6162: 1(FREE)
11393 GORUT503D:
11394     SETUP,          RETURN/TEST503E,          |RETURN TO START OF NEXT SUBTEST
11395     NEXT,          GOTO-PAGE(7),          |BUT TABLE
11396     J/BUTINSTR1          |FULL WIDTH
11397 (6162) DCS[0.00.0.0.0.0] BM[0110..00.11..10.00..000..111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.000.110]
11398
11399
11400
11401 1 - - - - -
11402
11403 1*** TEST 503F ***
11404 1READ EXFLAG<2:1> = "01" IN CUA=PORT<2:1>,
11405 1A1.80 "PREFETCH-SAVE-H" = "0" FROM PREVIOUS SETUP
11406 6700:
11407 TEST503E:
11408     PO,          LOAD-ENUA(ZTARGET402),          |SETUP FOR D=0 TEST
11409     LOAD-ERROR(TEST503E),          |ERROR DIRECTORY KEY
11410     DCS-CTR(C10.),          |COMPARE AT TARGET
11411     NEXT,          J/EXPEC503E
11412 (6700) DCS[1.00.1.0.0.0] BM[0101..00.11..11.00..000..010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...001.110.011]
11413 6163: 1(FREE)
11414 EXPEC503E:
11415     PO,          BUMP-VERIFY,          |COUNT
11416     P3,          CSFD[02]_EMIT,          |CUA PORT READS AS:
11417     EMIT/073732,          |10CUA<11:00>=EXFLAG<2:1>=PREFETCH
11418     NEXT,          J/GOBUT503E
11419 (6163) DCS[0.00.0.0.0.0] BM[0111..10.01..11.11..011..010...0.0.0.0.0.0.0.1101...1.0000.0...11.000...001.110.100]
11420 6164: 1(FREE)
11421 GORUT503E:
11422     SETUP,          RETURN/TEST503F,          |GO TO SUBR WHICH:
```

```
11423 NEXT, CALL(CUATOD) IPUTS CUAEXFLAG, XOR, C0P(02) -> D, BUT(D=ZERO)
(6164) DCS(0.00,0.0,0.0) BM(0110.00,11.10,01.00,0.0000.111.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,100.0.010,101.111)
11424
11425
11426
11427
11428 | - - - - -
11429
11430 |*** TEST 503F ***
11431 |DO THE "MULTIPLE BUT" ON EXFLAG<2> TO CHECK IT'S CLEAR
11432 6710:
11433 TEST503F:
11434 PO, LOAD-ENUA(ZTARGET406), IBIT<00> CLEAR
11435 LOAD-ERROR(TEST503F), IERROR DIRECTORY KEY
11436 DCS-CTR(C4.), ICOMPARE AT TARGET
11437 NEXT, J/GOBUT503F
(6710) DCS(1.00,1.0,0.0) BM(1011.00,11.11,00.000.110.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,000.0.001,110,101)
11438
11439 6165: I(FREE)
11440 GOBUT503F:
11441 SETUP, RETURN/TEST503G, IRETURN TO START OF NEXT SUBTEST
11442 NFXT, GOTO-PAGE(7), IBUT TABLE
11443 J/BUTEXFLAG2 IEXFLAG<2>H IN BIT<00>
(6165) DCS(0.00,0.0,0.0) BM(0110.00,11.10,10.000.111.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,100.0.011,011,011)
11444
11445
11446
11447
11448 | - - - - -
11449
11450 |*** TEST 503G ***
11451 |DO THE "MULTIPLE BUT" ON EXFLAG<1> TO CHECK IT'S SET
11452 6720:
11453 TEST503G:
11454 PO, LOAD-ENUA(ZTARGET407), IBIT<00> SET
11455 LOAD-ERROR(TEST503G), IERROR DIRECTORY KEY
11456 DCS-CTR(C4.), ICOMPARE AT TARGET
11457 BUMP-VERIFY, ICOUNT
11458 NEXT, J/GOBUT503G
(6720) DCS(1.00,1.0,0.1) BM(1011.00,11.11,00.000.111.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,000.0.001,110,110)
11459
11460 6166: I(FREE)
11461 GOBUT503G:
11462 SETUP, RETURN/TEST503H, IRETURN TO START OF NEXT SUBTEST
11463 NFXT, GOTO-PAGE(7), IBUT TABLE
11464 J/BUTEXFLAG1 IEXFLAG<1>H IN BIT<00>
(6166) DCS(0.00,0.0,0.0) BM(0110.00,11.10,11.010.111.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,100.0.011,010,111)
11465
11466
11467
11468
```

```
11469 | - - - - -
11470
11471 |*** TEST 503H ***
11472 |CHECK FPS<5> SET, VIA BUTR
11473 6732:
11474 TEST503H:
11475 PO, LOAD-ENUA(ZTARGET407), IBIT<00> SET
11476 LOAD-ERROR(TEST503H), IERROR DIRECTORY KEY
11477 DCS-CTR(C3.), ICOMPARE AT TARGET
11478 BUMP-VERIFY, ICOUNT
11479 NEXT, J/GOBUT503H
(6732) DCS(1.00,1.0,0.1) BM(1100.00,11.11,00.000.111.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,000.0.001,110,111)
11480
11481 6167: I(FREE)
11482 GOBUT503H:
11483 SETUP, RETURN/TEST503I, IRETURN TO START OF NEXT SUBTEST
11484 NEXT, GOTO-PAGE(7), IBUT TABLE
11485 J/BUTFPS05 IFPS<05> IN BIT<00>
(6167) DCS(0.00,0.0,0.0) BM(0110.00,11.11,00.010.111.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,100.0.011,001,010)
11486
11487
11488
11489
11490 | - - - - -
11491
11492 |*** TEST 503I ***
11493 |CHECK FLTPT-PD-H = F(FPS<7>16, FLAG<5,2,1,EX1>) IS CLEAR
11494 6742:
11495 TEST503I:
11496 PO, LOAD-ENUA(ZTARGET402), IBIT<00> CLEAR
11497 LOAD-ERROR(TEST503I), IERROR DIRECTORY KEY
11498 DCS-CTR(C3.), ICOMPARE AT TARGET
11499 NEXT, J/GOBUT503I
(6742) DCS(1.00,1.0,0.0) BM(1100.00,11.11,00.000.010.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,000.0.001,111,000)
11500
11501 6170: I(FREE)
11502 GOBUT503I:
11503 PO, BUMP-VERIFY, ICOUNT
11504 SETUP, RETURN/TEST503J, IRETURN TO START OF NEXT SUBTEST
11505 NFXT, GOTO-PAGE(7), IBUT TABLE
11506 J/BUTFPFD IFLTPT-PD-H IN BIT<00>
(6170) DCS(0.00,0.0,0.1) BM(0110.00,11.11,01.110.111.0.0,0.0.0.0.0.0000.0.0.0000.0.0.11,100.0.011,111,001)
11507
11508 | - - - - -
11509
11510 |*** TEST 503J ***
11511 |DO THE "MULTIPLE BUT" ON "D<00>" TO CHECK IT'S CLEAR (D WAS ZEROED IN TEST503E, ABOVE)
11512 6756:
11513 TEST503J:
11514 PO, LOAD-ENUA(ZTARGET406), IBIT<00> CLEAR
11515 LOAD-ERROR(TEST503J), IERROR DIRECTORY KEY
11516 DCS-CTR(C4.), ICOMPARE AT TARGET
```

```
11517 NEXT, J/GOBUTS03J
(6756) DCS[1.00,1.0,0,0] BM[1011.00,11.11,00.000.110...0,0,0,0...0.0000...0.0000,0...11,000...001,111,001]
11518
11519 6171: 1(FREE)
11520 GOBUTS03J:
11521 SETUP, RETURN/TESTS03K, RETURN TO START OF NEXT SUBTEST
11522 NEXT, GOTO-PAGE(7), IBUT TABLE
11523 J/BUTMD00 ID<00>=H IN BIT<00>
(6171) DCS[0.00,0,0,0,0] BM[0110.00,11.01,00.110.111...0,0,0,0...0.0000...0.0000,0...11,100...011,010,001]
11524
11525 1 - - - - -
11526
11527
11528 *** TEST 503K ***
11529 ADD THE "MULTIPLE BUT" ON "FLTP" TO CHECK IT'S SET, IR=(175282), FLTP=H INSTR
11530 6646:
11531 TESTS03K:
11532 PO, LOAD-ENUA(ZTARGET407), IBIT<00> SET
11533 LOAD-ERROR(TESTS03K), ERROR DIRECTORY KEY
11534 DCS-CTR(C4.), ICOMPARE AT TARGET
11535 BUMP-VERIFY, COUNT
11536 NEXT, J/GOBUTS03K
(6646) DCS[1.00,1.0,0,1] BM[1011.00,11.11,00.000.111...0,0,0,0...0.0000...0.0000,0...11,000...001,111,010]
11537
11538 6172: 1(FREE)
11539 GOBUTS03K:
11540 SETUP, RETURN/SCOPES03, RETURN TO SCOPE LOOP TEST WORD
11541 NEXT, GOTO-PAGE(7), IBUT TABLE
11542 J/BUTMFLTPS IFLTP=H IN BIT<00>
(6172) DCS[0.00,0,0,0,0] BM[0110.00,00.11,11.011.111...0,0,0,0...0.0000...0.0000,0...11,100...011,011,001]
11543
11544
11545
11546
11547
11548
11549 6173: 1(FREE)
11550 SCOPES03:
11551 PO, BUSDIN_EMIT-[1], RESET PROC UCONS
11552 EN=CLK-IR[15-00], NO ERRDR; "TESTS04A" (+1, WORDS)
11553 NEXT, BUTD[SCOPE], ERROR; "LOADFLAGS04A" (+34, WORDS)
11554 J/TESTS04A
(6173) DCS[0.00,0,1,0,0] BM[0000.00,00.00,01.000.100...0,0,0,0...0.0000...0.0000,0...11,000...110,001,101]
11555
11556
11557
11558
11559
11560
11561
11562
11563
```

```
11564 1 - - - - -
11565
11566 *** TEST 504 ***
11567
11568 TESTS 504 A-I USE DATA PATTERNS OF:
11569 1 FLAG<8:4,2:0>H = "01010101", EXFLAG<2:1>H = "10",
11570 1 FPS<7:0>H = "0101 0101"
11571
11572 1 - - - - -
11573
11574 *** TEST 504A ***
11575 LOAD FLAGS<8:4,2:0> WITH "01010101", EXFLAG<2:1> WITH "10", FPS<7:4> WITH "0101",
11576 IFPS<3:0> WITH "0101", AND READ BACK THRU "FLAGSPPS" PORT OF PROCESSOR MUX
11577 6615:
11578 TESTS04A:
11579 PO, LOAD-ENUA(4777), SETUP FOR COMPARE 1/2 WAY + 1 THRU BM SUBR
11580 LOAD-ERROR(TESTS04A), ERROR DIRECTORY KEY
11581 DCS-CTR(C11.), ICOMPARE AT ...
11582 NEXT, J/LOADFLAGS04A
(6615) DCS[1.00,1.0,0,0] BM[0100.00,10.01,11.111.111...0,0,0,0...0.0000...0.0000,0...11,000...110,010,010]
11583
11584 6622:
11585 LOADFLAGS04A:
11586 P2=U, IP_EMIT, ALSO NOTE IR=(NOT-PREFETCH), SO PREFETCH=SAVE WILL GET "0"
11587 P3, CSPD[05]_EMIT, AFTER SUBSEQUENT JAMUPP (IN LOADPPSC ROUTINE EXIT)
11588 EMIT/052644, INITIAL VALUE:
11589 NEXT, J/LOADPPS04A IFLAG<8:4,2:0>="01010101", EXFLAG<2:1>="10"
11590 (6622) DCS[0.00,0,0,0,0] BM[0101.10,01.01,10.100.100...0,0,0,0...0.0000...0.0000,0...11,000...001,111,100]
11591
11592 6174: 1(FREE)
11593 LOADPPS04A:
11594 P3, CSPD[06]_EMIT, INITIAL VALUE IN FPS<7:4>:
11595 EMIT/125133, "10101010 0101 1011"
11596 NEXT, J/EXPECS04A
(6174) DCS[0.00,0,0,0,0] BM[1010.10,10.10,01.011.011...0,0,0,0...0.0000...0.0000,0...11,000...001,111,101]
11597
11598 6175: 1(FREE)
11599 EXPECS04A:
11600 P3, CSPD[02]_EMIT, EXPECTED VALUE LOADS FLAGS, FPS<7:0>
11601 EMIT/052525, OF "FLAGSPPS" PORT
11602 NEXT, J/LOADPPCS04A "01010101 0101 0101"
(6175) DCS[0.00,0,0,0,0] BM[0101.10,01.01,01.010.101...0,0,0,0...0.0000...0.0000,0...11,000...001,111,110]
11603
11604 6176: 1(FREE)
11605 LOADPPCS04A:
11606 P3, CSPD[15]_EMIT, FPS<3:0> COME FROM CSP[15]<3:0>(WD1
11607 EMIT/000005, "0101"
11608 NEXT, J/DNPPCS04A
(6176) DCS[0.00,0,0,0,0] BM[0000.10,00.00,00.000.101...0,0,0,0...0.0010...0.0000,0...11,000...001,111,111]
11609
11610 6177: 1(FREE)
11611 DNPPCS04A:
11612 P3, CSPD[00]_EMIT, ICALL BM SUBR WHICH DOES THE LOAD
```

```
11613 RETURN/TESTS04A, RETURN INLINE
11614 NEXT, CALL(LOADPPSCC)
(6177) DCS(0.00,0.0,0.0) BM(0110..10.11..11.10..110..111...0.0.0.0.0.0.0.111...0.0000.0...11.100...010.110.100)
11615 | - - - - -
11616 |
11617 |NOW CHECK ALL THE RITE BITS WERE SET BY READING THEM BACK
11618 |
11619 |
11620 6766:
11621 TESTS04A:
11622 PO, LOAD-ENUA(ZTARGET402), |SETUP FOR DUBERG TEST
11623 LOAD-ERROR(TESTS04A), |ERROR DIRECTORY KEY
11624 DCS-CTR(C14), |COMPARE AT TARGET
11625 BUMP-VERIFY, |COUNT
11626 BUTA(CUA-TRACK), |RESET CUA TRACKING AFTER JAMUPP
11627 NEXT, J/GOBUTS04A
(6766) DCS(1.00,1.0,0.1) BM(0001..00.11..11.00..000..010...0.0.0.0.0.0.0.0000...0..0000.0...11.001...010.000.000)
11628 6200: I(FREE)
11629 GOBUTS04A:
11630 SETUP, RETURN/GOBUTS04A, |GO TO SUBR WHICH:
11631 NEXT, CALL(FLAGPPSEQL0D) |1) CSPT05) -> FLAG5, EXFLAG5
(6200) DCS(0.00,0.0,0.0) BM(0110..00.01..00.00..001..111...0.0.0.0.0.0.0.0000...0..0000.0...11.100...000.111.011)
11633 |2) CSPT06) -> FFS<7,6>
11634 |
11635 6201: I(FREE)
11636 GOBUTS04A:
11637 SETUP, RETURN/TESTS04B, |RETURN TO START OF NEXT SUBTEST
11638 NEXT, CALL(FLAGPPSAT0D) | FLAG5<FFS.XOF,CSPT02) -> D, BUT(D=0)
(6201) DCS(0.00,0.0,0.0) BM(0110..00.11..10.11..000..111...0.0.0.0.0.0.0.0000...0..0000.0...11.100...010.101.010)
11639 |
11640 |
11641 |
11642 | - - - - -
11643 |
11644 |*** TEST 504B ***
11645 |DO THE "MULTIPLE BUT" ON "FLAG7-H" TO CHECK IT'S SET
11646 6730:
11647 TESTS04B:
11648 PO, LOAD-ENUA(ZTARGET407), |BIT<00> SET
11649 LOAD-ERROR(TESTS04B), |ERROR DIRECTORY KEY
11650 DCS-CTR(C4), |COMPARE AT TARGET
11651 NEXT, J/GOBUTS04B
(6730) DCS(1.00,1.0,0.0) BM(1011..00.11..11.00..000..111...0.0.0.0.0.0.0.0000...0..0000.0...11.000...010.000.010)
11652 6202: I(FREE)
11653 GOBUTS04B:
11654 SETUP, RETURN/TESTS04C, |RETURN TO START OF NEXT SUBTEST
11655 NEXT, GOTO-PAGE(7), |BUT TABLE
11656 J/BUTFLAG7 |FLAG7-H IN BIT<00>
11657
```

```
(6202) DCS(0.00,0.0,0.0) BM(0110..00.11..11.00..000..111...0.0.0.0.0.0.0.0000...0..0000.0...11.100...011.010.101)
11658 |
11659 |
11660 |
11661 |
11662 | - - - - -
11663 |
11664 |*** TEST 504C ***
11665 |DO THE "BUT" ON "FLTP-PROC-H"="FLAG<5>-H=EXFLAG<1>=L" TO CHECK IT'S SET
11666 6740:
11667 TESTS04C:
11668 PO, LOAD-ENUA(ZTARGET403), |BIT<00> SET
11669 LOAD-ERROR(TESTS04C), |ERROR DIRECTORY KEY
11670 DCS-CTR(C3), |COMPARE AT TARGET
11671 BUMP-VERIFY, |COUNT
11672 NEXT, J/GOBUTS04C
(6740) DCS(1.00,1.0,0.1) BM(1100..00.11..11.00..000..011...0.0.0.0.0.0.0.0000...0..0000.0...11.000...010.000.011)
11673 6203: I(FREE)
11674 GOBUTS04C:
11675 SETUP, RETURN/TESTS04D, |RETURN TO START OF NEXT SUBTEST
11676 NEXT, GOTO-PAGE(7), |BUT TABLE
11677 J/BUTPPROC |FLAG5-H=EXFLAG1-L IN BIT<00>
(6203) DCS(0.00,0.0,0.0) BM(0110..00.11..11.01..000..111...0.0.0.0.0.0.0.0000...0..0000.0...11.100...011.101.100)
11679 |
11680 |
11681 |
11682 |
11683 | - - - - -
11684 |
11685 |*** TEST 504D ***
11686 |DO AN INSTR-1 FLOATING POINT DECODE, TO CHECK THAT FLAG<4:5> -> BIT<1:0>
11687 6750:
11688 TESTS04D:
11689 PO, LOAD-ENUA(ZTARGET475), |INSTR=1 FLTP, BIT<1:0>="01"
11690 LOAD-ERROR(TESTS04D), |ERROR DIRECTORY KEY
11691 DCS-CTR(C4), |COMPARE AT TARGET
11692 NEXT, J/LOADIRS04D
(6750) DCS(1.00,1.0,0.0) BM(1011..00.11..11.00..111..101...0.0.0.0.0.0.0.0000...0..0000.0...11.000...010.000.100)
11693 6204: I(FREE)
11694 LOADIRS04D:
11695 PO, BUMP-VERIFY, |COUNT
11696 P2=U, IR_EMIT, |IR <- FLTP INSTR
11697 EMIT/172525, |
11698 NEXT, J/GOBUTS04D |
11699 |
(6204) DCS(0.00,0.0,0.1) BM(1111..00.01..01.01..010..101...0.0.0.0.0.0.0.1.1010...0..0000.0...11.000...010.000.101)
11700 6205: I(FREE)
11701 GOBUTS04D:
11702 SETUP, RETURN/TESTS04E, |RETURN TO START OF NEXT SUBTEST
11703 NEXT, GOTO-PAGE(7), |BUT TABLE
11704 J/BUTINSTR |FULL WIDTH
11705
```



```

11799 1
11800
11801 *** TEST 504I ***
11802 !CHECK FLTP-FD-H = F(FPS<716>,FLAG<5,2,1,EX1>) IS SET
11803 67161
11804 TEST504I:
11805 PO, LOAD=ENUA(ZTARGET403), !BIT<00> SET
11806 LOAD=ERROR(TEST504I), !ERROR DIRECTORY KEY
11807 DCS=CTR(C.), !COMPARE AT TARGET
11808 J/GOBUTS04I !
(6716) DCS[1.00,1.0.0.0] BM[1100..00.11..11.00..000..011...0.0.0.0..0...0.0000...0..0000,0...11.000...010.001.011]
11809
11810 6213: 1(FREE)
11811 GOBUTS04I:
11812 PO, BUMP=VERIFY, !COUNT
11813 SETUP, RETURN/SCOPE504, !RETURN TO SCOPE LOOP TEST WORD
11814 NEXT, GOTO-PAGE(7), !SUB TABLE
11815 J/BUTFPFD !FLTP-FD-H IN BIT<00>
(6213) DCS[0.00,0.0.0.1] BM[0110..00.01..00.01..100..111...0.0.0.0..0...0.0000...0..0000,0...11.100...011.111.001]
11816
11817
11818
11819 6214: 1(FREE)
11820 SCOPE504:
11821 PO, BUSDIN_EMIT-[I], !RESET PROC UCON
11822 EN=CLK-IR(15=00), !
11823 NEXT, BUTD[SCOPE], !NO ERROR; "TEST505A" (+1, WORDS)
11824 J/TEST505A ! ERROR; "LOADFLAG504A" (-27, WORDS)
(6214) DCS[0.00,0.1.0.0] BM[0000..00.00..00.01..000..100...0.0.0.0..0...1.1001...0..0000,0...11.000...110.010.011]
11825
11826
11827
11828
11829
11830
11831
11832
11833
11834
11835
11836
11837
11838 *** TEST 505 ***
11839 !TESTS 505 A-C USE DATA PATTERNS OF:
11840 ! FLAG<8;4,2;0>H = "11111000", EXFLAG<2;1>H = "10",
11841 ! FPS<710>H = "0101 0101"
11842
11843
11844 *** TEST 505A ***
11845 !CHECK THAT BUTA(CLEAR-FLAG-RES-UCON) ONLY CLEARS SHORT-TERM FLAGS
11846 6623:
11847 TEST505A:
11848 PO, LOAD=ENUA(ZTARGET402), !SERUP FOR D=ZERO TEST

```

```

11849          LOAD-ERROR(TEST505A),          IERROR DIRECTORY KEY
11850          DCS-CTR(C15, ),                  ICOMPARE AT TARGET
11851          J/LOAD505A                          I
(6623) DCS(1.00,1.0,0.0) BM(0000.00.11..11.00.000.010..0.0.0.0.0..0.0000...0..0000,0...11.000...110.010,000)

11852          66201
11853          I/AD505A:
11854          P3,                                CSPD(05) EMIT,          ILOAD FLAG<014,2:0>,
11855          EMIT/177406,                      IEXFLAG<2:1> WITH ALL 1-8
11856          NEXT, J/EXPEC505A                  I
(6620) DCS(0.00,0.0,0.0) BM(1111..10.11..11.00.000..110..0.0.0.0.0..0.0.1010...1..0000,0...11.000...010.001,101)

11858          6215: I(FREE)
11859          EXPEC505A:
11860          P0,                                BUMP-VERIFY,          ICOUNT
11861          P3,                                CSPD(02) EMIT,          IAFTR BUTA(CLR-FLAG-...), EXPECT THIS IN FLAG$FPS PORT:
11862          EMIT/174125,                      I"1111000 0101 0101"
11863          NEXT, J/SETFLAG505A                I
(6215) DCS(0.00,0.0,0.1) BM(1111..10.10.00.01..010..101..0.0.0.0.0..0.0.1101...1..0000,0...11.000...010.001,110)

11865          6216: I(FREE)
11866          SFTFLAG505A:
11867          P0,                                BUMP-VERIFY,          ICOUNT
11868          SETUP, RETURN/BUTCLR505A,          IGO TO SUBR WHICH:
11869          NEXT, CALL(FLAGLOD)                 IPUTS CSP(05) INTO THE FLAG$
(6216) DCS(0.00,0.0,0.1) BM(0110..00.01..00.01..111..111...0.0.0.0.0..0.0.0000...0..0000,0...11.100...000.111,110)

11871          6217: I(FREE)
11872          BUTCLR505A:
11873          P0,                                DCS-CTR(C9, ),          IPOINT COUNTER AT TARGET
11874          P3,                                BUTA(CLR-FLAG-RES-UCON), ICLEAR SHORT-TERM FLAG$
11875          NEXT, J/GOPUT505A                    I
(6217) DCS(0.00,1.0,0.0) BM(0110..00.00..00.00.000.000..0.0.0.0.0..0.0.0000...0..0000,0...11.010...010.010,000)

11877          6220: I(FREE)
11878          GOPUT505A:
11879          SETUP, RETURN/TEST505B,          IGO TO SUBR WHICH:
11880          NEXT, CALL(FLAG$PSIDD)              IFLAG$FPP$ XOR,CSP(2) -> D, BUT(D=ZERO)
(6220) DCS(0.00,0.0,0.0) BM(0110..00.11..10.01..010..111...0.0.0.0.0..0.0.0000...0..0000,0...11.100...010.101,010)

11882          I - - - - -
11883          I
11884          I
11885          I
11886          I - - - - -
11887          I
11888          I** TEST 505B ***

11889          ICHECK EXFLAG<2> NOT CLEARED, VIA BUTM
11890          6712:
11891          TEST505B:
11892          P0,                                LOAD-ENVA(ZTARGET407), IBIT<00> SET
11893          LOAD-ERROR(TEST505B),              IERROR DIRECTORY KEY
11894          DCS-CTR(C4, ),                      ICOMPARE AT TARGET

```

```
11895      BUMP=VERIFY,          ICOUNT
11896      NEXT, J/GOBUT505B      I
(6712) DCS[1,00,1,0,0,1] BM[1011,00,11,11,00,000,111,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,000,0,010,010,001]
11897
11898      6221: 1(FREE)
11899      GOBUT505B:
11900      SETUP, RETURN/TEST505C,      IRETURN TO START OF NEXT SUBTEST
11901      NEXT, GOTO=PAGE(7),          IBUT TABLE
11902      J/BUTMEXFLAG2      IEXFLAG<1> IN BIT<00>
(6221) DCS[0,00,0,0,0,0] BM[0110,00,11,10,10,010,111,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,100,0,011,011,011]
11903
11904
11905
11906
11907      I - - - - -
11908
11909      I*** TEST 505C ***
11910      ICHECK EXFLAG<1> WAS CLEARED, VIA BUTM
11911      6722:
11912      TEST505C:
11913      P0,      LOAD=ENUA(ZTARGET406),      IBIT<00> CLEAR
11914      LOAD=ERROR(TEST505C),      IERROR DIRECTORY KEY
11915      DCS=CTR(C4,),      ICOMPARE AT TARGET
11916      NEXT, J/GOBUT505C      I
(6722) DCS[1,00,1,0,0,0] BM[1011,00,11,11,00,000,110,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,000,0,010,010,011]
11917
11918      6223: 1(FREE)
11919      GOBUT505C:
11920      SETUP, RETURN/SCOPE505,      IRETURN TO SCOPE LOOP TEST WORD
11921      NEXT, GOTO=PAGE(7),          IBUT TABLE
11922      J/BUTMEXFLAG1      IEXFLAG<1> IN BIT<00>
(6223) DCS[0,00,0,0,0,0] BM[0110,00,01,00,10,100,111,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,100,0,011,010,111]
11923
11924      6224: 1(FREE)
11925      SCOPE505:
11926      P0,      BUMP=VERIFY,          ICOUNT
11927      BUSDIN_EMIT-[I],      IEMIT FOR CONSTANTS
11928      P3,      FLAG[8-0]_D[15-8]-[I],      IZERO ALL FLAGS, D WAS LEFT
11929      FPS[7-4]_D[7-4]-[I],      IZERO FROM TEST505A, IF ALL OK,
11930      NEXT, BUT[SCOPE],      INO ERROR: "TEST505A" (+1, WORDS)
11931      J/TEST506      IERROR: "LOAD505A" (-9, WORDS)
(6224) DCS[0,00,0,1,0,1] BM[0000,00,00,01,100,001,0,0,0,0,0,0,0,1,1011,0,0,0,0000,0,11,000,0,110,010,001]
11932
11933
11934
11935
11936
11937
11938
11939
11940      I - - - - -
11941      I*** TEST 506 ***
```

```
11942      ITESTS 506 A-E USE A "0-1-1010-1010" PATTERN IN PS<15,13,7:4,3:0>
11943
11944      I - - - - -
11945
11946      I* TEST 506A *
11947      ILOAD UP PS<HI,MID,LO> IN ORDER, READ BACK THRU PS PORT OF PROC MUX
11948      I ALSO CHECK THAT BUTA(CLR=FLAG-RES=UCON) CLEARS UCON REGISTER, SO THAT:
11949      I (1) BUSDIN_EMIT IS SELECTED, VIA UCON=SELECT(1)H=LOW, AND
11950      I (2) MAKE SURE THAT THE OTHER UCON BIT LATCHES ARE ALSO CLEARED
11951      6671:
11952      TEST506A:
11953      P0,      LOAD=ENUA(ZTARGET402),      ISETUP FOR D=ZERO TEST
11954      LOAD=ERROR(TEST506A),      IERROR DIRECTORY KEY
11955      DCS=CTR(C18,),      ISTALL FOR NOW
11956      NEXT, J/EXPEC506A      I
(6621) DCS[1,00,1,0,0,0] BM[0000,00,11,11,00,000,010,0,0,0,0,0,0,0,0000,0,0,0,0000,0,11,000,0,110,001,010]
11957
11958      6617:
11959      EXPEC506A:
11960      P3,      CSPD[02]_EMIT,          IEXPECTED VALUE TO BE READ OUT OF
11961      EMIT/030252,      IPS AFTER LOADING:
11962      NEXT, J/SETUCON406A      I"0011 0000 1010 1010"
(6617) DCS[0,00,0,0,0,0] BM[0011,00,00,10,101,010,0,0,0,0,0,0,0,0,1101,0,0,0,0000,0,11,000,0,010,010,101]
11963
11964      6225: 1(FREE)
11965      SETUCON506A:
11966      P0,      BUSDIN_FLAGS=FPS-[I],      ITAKE EMIT OFF BUSDIN, FLAG=FPS=(000005)
11967      P3,      BUTA(CLR=FLAG-RES=UCON),      ITHIS SHOULD NOW CLEAR THE UCON REGISTER,
11968      I SETTING UCON=SELECT(1)H=L, FORCING BUSDIN_EMIT
11969      NEXT, J/PSHI506A      I
(6225) DCS[0,00,0,0,0,0] BM[0000,00,00,11,01,000,000,0,0,0,0,0,0,0,1,1001,0,0,0,0000,0,11,010,0,010,010,110]
11970
11971      6226: 1(FREE)
11972      PSHI506A:
11973      P0,      BUMP=VERIFY,          ICOUNT
11974      P3,      CSPD[05]_EMIT,      IVALUE IN D WHEN LOAD PS<15,13>:
11975      EMIT/063125,      I"0110 0110 0101 0101"
11976      NEXT, J/PSMID506A      I
(6226) DCS[0,00,0,0,0,1] BM[0110,10,01,10,01,010,101,0,0,0,0,0,0,0,0,1010,0,1,0,0000,0,11,000,0,010,010,111]
11977
11978      6227: 1(FREE)
11979      PSMID506A:
11980      P3,      CSPD[06]_EMIT,      IVALUE IN D WHEN LOAD PS<7:4>:
11981      EMIT/143245,      I"1100 0110 1010 0101"
11982      NEXT, J/PSLOS06A      I
(6227) DCS[0,00,0,0,0,0] BM[1100,10,01,10,10,100,101,0,0,0,0,0,0,0,0,1001,0,1,0,0000,0,11,000,0,010,011,000]
11983
11984      6230: 1(FREE)
11985      PSLOS06A:
11986      P0,      BUMP=VERIFY,          ICOUNT
11987      P3,      CSPD[07]_EMIT,      IVALUE IN D WHEN LOAD PS<3:0>:
11988      EMIT/143132,      I"1100 0110 0101 1010"
11989      NEXT, J/PUDGEPS406A      I
(6230) DCS[0,00,0,0,0,1] BM[1100,10,01,10,01,010,010,0,0,0,0,0,0,0,0,1000,0,1,0,0000,0,11,000,0,010,011,001]
```

```

11990
11991      6231: 1(FREE)
11992      FUDGEPS06A:
11993      SETUP, RETURN/LOADUCON506A,          !GO TO SUBR WHICH:
11994                                          !1) CSP(05) -> PS<15,13>
11995                                          !2) CSP(06) -> PS<7:4>
11996                                          !3) CSP(07) -> PS<3:0>
11997      NEXT, CALL[PSSEQLOD]
11998      (6231) DCS[0.00,0.0,0.0] BM[0111..00.00..01.10..111..111...0.0,0.0,0.0...0.0000...0..0000,0...11,100...001,000,000]
11999
12000      7067: 1(FREE)
12001      LOADUCON506A:
12002      SETUP, UCON-PROC,                      !SETUP FOR UCON-PROC:
12003      ENABLE, SUBDIR_FLAGS#FPS-(1),          !FLAGS#FPS=(000005)
12004      EN-CLK-PS(15-12), EN-CLK-PS(7-6),      !SETUP FOR CLOCKING PS BITS
12005      EN-CLK-PS(3-0),
12006      PO, SET-UCON-CONTROL,
12007      P3, BUTA[CLR-FLAG-RES-UCON],           !WRITE THE UCON REGISTER AT PO,
12008      NEXT, J/DOITS06A                       !BUT AT P3, NOW CLEAR IT OUT
12009      (7067) DCS[0.00,0.0,0.0] BM[1000..00.00..11.01..010..010...0.0,0.0,0.0...1.1001...0..0000,0...11,010...000,111,010]
12010
12011      7072: 1(FREE)
12012      DOITS06A:
12013      P2-T, D_ZERO,                          !SETUP D WITH A (000000)
12014      NEXT, GOTO-PAGE(4),                     !XFER
12015      J/LOADDCS06A
12016      (7072) DCS[0.00,0.0,0.0] BM[0011..00.00..00.00..000..110...0.0,0.0,0.0...0.0000...0..0000,0...11,100...001,000,100]
12017
12018      6104: 1(FREE)
12019      LOADDCS06A:
12020      PO, DCS-CTR(C9.),                      !COMPARE AT TARGET
12021      P2, UCON-OPERATION,                    !IF THE UCON REGISTER WASN'T CLEARED, ABOVE,
12022      NEXT, GOTO-PAGE(7),                     ! THE PS SHOULD BE OVERWRITTEN WITH (000000),
12023      J/GOBUTS06A                             ! IN SOME COMBINATION OF PS<15:12,7:4,3:0>
12024      (6104) DCS[0.00,1.0,0.0] BM[0110..00.00..00.00..000..111...0.0,0.0,0.0...1.1010...0..0000,0...11,100...000,111,001]
12025
12026      7071: 1(FREE)
12027      GOBUTS06A:
12028      SETUP, RETURN/TESTS06B,                 !RETURN TO START OF NEXT SUBTEST
12029      NEXT, CALL[PSCTD]                      ! PS=H0R-CSP(02) -> D, BUT(D=ZERO)
12030      (7071) DCS[0.00,0.0,0.0] BM[0110..00.11..11.01..100..111...0.0,0.0,0.0...0.0000...0..0000,0...11,100...010,101,011]
12031
12032      | - - - - -
12033      !* TEST 506B *
12034      !CHECK THAT BUTP(PS15) SHOWS PS<15>H CLEAR
12035      6754:
12036      TEST506B:
12037      PO, LOAD-ENUA(ZTARGET402),              !BIT<00> CLEAR

```

```

12036      LOAD-ERROR(TEST506B),                 !ERROR DIRECTORY KEY
12037      DCS-CTR(C3.),                          !COMPARE AT TARGET
12038      NEXT, J/GOBUTS06B
12039      (6754) DCS[1.00,1.0,0.0] BM[1100..00.11..11.00..000..010...0.0,0.0,0.0...0.0000...0..0000,0...11,000...010,011,011]
12040
12041      6233: 1(FREE)
12042      GOBUTS06B:
12043      SETUP, RETURN/TESTS06C,                 !RETURN TO START OF NEXT SUBTEST
12044      NEXT, GOTO-PAGE(7),                     !BUT TABLE HERE
12045      J/BUTPS15                               !PS<15>H IN BIT<00>
12046      (6233) DCS[0.00,0.0,0.0] BM[0110..00.11..11.00..100..111...0.0,0.0,0.0...0.0000...0..0000,0...11,100...011,100,011]
12047
12048      | - - - - -
12049      !* TEST 506C *
12050      !CHECK THAT BUTM(PS03) SHOWS PS<03>H SET
12051      6744:
12052      TEST506C:
12053      PO, LOAD-ENUA(ZTARGET407),              !BIT<00> SET
12054      LOAD-ERROR(TEST506C),                  !ERROR DIRECTORY KEY
12055      DCS-CTR(C4.),                          !COMPARE AT TARGET
12056      BUMP-VERIFY,
12057      NEXT, J/GOBUTS06C                       !COUNT
12058      (6744) DCS[1.00,1.0,0.1] BM[1011..00.11..11.00..000..111...0.0,0.0,0.0...0.0000...0..0000,0...11,000...010,011,100]
12059
12060      6234: 1(FREE)
12061      GOBUTS06C:
12062      SETUP, RETURN/TESTS06D,                 !RETURN TO START OF NEXT SUBTEST
12063      NEXT, GOTO-PAGE(7),                     !BUT TABLE HERE
12064      J/BUTMPS15                             !PS(N)=PS<03>H IN BIT<00>
12065      (6234) DCS[0.00,0.0,0.0] BM[0110..00.11..10.11..100..111...0.0,0.0,0.0...0.0000...0..0000,0...11,100...011,010,011]
12066
12067      | - - - - -
12068      !* TEST 506D *
12069      !CHECK THAT WHEN FLAG<0>H=L AND PS<04>H=L,
12070      ! MASKED-PS(T)-H=FLAG<0>L=PS<04>H, IS LOW
12071      6734:
12072      TEST506D:
12073      PO, LOAD-ENUA(ZTARGET406),              !BIT<00> CLEAR
12074      LOAD-ERROR(TEST506D),                  !ERROR DIRECTORY KEY
12075      DCS-CTR(C4.),                          !COMPARE AT TARGET
12076      NEXT, J/GOBUTS06D
12077      (6734) DCS[1.00,1.0,0.0] BM[1011..00.11..11.00..000..110...0.0,0.0,0.0...0.0000...0..0000,0...11,000...010,011,101]
12078
12079      6235: 1(FREE)

```

```
12083 G0BUTS06D:
12084 PO, BUMP-VERIFY, ICOUNT
12085 SETUP, RETURN/TESTS06E, IRETURN TO START OF NEXT SUBTEST
12086 NEXT, GOTO-PAGE(7), I BUT TABLE HERE
12087 J/BUTMASKPS(T) I FLAG<00>=PS<00> IN BIT<00>
(6235) DCS(0,00,0,0,0,1) BM(0110..00,11..00,00..101..111...0,0,0,0,0...0,0000...0..0000,0...11,100...011,001,111)
12088
12089
12090
12091
12092 | - - - - -
12093
12094 I* TEST 506 *
12095 I CHECK THAT INTR-HIGH-H=SERVICE(0)H=NOT(MASKED-PS(T)-H) IS HIGH,
12096 I WHEN SERVICE(0)H=HIGH, MASKED-PS(T)-H=LOW
12097 I NOTE: THE SUBR CALLED SHOULD SET SERVICE(0)H=HIGH (NEGATED) BY CLEARING OUT ALL ITS INPUT CONDITIONS
12098 6605:
12099 TESTS06E:
12100 PO, LOAD-ENUA(ZTARGET403), I BIT<04> SET
12101 LOAD-ERRDR(TESTS06E), I ERROR DIRECTORY KEY
12102 DCS-CTR(C9,), I COMPARE AT TARGET
12103 NEXT, J/CLEAR506E I
(6605) DCS(1,00,1,0,0,0) BM(0110..00,11..11,00..000..011...0,0,0,0,0...0,0000...0..0000,0...11,000...010,011,110)
12104
12105 6236: I(FREE)
12106 CLEAR506E:
12107 PO, BUMP-VERIFY, I COUNT
12108 SETUP, RETURN/G0BUTS06E, I GO TO SUBR WHICH CLEARS OUT
12109 NEXT, CALL(CLEAR-I-0-B) I ALL THE UNIBUS/I-O/SERVICE LATCHES (NO UNIBUS INIT)
(6236) DCS(0,00,0,0,0,0) BM(0110..00,01..00,11..111..111...0,0,0,0,0...0,0000...0..0000,0...11,100...010,011,000)
12110
12111 6237: I(FREE)
12112 G0BUTS06E:
12113 SFTUP, RETURN/SCOPE506, I RETURN TO SCOPE LOOP TEST WORD
12114 NEXT, GOTO-PAGE(7), I BUT TABLE HERE
12115 J/BUTINTRHIGH I INTR HIGH H IN BIT<01>
(6237) DCS(0,00,0,0,0,0) BM(0110..00,01..01,00..000..111...0,0,0,0,0...0,0000...0..0000,0...11,100...011,101,101)
12116
12117
12118
12119
12120 6240: I(FREE)
12121 SCOPE506:
12122 PO, BUSDIN_EMIT(I), I KEEP EMIT FOR CONSTANTS
12123 P2, PS_D(I), I ZERO PS; D LEFT ZERO FROM
12124 I PREVIOUS TESTS IF ALL OK
12125 NEXT, BUID(SCOPE), I NO ERROR; "TEST507A" (+1, WORDS)
12126 J/TEST507A I ERROR; "MASK506A" (+16, WORDS)
(6240) DCS(0,00,0,1,0,0) BM(1000..00,00..00,01..010..010...0,0,0,0,0...0,1011...0..0000,0...11,000...110,001,011)
12127
12128
```

```
12129
12130
12131
12132
12133 | - - - - -
12134
12135 I*** TEST 507 ***
12136
12137 I TESTS 507 A-F USE A "1-0-0101-0101" PATTERN IN PS<15,13,7;4,3;0>
12138
12139 | - - - - -
12140
12141 I* TEST 507A *
12142 I LOAD UP PS<HI,MID,LO> IN ORDER, READ BACK THRU PS PORT OF PROC MUX
12143 6613:
12144 TEST507A:
12145 PO, LOAD-ENUA(ZTARGET402), I SETUP FOR D=ZERO TEST
12146 LOAD-ERRDR(TEST507A), I ERROR DIRECTORY KEY
12147 DCS-CTR(C15,), I INSTALL FOR NOW
12148 NEXT, J/EXPEC507A I
(6613) DCS(1,00,1,0,0,0) BM(0000..00,11..11,00..000..010...0,0,0,0,0...0,0000...0..0000,0...11,000...110,000,010)
12149
12150 6602:
12151 EXPEC507A:
12152 P3, CSPD(02)_EMIT, I EXPECTED VALUE TO BE READ OUT OF
12153 EMIT/140125, I PS AFTER LOADING:
12154 NFXT, J/PSHI507A I "1100 0000 0101 0101"
(6602) DCS(0,00,0,0,0,0) BM(1100..10,00..00,01..010..101...0,0,0,0,0...0,1101...1..0000,0...11,000...010,100,001)
12155
12156 6241: I(FREE)
12157 PSHI507A:
12158 PO, BUMP-VERIFY, I COUNT
12159 P3, CSPD(05)_EMIT, I VALUE IN D WHEN LOAD PS<15,13>:
12160 EMIT/113252, I "1001 0110 1010 1010"
12161 NEXT, J/PSMID507A I
(6241) DCS(0,00,0,0,0,1) BM(1001..10,01..10,10..101..010...0,0,0,0,0...0,1010...1..0000,0...11,000...010,100,010)
12162
12163 6242: I(FREE)
12164 PSMID507A:
12165 P3, CSPD(06)_EMIT, I VALUE IN D WHEN LOAD PS<7;4>:
12166 EMIT/033122, I "0011 0110 0101 1010"
12167 NFXT, J/PSLO507A I
(6242) DCS(0,00,0,0,0,0) BM(0011..10,01..10,01..010..010...0,0,0,0,0...0,1001...1..0000,0...11,000...010,100,011)
12168
12169 6243: I(FREE)
12170 PSLO507A:
12171 PO, BUMP-VERIFY, I COUNT
12172 P3, CSPD(07)_EMIT, I VALUE IN D WHEN LOAD PS<3;0>:
12173 EMIT/033245, I "0011 0110 1010 0101"
12174 NFXT, J/FUDGEPS507A I
(6243) DCS(0,00,0,0,0,1) BM(0011..10,01..10,10..100..101...0,0,0,0,0...0,1000...1..0000,0...11,000...010,100,100)
12175
12176 6244: I(FREE)
12177 FUDGEPS507A:
12178
```

```

12178      SETUP,      RETURN/LOADDCS507A,      100 TO SUBR WHICH:
12179      NEXT,        CALL{PSSEOL00}             11) CDP{P5} -> PS<15,13>
(6244) DCS10,00,0,0,0,0] BM10110..00,01..01,00..101..111...0,0,0...0,0,0000...0,0000,0...11,100...001,000,000]
12180                                     12) CDP{06} -> PS<7,14>
12181                                     13) CDP{07} -> PS<310>
12182
12183      6245: 1(FREE)
12184      LOADDCS507A:
12185      PO,          DCS-CTR{C9,},              1COMPARE AT TARGET
12186      NEXT,        J/GOBUTS07A                1
(6245) DCS10,00,1,0,0,0] BM10110..00,00,00,00,000,000...0,0,0,0,0...0,0000...0,0000,0...11,000...010,100,110]
12187
12188      6246: 1(FREE)
12189      GOBUTS07A:
12190      SETUP,      RETURN/TESTS07B,             1RETURN TO START OF NEXT SUBTEST
12191      NEXT,        CALL{PSTOD}                 1 CDP{03}-XOR-PS -> D, BUT{D=ZERO}
(6246) DCS10,00,0,0,0,0] BM10110..00,11..10,10,100,111...0,0,0,0,0...0,0000...0,0000,0...11,100...010,101,011]
12192
12193
12194
12195
12196      1 - - - - -
12197
12198      1* TEST 507B *
12199      1CHECK THAT BUTR{P815} SHOWS PS<18>H SET
12200      6724:
12201      TESTS07B:
12202      PO,          LOAD=ENUA(2TARGET403),       1BIT<00> SET
12203      LOAD=ERROR{TESTS07B},                     1ERROR DIRECTORY KEY
12204      NEXT,        DCS-CTR{C3,},                1COMPARE AT TARGET
12205      J/GOBUTS07B                                1
(6724) DCS11,00,1,0,0,0] BM11100..00,11..11,00,000,001...0,0,0,0,0...0,0000...0,0000,0...11,000...010,100,111]
12206
12207      6247: 1(FREF)
12208      GOBUTS07B:
12209      SETUP,      RETURN/TESTS07C,             1RETURN TO START OF NEXT SUBTEST
12210      NEXT,        GOTO=PAGE(7),                1BUT TABLE HERE
12211      J/BUTPS15                                   1PS<15>H IN BIT<00>
(6247) DCS10,00,0,0,0,0] BM10110..00,11..00,01,001,111...0,0,0,0,0...0,0000...0,0000,0...11,100...011,100,011]
12212
12213
12214
12215
12216      1 - - - - -
12217
12218      1* TEST 507C *
12219      1CHECK THAT BUTN{P803} SHOWS PS<03>H CLEAR
12220      6611:
12221      TESTS07C:
12222      PO,          LOAD=ENUA(2TARGET406),       1BIT<00> CLEAR
12223      LOAD=ERROR{TESTS07C},                     1ERROR DIRECTORY KEY

```

```

12224 DCS=CTR(C4.), COMPARE AT TARGET
12225 BUMP=VERIFY, ICOUNT
12226 JGOBUT507C
(6611) DCS(1.00,1.0,0.1) BM(1011..00.11..11.00..000..110...0,0,0..0..0...0.0000...0...0000,0...11.000...010,101,000)

12227
12228 6250: 1(FREF)
12229 GORUT507C1
12230 SETUP, RETURN/TEST507D, IRETURN TO START OF NEXT SUBTEST
12231 NEXT, GOTO=PAGE(7), I BUT TABLE HERE
12232 J/BUTMP8(N) IFLAG<0>L<PS<04>H IN BIT<00>
(6250) DCS(0.00,0,0,0.0) BM(0110..00.10..11.01..111..111...0,0,0..0..0...0.0000...0...0000,0...11.100...011,010,011)

12233
12234
12235
12236
12237 | - - - - -
12238
12239 I* TEST 507D *
12240 I CHECK THAT WHEN FLAG<0>H=L AND PS<04>H=H,
12241 I MASKED=PS(T)-H=FLAG<0>L<PS<04>H, IS HIGH
12242 6557:
12243 TEST507D:
12244 PO, LOAD=ENUA(ZTARGET407), I BIT<00> SET
12245 LOAD=ERRR(TEST507D), I ERROR DIRECTORY KEY
12246 DCS=CTR(C4.), I COMPARE AT TARGET
12247 JGOBUT507D
(6557) DCS(1.00,1.0,0.0) BM(1011..00.11..11.00..000..111...0,0,0..0..0...0.0000...0...0000,0...11.000...010,101,001)

12248
12249 6251: 1(FREE)
12250 GORUT507D:
12251 PO, BUMP=VERIFY, I COUNT
12252 SETUP, RETURN/TEST507E, I RETURN TO START OF NEXT SUBTEST
12253 NEXT, GOTO=PAGE(7), I BUT TABLE HERE
12254 J/BUTMASKPS(T) I FLAG<0>L<PS<04>H IN BIT<00>
(6251) DCS(0.00,0,0,0.0,1) BM(0110..00.11..01.11..100..111...0,0,0..0..0...0.0000...0...0000,0...11.100...011,001,111)

12255
12256
12257
12258
12259 | - - - - -
12260
12261 I* TEST 507E *
12262 I CHECK THAT INTR-HIGH=H=SERVICE(0)H=NOT(MASKED=PS(T)-H) IS LOW,
12263 I WHEN SERVTCF(0)H=HIGH, MASKED=PS(T)-H=HIGH
12264 6674:
12265 TEST507E:
12266 PO, LOAD=ENUA(ZTARGET401), I BIT<01> CLEAR
12267 LOAD=ERRR(TEST507E), I ERROR DIRECTORY KEY
12268 DCS=CTR(C3.), I COMPARE AT TARGET
12269
12270 RUMP=VERIFY, I COUNT
12271 JGOBUT507E
(6674) DCS(1.00,1.0,0.1) BM(1100..00.11..11.00..000..001...0,0,0..0..0...0.0000...0...0000,0...11.000...010,101,011)

```

```

12271
12272 6253: 1(FREE)
12273 GOBUT507F:
12274 SETUP, RETURN/TEST507F, ;RETURN TO START OF NEXT SUBTEST
12275 NEXT, GOTO-PAGE(7), ;BUT TABLE HERE
12276 J/ROUTINTRNHIGH ;INTR HIGH N IN BIT<01>
(6253) DCS[0.00,0.0,0,0] BM[0110..00.11..01.00..111..111...0,0,0,0...0.0000...0..0000,0...11.100...011,101,101]
12277
12278
12279
12280
12281
12282 !* TEST 507F *
12283 !CHECK THAT PS[C]-H CAN BE READ AS A "1" THRU D[C]-CINMUX=PS[C], AND D[C]-PS[C]
12284 ! IT WAS PREVIOUSLY CHECKED, IN ALU LOGIC TESTS, THAT THESE WERE OK WITH PS[C]="0"
12285 6647:
12286 TEST507F:
12287 P0, LOAD-ENUA(ZTARGET407), ;SETUP FOR SR<3:0> = "0111"
12288 LOAD-ERROR(TEST507F), ;ERROR DIRECTORY KEY
12289 DCS-CTR(C7), ;COMPARE AT TARGET
12290 NEXT, J/CIN507F
(6647) DCS[1.00,1.0,0,0] BM[1000..00.11..11.00..000..111...0,0,0,0...0.0000...0..0000,0...11.000...010,101,100]
12291
12292 6254: 1(FREE)
12293 CIN507F:
12294 P0, BUMP-VERIFY, ;COUNT
12295 P2-T, D_ZERO, DIC]-CINMUX, ;SET D[C]=CINMUX=PS[C]-(1)
12296 SR_ZERO, ;SET SR<3:0>="0000"
12297 P3, BSPLO[17]_D, ;B-SIDE COPY OF SR
12298 NEXT, J/SETP5507F
(6254) DCS[0.00,0.0,0,1] BM[0011..00.11..00.00..011..000...0,1,1,0...0.0000...0..0110,0...11.000...010,101,110]
12299
12300 6256: 1(FREE)
12301 SETP5507F:
12302 P2-T, D_A-PLUS-B-PLUS-DIC], D[C]-PS[C], ;SET D[C]=PS[C]-(1)
12303 SR_A-PLUS-B-PLUS-D[C], ;D, SR, BSP <- SR-LEFT-1, SR<02> <- PREV D[C] = CINMUX/PS[C]
12304 BUS-A_SR, ;
12305 BUS-B_BSPLO(R17), ;
12306 P3, BSPLO[17]_D, ;SAVE ON B-SIDE
12307 NEXT, J/SETSR507F
(6256) DCS[0.00,0.0,0,0] BM[0100..00.11..00.00..011..001...0,1,1,0...0.0000...0..0110,0...11.000...010,101,111]
12308
12309 6257: 1(FREE)
12310 SETSR507F:
12311 P0, BUMP-VERIFY, ;COUNT
12312 P2-T, D_A-PLUS-B-PLUS-DIC], D[C]-ALU15, ;SET D[C]=0
12313 SR_A-PLUS-B-PLUS-D[C], ;D, SR, BSP <- SR-LEFT-1, SR<01> <- PREV D[C] = PS[C]
12314 BUS-A_SR, ;
12315 BUS-B_BSPLO(R17), ;
12316 P3, BSPLO[17]_D, ;SAVE ON B-SIDE
12317 NEXT, J/SETCIN507F
(6257) DCS[0.00,0.0,0,1] BM[0100..00.11..00.00..011..100...0,1,1,0...0.0000...0..0110,0...11.000...010,110,000]
12318

```

```

12319 6260: 1(FREE)
12320 SETCIN507F:
12321 P2-T, D_A-PLUS-B-PLUS-PS[C], D[C]-ALU15, ;SET D[C]=0
12322 SR_A-PLUS-B-PLUS-PS[C], ;D, SR, BSP <- SR-LEFT-1, SR<00> <- CIN/PS[C]
12323 BUS-A_SR, ;
12324 BUS-B_BSPLO(R17), ;
12325 NEXT, J/GOBUT507F
(6260) DCS[0.00,0.0,0,0] BM[0001..00.11..00.00..011..100...0,1,1,0...0.0000...0..0000,0...11.000...010,110,001]
12326
12327 6261: 1(FREE)
12328 GOBUT507F:
12329 SETUP, RETURN/SCOPE507, ;RETURN TO SCOPE LOOP TEST WORD
12330 NEXT, GOTO-PAGE(7), ;BUT TABLE
12331 J/BUTSR3=0 ;CHECK THAT WE GOT "0111" = (07)
(6261) DCS[0.00,0.0,0,0] BM[0110..00.01..01.10..010..111...0,0,0,0...0.0000...0..0000,0...11.100...010,111,110]
12332
12333
12334
12335
12336
12337 6262: 1(FREE)
12338 SCOPE507:
12339 P0, BUSDIN_EMIT[1], ;KEEP EMIT FOR CONSTANTS
12340 P2, PS-D[1], ;ZERO PS; D LEFT ZERO FROM
12341 ;PREVIOUS TESTS IF ALL OK
12342 NEXT, RUTD[SCOPE], ;NO ERROR: "TEST510A" (+1, WORDS)
12343 J/TEST510A ;ERROR: "EXPEC507A" (-21, WORDS)
(6262) DCS[0.00,0.1,0,0] BM[1000..00.00..00.01..010..010...0,0,0,0...0.0000...0..0000,0...11.000...110,000,011]
12344
12345
12346
12347
12348
12349
12350
12351
12352
12353
12354 !*** TEST 510 ***
12355 !TESTS 510 A-F USE PS<7:5>H="111", PS<4>H="1", VARIOUS FLAG<7,0>H COMBINATIONS,
12356 ! TO TEST THE INTP-HIGH-H, SERVICE-H, AND MASKED-PS[T]-H LOGIC
12357
12358
12359
12360 !* TEST 510A *
12361 !CHECK THAT BG-SERVICE(0)H=HIGH WHEN PS<7:5>H="111" (PSN PRIORITY 7); EG, BR>PS-H=LOW, SINCE NO
12362 ! EXTERNAL UNIBUS DEVICE CAN THEN REQUEST AN INTERRUPT (IE, IT IS MASKED OUT)
12363
12364 6603:
12365 TFS510A:
12366 P0, LOAD-ENUA(ZTARGET407), ;IRIT<02> SET
12367 LOAD-ERROR(TEST510A), ;ERROR DIRECTORY KEY
12368 DCS-CTR(C5), ;COMPARE AT TARGET
12369 NEXT, J/SETONE510A
(6603) DCS[1.00,1.0,0,0] BM[1010..00.11..11.00..000..111...0,0,0,0...0.0000...0..0000,0...11.000...110,000,110]

```

```
12368
12369 6606:
12370 SETONES10A:
12371 P2-T, D_ALL-ONES, SAVE-D[C], |ALL ONES FOR LOAD PRIORITY 7
12372 NEXT, J/LOADPRIOS10A
(6606) DCS[0.00,0.0,0.0] BM[1111.00.00.11.01.101.111...0.1,0.0.0.0...0.0000...0.0000,0...11.000...010,110,011]
12373
12374 6263: 1(FREE)
12375 LOADPRIOS10A:
12376 PO, BUSDIN_EMIT-[I], |KEEP IT ON
12377 P3-T, PS[7-4]_D[7-4]-[I], |PRIO 7, T-BIT (PS04) SET
12378 NEXT, J/GOBUTS10A
(6263) DCS[0.00,0.0,0.0] BM[0000.00.00.00.01.010.000...1,0,0.0.0...1.1011...0.0000,0...11.000...010,110,100]
12379
12380 6264: 1(FREE)
12381 GOBUTS10A:
12382 SETUP, RETURN/TESTS10B, |RETURN TO START OF NEXT SUBTEST
12383 NFXT, GOTO-PAGE(7), |BUT TABLE
12384 J/BUTSGSERVL |BUT-SERVICE(0)H IN BIT<02>
(6264) DCS[0.00,0.0,0.0] BM[0110.00.11.01.11.101.111...0,0,0.0.0...0.0000...0.0000,0...11.100...011,001,110]
12385
12386
12387
12388
12389
12390 | - - - - -
12391
12392 1* TEST 510B *
12393 |CHECK THAT BUT[ NOT(BG-SERVICE(0)H) + NOT(FLTPT-SERVICE-L) ] IN BIT<03>
12394 | SEES BG-SERVICE(0)H=HIGH, FLTPT-SERVICE-L=HIGH, AND THUS IS NOT ASSERTED (=LOW)
12395 6675:
12396 TEST510B:
12397 PO, LOAD-ENUA(ZTARGET407), |BIT<03> CLEAR
12398 LOAD-ERROR(TEST510B), |ERROR DIRECTORY KEY
12399 DCS-CTR(C3), |COMPARE AT TARGET
12400 NEXT, J/GOBUTS10B
(6675) DCS[1.00,1.0,0.0] BM[1100.00.11.11.00.000...111...0,0,0.0.0...0.0000...0.0000,0...11.000...010,110,101]
12401
12402 6265: 1(FREE)
12403 GOBUTS10B:
12404 SETUP, RETURN/TESTS10C, |RETURN TO START OF NEXT SUBTEST
12405 NEXT, GOTO-PAGE(7), |BUT TABLE
12406 J/BUTSGFPSERV |RESULT IN BIT<03>
(6265) DCS[0.00,0.0,0.0] BM[0110.00.11.00.01.110.111...0,0,0.0.0...0.0000...0.0000,0...11.100...011,000,111]
12407
12408
12409
12410
12411
12412 | - - - - -
12413
12414 1* TEST 510C *
```

```
12415 |CHECK THAT SERVICE-H=INTR-HIGH-H=BG-SERVICE(0)H=NOT(FLAG<7>H) IS HIGH,
12416 | WHEN INTR-HIGH-H=LOW, BG-SERVICE(0)H=HIGH, FLAG<7>H=LOW
12417 6616:
12418 TEST510C:
12419 PO, LOAD-ENUA(ZTARGET403), |BIT<00> SET
12420 LOAD-ERROR(TEST510C), |ERROR DIRECTORY KEY
12421 DCS-CTR(C3), |COMPARE AT TARGET
12422 NEXT, J/GOBUTS10C
(6616) DCS[1.00,1.0,0.0] BM[1100.00.11.11.00.000...011...0,0,0.0.0...0.0000...0.0000,0...11.000...010,110,110]
12423
12424 6266: 1(FREE)
12425 GOBUTS10C:
12426 SETUP, RETURN/TESTS10D, |RETURN TO START OF NEXT SUBTEST
12427 NEXT, GOTO-PAGE(7), |BUT TABLE
12428 J/BUTSERVICE |SERVICE-H IN BIT<00>
(6266) DCS[0.00,0.0,0.0] BM[0110.00.11.00.00.000...111...0,0,0.0.0...0.0000...0.0000,0...11.100...011,100,101]
12429
12430
12431
12432 | - - - - -
12433
12434 1* TEST 510D *
12435 |SET ONLY FLAG<0>H, FLAG<7>H, LEAVE PS<04>H SET,
12436 |CHECK MASKED-PS[1]-H=PS<04>H=FLAG<0>H IS LOW
12437 6600:
12438 TEST510D:
12439 PO, LOAD-ENUA(ZTARGET406), |BIT<00> CLEAR
12440 LOAD-ERROR(TEST510D), |ERROR DIRECTORY KEY
12441 DCS-CTR(C7), |COMPARE AT TARGET
12442 BUMP-VERIFY, |COUNT
12443 NEXT, J/FLAGS10D
(6600) DCS[1.00,1.0,0.1] BM[1000.00.11.11.00.000...110...0,0,0.0.0...0.0000...0.0000,0...11.000...010,110,111]
12444
12445 6267: 1(FREE)
12446 FLAGS10D:
12447 P3, CSPD[01]_EMIT, |
12448 EMIT/040401, |BITS FOR FLAG<7,0> ONLY, BIT<00> OF D
12449 NEXT, J/ENFLAGS10D
(6267) DCS[0.00,0.0,0.0] BM[0100.10.00.01.00.000...001...0,0,0.0.0...0.1110...1.0000,0...11.000...010,111,000]
12450
12451 6270: 1(FREE)
12452 ENFLAGS10D:
12453 P3-T, D_CSPD(D01), D[C]-0, |INTO D
12454 NEXT, J/SETFLAGS10D
(6270) DCS[0.00,0.0,0.0] BM[1010.10.00.00.00.000...000...1,1,0.0.0.0...1.110...0.0000,0...11.000...010,111,001]
12455
12456 6271: 1(FREE)
12457 SETFLAGS10D:
12458 P3-T, FLAG[0-0]_D[15-8]-[I], |SET FLAGS<7,0>H ONLY
12459 NFXT, J/GOBUTS10D
(6271) DCS[0.00,0.0,0.0] BM[0000.00.00.00.01.000...001...1,0,0.0.0...1.1011...0.0000,0...11.000...010,111,010]
12460
12461 6272: 1(FREE)
```

```
12462 G0BUTS10D1
12463     SETUP, RETURN/TESTS10DA,      IRETURN TO START OF NEXT SUBTEST
12464     NEXT,  GOTO-PAGE(7),           IBUT TABLE HERE
12465     J/BUTMHASKPSIT)               IFLAG<0>=PS<04>H IN BIT<00>
(6272) DCS[0.00,0.0,0.0] BM[0110..00.11..00.01..111..111...0.0,0.0,0.0...0.0000...0.0000,0...11,100...011,001,111]

12466
12467
12468
12469
12470
12471
12472 I* * * * *
12473 I*** TEST 510DA ***
12474 I DO THE "MULTIPLE BUT" ON "D<00>H" TO CHECK IT'S SET (DURING TESTS10D, ABOVE)
12475 66171
12476 TESTS10DA:
12477     PO,      LOAD-ENUA(ZTARGET407),      IBIT<00> SET
12478     LOAD-ERROR(TESTS10DA),              IERROR DIRECTORY KEY
12479     DCS-CTR(C4.),                       ICOMPARE AT TARGET
12480     BUMP-VERIFY,                        ICOUNT
12481     NFXT,   J/G0BUTS10DA
(6617) DCS[1.00,1.0,0.1] BM[1011..00.11..11,00..000..111...0.0,0.0,0.0...0.0000...0.0000,0...11,000...010,111,011]

12482
12483 6273: I(PREF)
12484 G0BUTS10DA:
12485     SETUP, RETURN/TESTS10E,      IRETURN TO START OF NEXT SUBTEST
12486     NEXT,  GOTO-PAGE(7),           IBUT TABLE
12487     J/BUTMD00                      ID<00>=H IN BIT<00>
(6273) DCS[0.00,0.0,0.0] BM[0110..00.11..00,00..100..111...0.0,0.0,0.0...0.0000...0.0000,0...11,100...011,010,001]

12488
12489
12490
12491
12492
12493 I* * * * *
12494 I* TEST 510E *
12495 I CHECK THAT SERVICE-H=INTR-HIGH-H=BG-SERVICE(0)H=NOT(FLAG<7>H) IS HIGH,
12496 I WHEN INTR-HIGH-H=HIGH, BG-SERVICE(0)H=HIGH, FLAG<7>H=HIGH
12497 66041
12498 TESTS10E:
12499     PO,      LOAD-ENUA(ZTARGET403),      IBIT<00> SET
12500     LOAD-ERROR(TESTS10E),              IERROR DIRECTORY KEY
12501     DCS-CTR(C4.),                       ICOMPARE AT TARGET
12502     NFXT,   J/ZERODS10E
(6604) DCS[1.00,1.0,0.0] BM[1011..00.11..11,00..000..011...0.0,0.0,0.0...0.0000...0.0000,0...11,000...010,111,100]

12503
12504 6274: I(FREE)
12505 ZERODS10E:
12506     P2-T,   D_ZERO, DIC1-ALU15,        IZERODS FOR BELOW
12507     NFXT,   J/G0BUTS10E
(6274) DCS[0.00,0.0,0.0] BM[0011..00.00..00,00..000..100...0.1,0.0,0.0...0.0000...0.0000,0...11,000...010,111,101]

12508
12509 6275: I(FREE)
```

```
12509 G0BUTS10F:
12510     PO,      BUMP-VERIFY,              ICOUNT
12511     SETUP, RETURN/TESTS10F,          IRETURN TO START OF NEXT SUBTEST
12512     NEXT,  GOTO-PAGE(7),           IBUT TABLE HERE
12513     J/BUTSERVICE                   ISERVICE-H IN BIT<00>
(6275) DCS[0.00,0.0,0.1] BM[0110..00.11..00,01..000..111...0.0,0.0,0.0...0.0000...0.0000,0...11,100...011,100,101]

12514
12515
12516
12517
12518
12519
12520 I* * * * *
12521 I* TEST 510F *
12522 I CHECK THAT SERVICE-H=INTR-HIGH-H=BG-SERVICE(0)H=NOT(FLAG<7>H) IS LOW,
12523 I WHEN INTR-HIGH-H=HIGH, BG-SERVICE(0)H=HIGH, FLAG<7>H=LOW
12524 66101
12525 TESTS10F:
12526     PO,      LOAD-ENUA(ZTARGET402),      IBIT<00> CLEAR
12527     LOAD-ERROR(TESTS10F),              IERROR DIRECTORY KEY
12528     DCS-CTR(C4.),                       ICOMPARE AT TARGET
12529     NFXT,   J/ZEROF1AGS10E
(6610) DCS[1.00,1.0,0.0] BM[1011..00.11..11,00..000..010...0.0,0.0,0.0...0.0000...0.0000,0...11,000...010,111,110]

12530
12531 6276: I(PREF)
12532 ZEROF1AGS10E:
12533     P2,      PS-D-[I],                IZERO PS-T-BIT, AND ALL THE OTHERS,
12534     FPS(7-4)-D[7-4]-[I],              IZERO THE FPS
12535     P3,      FLAG[8-0]-D[15-8]-[I],    IZERO FLAGS (D ZEROED ABOVE)
12536     NEXT,   J/G0BUTS10F
(6276) DCS[0.00,0.0,0.0] BM[1000..00.00..00,01..110..011...0.0,0.0,0.0...1.1011...0.0000,0...11,000...010,111,111]

12537
12538 6277: I(FREE)
12539 G0BUTS10F:
12540     SETUP, RETURN/SCOPE510,          IRETURN TO SCOPE LOOP TEST WORD
12541     NEXT,  GOTO-PAGE(7),           IBUT TABLE HERE
12542     J/BUTSERVICE                   ISERVICE-H IN BIT<00>
(6277) DCS[0.00,0.0,0.0] BM[0110..00.01..10,00..000..111...0.0,0.0,0.0...0.0000...0.0000,0...11,100...011,100,101]

12543
12544
12545
12546
12547 6300: I(FREE)
12548 SCOPE510:
12549     PO,      BUSDIN_EMIT-[I],          IKEEP EMIT FOR CONSTANTS
12550     P2,      PS-T-[I],                IZERO PS; D LEFT ZERO FROM

12551
12552     NFXT,   BUTD[SCOPE],              IPREVIOUS TESTS IF ALL OK
12553     J/TESTS11A                       I NO ERROR: "TESTS11A" (+11, WORDS)
12554     I ERROR: "SETONES10A" (-20, WORDS)
(6300) DCS[0.00,0.1,0.0] BM[1000..00.00..00,01..010..010...0.0,0.0,0.0...1.1011...0.0000,0...11,000...110,000,111]

12555
```

```

12556
12557 | - - - - -
12558
12559 |
12560 | THIS FIRST SUBROUTINE COPIES:
12561 | CSP(06) -> PS<7:4>
12562 | CSP(05) -> FLAGS<8:4,2:0>, EXFLAG<2:1>
12563 | THEN RETURNS
12564 |
12565 | 7073: 1(FREE)
12566 | FLAGFPPSEQL0D1
12567 | P3-T, D_CSPD(D06), D(C)-0, |GET VALUE TO LOAD TO PS<7:4>H
12568 | NEXT, J/FLAGFPPS02 |
(7073) DCS(0.00,0.0,0.0,0) BM(1010..10.00..00.00..000..000...1.1,0..0..0..0.1001...0..0000,0...11,000...000,111,1011)

12569 |
12570 | 7075: 1(FREE)
12571 | FLAGFPPS02:
12572 | P0, BUSDIN_EMIT-[I], |
12573 | P3-T, PS(7-4)_D(7-4)-[I], |LOAD PS<7:4>H FROM D<7:4>H
12574 | NEXT, J/FLAGFPPS03 |
(7075) DCS(0.00,0.0,0.0,0) BM(0000..00.00..00.01..100..000...1.0,0..0..0..1.1011...0..0000,0...11,000...000,111,110)

12575 |
12576 | 7076: 1(FREE)
12577 | FLAGFPPS03:
12578 | P3-T, D_CSPD(D05), D(C)-0, |GET VALUE TO LOAD TO FLAGS<8:0>H,
12579 | NEXT, J/FLAGFPPS04 |EXFLAG<2:1>H
(7076) DCS(0.00,0.0,0.0,0) BM(1010..10.00..00.00..000..000...1.1,0..0..0..0.1010...0..0000,0...11,000...000,111,111)

12580 |
12581 | 7077: 1(FREE)
12582 | FLAGFPPS04:
12583 | P0, BUSDIN_EMIT-[I], |KEEP IT ON
12584 | P3-T, FLAG(8-0)_D(15-8)-[I], |LOAD FLAGS FROM D
12585 | NEXT, BUTA(RETURN), |AND RETURN
12586 | J/BUTERROR7 |
(7077) DCS(0.00,0.0,0.0,0) BM(0000..00.00..00.01..000..001...1.0,0..0..0..1.1011...0..0000,0...11,111...011,111,110)

12587 |
12588 | THIS SECOND SUBROUTINE COPIES:
12589 | CSP(08) -> PS<15:12>
12590 | CSP(06) -> PS<7:4>
12591 | CSP(07) -> PS<3:0>
12592 | THEN RETURNS
12593 |
12594 |
12595 | 7100: 1(FREE)
12596 | PSSEQL0D1
12597 | P3-T, D_CSPD(D05), D(C)-0, |GET VALUE TO LOAD PS<15:12>H
12598 | NEXT, J/PSSEQL0D02 |
(7100) DCS(0.00,0.0,0.0,0) BM(1010..10.00..00.00..000..000...1.1,0..0..0..0.1010...0..0000,0...11,000...001,000,001)

12599 |
12600 | 7101: 1(FREE)
12601 | PSSEQL0D02:
12602 | P0, BUSDIN_EMIT-[I], |
12603 | P3-T, PS(15-12)_D(15:13)-[I], |LOAD PS<15-12>H FROM D<15,13>H
12604 | NEXT, J/PSSEQL0D03 |

```

```

(7101) DCS(0.00,0.0,0.0,0) BM(0000..00.00..00.01..000..010...1.0,0..0..0..1.1011...0..0000,0...11,000...001,000,010)

12605 |
12606 | 7102: 1(FREE)
12607 | PSSEQL0D03:
12608 | P3-T, D_CSPD(D06), D(C)-0, |GET VALUE TO LOAD PS<7:4>H
12609 | NEXT, J/PSSEQL0D04 |
(7102) DCS(0.00,0.0,0.0,0) BM(1010..10.00..00.00..000..000...1.1,0..0..0..0.1001...0..0000,0...11,000...001,000,011)

12610 |
12611 | 7103: 1(FREE)
12612 | PSSEQL0D04:
12613 | P0, BUSDIN_EMIT-[I], |
12614 | P3-T, PS(7-4)_D(7-4)-[I], |LOAD PS<7:4>H FROM D<7:4>H
12615 | NEXT, J/PSSEQL0D05 |
(7103) DCS(0.00,0.0,0.0,0) BM(0000..00.00..00.01..010..000...1.0,0..0..0..1.1011...0..0000,0...11,000...001,000,100)

12616 |
12617 | 7104: 1(FREE)
12618 | PSSEQL0D05:
12619 | P3-T, D_CSPD(D07), D(C)-0, |GET VALUE TO LOAD TO PS<3:0>H
12620 | NEXT, J/PSSEQL0D06 |
(7104) DCS(0.00,0.0,0.0,0) BM(1010..10.00..00.00..000..000...1.1,0..0..0..0.1000...0..0000,0...11,000...001,000,101)

12621 |
12622 | 7105: 1(FREE)
12623 | PSSEQL0D06:
12624 | P0, BUSDIN_EMIT-[I], |KEEP IT ON
12625 | P2-T, PS(3-0)_D(3-0)-[I], |LOAD PS<3:0>H FROM D<3:0>H
12626 | NEXT, BUTA(RETURN), |AND RETURN
12627 | J/BUTERROR7 |
(7105) DCS(0.00,0.0,0.0,0) BM(1000..00.00..00.01..000..000...0.0,0..0..0..0.1011...0..0000,0...11,111...011,111,110)

12628 |
12629 |
12630 |
12631 |
12632 |
12633 | 1.PAGE=====
12634 |
12635 |
12636 | .T0C * TF8T511: MF88 LOGIC TESTS
12637 |
12638 |
12639 | 1=====
12640 | 1*
12641 | 1* TESTS: 511 A - B UNWORDS: 022 + 060
12642 | 1*
12643 | 1* FUNCTIONS:
12644 | 1*
12645 | 1* THE FOLLOWING TESTS VERIFY THAT THE "MF SAME STACK" LOGIC OPERATES
12646 | 1* CORRECTLY, AND THAT THE "SR6-H" DECODE IS CORRECT.
12647 | 1*
12648 | 1=====
12649 |
12650 |
12651 |
12652 |
12653 | SUMMARY OF "MF SAME STACK H" LOGIC TESTS

```

```
12654 1
12655 1 MF SAME STACK H = FLAG2-H * IR8-H * IR7-H * IR6-L * (PS15-H*PS13-H)-H
12656 1
12657 1 TEST PS15,13H FLAG2H IR8-6H/IR6H MFSS-H DATA
12658 1 -----
12659 1
12660 1 A1 0-0 1 110-1 1 002801
12661 1
12662 1 A2 0-1 1 110-1 0 022600
12663 1
12664 1 A3 1-1 1 110-1 1 123601
12665 1
12666 1 A4 1-0 1 110-1 0 103600
12667 1
12668 1
12669 1 B1 0-0 0 110-1 0 000600
12670 1
12671 1 B2 0-0 1 010-0 0 002200
12672 1
12673 1 B3 0-0 1 100-0 0 002400
12674 1
12675 1 B4 0-0 1 111-0 0 002700
12676 1
12677 1
12678 1
12679 1 TESTING SUBR USED FOR ABOVE TESTS USES THE DATA AS FOLLOWS:
12680 1
12681 1 DATA<15,13> => PS<15,13>, DATA<10> => FLAG<2>,
12682 1 DATA<8,6> => IR<8,6>, DATA<0> = EXPECTED MFSS-H OUTPUT
12683 1
12684 1
12685 1 62321 1(FREE)
12686 1 MFSS011
12687 1 P2=U, IR_DBUF-[I], IDONT CARE ABOUT AFFECT, ONLY SET
12688 1 P3, DBUF_D-[I], THESE UCONS UP
12689 1 NEXT, J/MFSS02
12690 1 (6232) DCS[0,00,0,0,0,0] BM[0100..00,00..00,01..0000..100..0,0,0,0,0,0..1,1011..0..0000,0..11,000..011,000,010]
12691 1
12692 1 63021 1(FREE)
12693 1 MFSS021
12694 1 P2=T, D_CSP8(817), D(C)_ALU00, IGET DATA INTO D, D(C)_EXPECTED MFSS-H
12695 1 P2=U, IR_DBUF, IGNORE FOR NOW
12696 1 P3, DBUF_D, IGET DATA INTO DBUF, TO GO TO IR NEXT
12697 1 NEXT, J/MFSS03
12698 1 (6302) DCS[0,00,0,0,0,0] BM[1010..11,00..00,00..0000..010..0,1,0,0,0,0..1,1010..0..0000,0..11,000..011,000,011]
12699 1
12700 1 63031 1(FREE)
12701 1 MFSS031
12702 1 P0, BUMP-VERIFY, ICOUNT
12703 1 P2=U, IR_DBUF, ISETUP IR<8,6> FROM DATA<8,6>
12704 1 P3, DBUF_D, IGNORE FOR NOW
12705 1 NEXT, J/MFSS04
12706 1 (6303) DCS[0,00,0,0,0,0] BM[0000..00,00..00,00..0000..000..0,0,0,0,0,0..1,1010..0..0000,0..11,000..011,000,100]
12707 1
12708 1
12709 1
12710 1
12711 1
```

```
12705 1 63041 1(FREE)
12706 1 MFSS041
12707 1 P0, BUSDIN_EMIT-[I], IRESET
12708 1 P3, FLAG[8-0]_D[15-8]-[I], ISETUP FLAG<2> FROM DATA<10>
12709 1 PS[15-12]_D[15-13]-[I], ISETUP PS<15,13> FROM D<15,13>
12710 1 NEXT, BUTR(D[C]-A), IIF EXPECT MFSS-H(1), J/MFSS06
12711 1 J/MFSS05 IIF EXPECT MFSS-H(0), J/MFSS05
12712 1 (6304) DCS[0,00,0,0,0,0] BM[0000..00,00..00,01..0000..011..0,0,0,0,0,0..1,1011..0..0000,0..01,111..101,101,010]
12713 1 I* COME HERE IF EXPECT MFSS-H=(0)
12714 1 65521
12715 1 MFSS051
12716 1 P0, LOAD-ENUA(MFSS05EXPECO), ISETUP FOR (0)
12717 1 LOAD-ERROR(MFSS05), IERROR DIRECTORY KEY
12718 1 DCS-CTR(C1.), ICOMPARE AT TARGET
12719 1 BUMP-VERIFY, ICOUNT
12720 1 NEXT, BUTR(MFSS), ITEST MFSS: (0)=MFSS05EXPECO, (1)=MFSS05EXPEC1
12721 1 J/MFSS05EXPECO
12722 1 (6552) DCS[1,00,1,0,0,0] BM[1110..00,11..01,01..100..101..0,0,0,0,0,0..0,0000..0..0000,0..01,100..101,100,101]
12723 1
12724 1 I* COME HERE IF EXPECT MFSS-H=(1)
12725 1 65531
12726 1 MFSS061
12727 1 P1, LOAD-ENUA(MFSS06EXPEC1), ISETUP FOR (0)
12728 1 LOAD-ERROR(MFSS06), IERROR DIRECTORY KEY
12729 1 DCS-CTR(C1.), ICOMPARE AT TARGET
12730 1 NEXT, BUTR(MFSS), ITEST MFSS: (0)=MFSS06EXPECO, (1)=MFSS06EXPEC1
12731 1 J/MFSS06EXPEC1
12732 1 (6553) DCS[1,00,1,0,0,0] BM[1110..00,11..01,01..100..111..0,0,0,0,0,0..0,0000..0..0000,0..01,100..101,100,101]
12733 1
12734 1 I* COME HERE IF MFSS-H TESTS AS A (0)
12735 1 65451
12736 1 MFSS06EXPEC1
12737 1 P0, BUSDIN_EMIT-[I], IRESET PROC UCOS
12738 1 NEXT, BUTA(RETURN), IAND RETURN
12739 1 J/BUTERROR6 I*** COMPARE DONE HERE ***
12740 1 (6545) DCS[0,00,0,0,0,0] BM[0000..00,00..00,01..0000..000..0,0,0,0,0,0..1,1001..0..0000,0..11,111..011,111,110]
12741 1
12742 1 I* COME HERE IF MFSS-H TESTS AS A (1)
12743 1 65471
12744 1 MFSS06EXPEC1
12745 1 P0, BUSDIN_EMIT-[I], IRESET PROC UCOS
12746 1 NEXT, BUTA(RETURN), IAND RETURN
12747 1 J/BUTERROR6 I*** COMPARE DONE HERE ***
12748 1 (6547) DCS[0,00,0,0,0,0] BM[0000..00,00..00,01..0000..000..0,0,0,0,0,0..1,1001..0..0000,0..11,111..011,111,110]
12749 1
12750 1
12751 1
12752 1
12753 1
```

```
12754 6607:
12755 TESTS11A1:
12756 P0, LOAD-ERROR(TESTS11A), ERROR DIRECTORY KEY
12757 P3, CSPD[17]_EMIT, EMIT/002601, I* TEST 511 A1 DATA *
12758 NEXT, J/GOTESTS11A1
(6607) DCS[0.00,0.0,0.0] BM[0000..10.01..01.10..000..001...0.0,0..0..0..0.0000...1..0000,0...11.000...101.101.000]

12759
12760 6550:
12761 GOTESTS11A1:
12762 SETUP, RETURN/TESTS11A2, IGO DO THE TEST
12763 NEXT, CALL[MFSS-TEST]
(6550) DCS[0.00,0.0,0.0] BM[0111..00.00..01.11..100..110...0.0,0..0..0..0.0000...0..0000,0...11.100...010.011.010]

12764
12765 7074: 1(FREE)
12766 TESTS11A2:
12767 P3, CSPD[17]_EMIT, EMIT/022600, I* TEST 511 A2 DATA *
12768 NEXT, J/GOTESTS11A2
(7074) DCS[0.00,0.0,0.0] BM[0010..10.01..01.10..000..000...0.0,0..0..0..0.0000...1..0000,0...11.000...001.000.111]

12770
12771 7107: 1(FREE)
12772 GOTESTS11A2:
12773 SETUP, RETURN/TESTS11A3, IGO DO THE TEST
12774 NEXT, CALL[MFSS-TEST]
(7107) DCS[0.00,0.0,0.0] BM[0111..00.00..10.01..000..110...0.0,0..0..0..0.0000...0..0000,0...11.100...010.011.010]

12775
12776 7110: 1(FREE)
12777 TESTS11A3:
12778 P3, CSPD[17]_EMIT, EMIT/122601, I* TEST 511 A3 DATA *
12779 NEXT, J/GOTESTS11A3
(7110) DCS[0.00,0.0,0.0] BM[1010..10.01..01.10..000..001...0.0,0..0..0..0.0000...1..0000,0...11.000...001.001.001]

12781
12782 7111: 1(FREE)
12783 GOTESTS11A3:
12784 SETUP, RETURN/TESTS11A4, IGO DO THE TEST
12785 NEXT, CALL[MFSS-TEST]
(7111) DCS[0.00,0.0,0.0] BM[0111..00.00..10.01..010..110...0.0,0..0..0..0.0000...0..0000,0...11.100...010.011.010]

12786
12787 7112: 1(FREE)
12788 TESTS11A4:
12789 P3, CSPD[17]_EMIT, EMIT/102600, I* TEST 511 A4 DATA *
12790 NEXT, J/GOTESTS11A4
(7112) DCS[0.00,0.0,0.0] BM[1000..10.01..01.10..000..000...0.0,0..0..0..0.0000...1..0000,0...11.000...001.001.011]

12792
12793 7113: 1(FREE)
12794 GOTESTS11A4:
12795 SETUP, RETURN/SCOPE511A, IGO DO THE TEST
12796 NEXT, CALL[MFSS-TEST]
(7113) DCS[0.00,0.0,0.0] BM[0110..00.01..10.00..001..110...0.0,0..0..0..0.0000...0..0000,0...11.100...010.011.010]
```

```
12797
12798
12799 6301: 1(FREE)
12800 SCOPE511A:
12801 P3, CSPD[17]_EMIT, EMIT/002601, I* RESET DATA FOR TEST 511 A1
12802 NEXT, AUTO[SCOPE], I* NO ERRORS "TESTS11B1" (+1, WORDS)
12803 J/TESTS11B1 I* ERRORS "GOTESTS11A1" (-7, WORDS)
(6301) DCS[0.00,0.1,0.0] BM[0000..10.01..01.10..000..001...0.0,0..0..0..0.0000...1..0000,0...11.000...101.101.001]

12804
12805
12806
12807
12808
12809
12810
12811 6551:
12812 TESTS11B1:
12813 P3, CSPD[17]_EMIT, EMIT/000600, I* TEST 511 B1 DATA *
12814 NEXT, J/GOTESTS11B1
(6551) DCS[0.00,0.0,0.0] BM[0000..10.00..01.10..000..000...0.0,0..0..0..0.0000...1..0000,0...11.000...101.101.100]

12815
12816 6554:
12817 GOTESTS11B1:
12818 SETUP, RETURN/TESTS11B2, IGO DO THE TEST
12819 NEXT, CALL[MFSS-TEST]
(6554) DCS[0.00,0.0,0.0] BM[0111..00.00..10.00..110..110...0.0,0..0..0..0.0000...0..0000,0...11.100...010.011.010]

12820
12821 7106: 1(FREE)
12822 TESTS11B2:
12823 P3, CSPD[17]_EMIT, EMIT/002200, I* TEST 511 B2 DATA *
12824 NEXT, J/GOTESTS11B2
(7106) DCS[0.00,0.0,0.0] BM[0000..10.01..00.10..000..000...0.0,0..0..0..0.0000...1..0000,0...11.000...001.001.101]

12826
12827 7115: 1(FREE)
12828 GOTESTS11B2:
12829 SETUP, RETURN/TESTS11B3, IGO DO THE TEST
12830 NEXT, CALL[MFSS-TEST]
(7115) DCS[0.00,0.0,0.0] BM[0111..00.00..10.01..110..110...0.0,0..0..0..0.0000...0..0000,0...11.100...010.011.010]

12831
12832 7116: 1(FREE)
12833 TESTS11B3:
12834 P3, CSPD[17]_EMIT, EMIT/002400, I* TEST 511 B3 DATA *
12835 NEXT, J/GOTESTS11B3
(7116) DCS[0.00,0.0,0.0] BM[0000..10.01..01.00..000..000...0.0,0..0..0..0.0000...1..0000,0...11.000...001.001.111]

12837
12838 7117: 1(FREE)
12839 GOTESTS11B3:
12840 SETUP, RETURN/TESTS11B4, IGO DO THE TEST
12841 NEXT, CALL[MFSS-TEST]
(7117) DCS[0.00,0.0,0.0] BM[0111..00.00..10.10..000..110...0.0,0..0..0..0.0000...0..0000,0...11.100...010.011.010]
```

```

12842
12843
12844 71201 1(FREE)
12845 TESTS11B4:
12846 P3, CSPD[17]_EMIT, EMIT/002700, 1* TEST 511 B4 DATA *
12847 NEXT, J/GOTESTS11B4
(7120) DCS[0,00,0,0,0,0] BM[0000..10,01..01,11..000,000...0,0,0,0,0...0,0000...1..0000,0...11,000...001,010,001]

12848
12849 71211 1(FREE)
12850 GOTESTS11B4:
12851 SETUP, RETURN/SCOPES11B, 1GO DO THE TEST
12852 NEXT, CALL[MFSS-TEST]
(7121) DCS[0,00,0,0,0,0] BM[0110..00,01..10,00..101..110...0,0,0,0,0...0,0000...0..0000,0...11,100...010,011,010]

12853
12854
12855 6305: 1(FREE)
12856 SCOPES11B:
12857 P3, CSPD[17]_EMIT, EMIT/000600, 1RESET DATA FOR TEST 511 B1
12858 NEXT, BUTD[SCOPE], 1NO ERROR; "SETUPS12A" (+1, WORDS)
12859 J/SETUPS12A 1 ERROR; "GOTESTS11B1" (-7, WORDS)
(6305) DCS[0,00,0,1,0,0] BM[0000..10,00..01,10..000,000...0,0,0,0,0...0,0000...1..0000,0...11,000...101,101,101]

12860
12861
12862
12863
12864
12865 1, PAGE=====
12866
12867
12868 .TOC * TESTS12: KT SRC/DST ADDRESSING LOGIC TESTS
12869
12870
12871
12872 1=====
12873 1*
12874 1* TESTS: 512 A - E UNWORDS: 044 + 062
12875 1*
12876 1* FUNCTIONS:
12877 1*
12878 1* THE FOLLOWING TEN TESTS VERIFY THE KT-SRC/DST-ADDRS ROM OUTPUT AND INPUT LINES
12879 1* FUNCTION CORRECTLY, IN RESPECT TO NO STUCK ONE/ZERO CONDITIONS.
12880 1*
12881 1=====
12882
12883
12884 1 KT SRC/DST LOGIC EQUATIONS: (IMPLEMENTED IN ROM)
12885 1
12886 1 KT-SRC-ADDRS-3 = NOT-F2,AND,SR6,AND,PS15
12887 1 .OR. F2,AND,SR6,AND,NOT-SM0,AND,PS15
12888 1 .OR. F2,AND,SR6,AND,SM0,AND,PS13,AND,NOT-FLTPT
12889 1 .OR. SR6,AND,PS15,AND,FLTPT
12890 1
12891 1 KT-DST-ADDRS-3 = NOT-F1,AND,DR6,AND,PS15

```

```

12892 1 .OR. F1,AND,DR6,AND,NOT-DM0,AND,PS15
12893 1 .OR. F1,AND,DR6,AND,DM0,AND,PS13,AND,NOT-FLTPT
12894 1 .OR. DR6,AND,PS15,AND,FLTPT
12895 1
12896 1
12897 1 SUMMARY OF KT ASP/BSP SRC/DST STACK POINTER ADDRESSING LOGIC:
12898 1
12899 1
12900 1
12901 1
12902 1
12903 1
12904 1
12905 1
12906 1
12907 1
12908 1
12909 1
12910 1
12911 1
12912 1
12913 1
12914 1
12915 1
12916 1
12917 1
12918 1
12919 1
12920 1
12921 1
12922 1
12923 1
12924 1
12925 1
12926 1
12927 1
12928 1
12929 1
12930 1
12931 1
12932 1
12933 1
12934 1
12935 1
12936 1
12937 1
12938 1
12939 1
12940 1
12941 1
12942 1
12943 1

```

TEST NUMB	PS	FLAG	IR	FLTL	SM0H	SR6H	DM0H	DR6H	KT-SRC ADR-H	KT-DST ADR-H	HOW READ:
A1	1,0	1,1	172206	0	0	0	1	1	0	1	ASPHI[SF]=(02) BSPHI[DF]=(16)
A2											
B1	0,1	1,0	070606	1	1	1	1	1	1	0	ASPHI[DF]=(06) BSPHI[SF]=(16)
B2											
C1	0,1	1,1	134606	1	0	1	1	1	0	1	ASPHI[DF]=(16) BSPHI[SF]=(06)
C2											
D1	1,0	0,1	160612	1	1	1	0	0	1	0	ASPHI[SF]=(16) BSPHI[DF]=(02)
D2											
E1	1,0	1,1	150626	1	1	1	0	1	0	1	ASPHI[DF]=(16) BSPHI[SF]=(06)
E2											

```

12922 1
12923 1
12924 1
12925 1
12926 1
12927 1
12928 1
12929 1
12930 1
12931 1
12932 1
12933 1
12934 1
12935 1
12936 1
12937 1
12938 1
12939 1
12940 1
12941 1
12942 1
12943 1

```

```

12944 KTSRCDST03;
12945 P2-U, IR_DBUF,          !COPY IR DATA FROM DBUF -> IR
12946 P3-T, D_CSPB(816), D(C)_ALU00, !GET PS/FLAGS/REGISTER SELECT DATA
12947 P3, DBUF_D,          !D(C) = REGISTER SELECT FLAG
12948 NEXT, J/KTSRCDST04
(7124) DCS[0.00.0.0.0.0] BM[1010..11.01..00.00..000..010...1.1.0..0.0...1.1010...0..0000.0...11.000...001.010.101]
12949
12950 7125: 1(FREE)
12951 KTSRCDST04;
12952 P0, BUSDIN_EMIT-[I], !KEEP IT ON
12953 P3, PS[15-12]_D[15:13]-[I], !SETUP PS<15,13> AS REQUIRED
12954 NEXT, FLAG[8-0]_D[15-8]-[I], !SETUP FLAG<2:1> AS REQUIRED
12955 J/KTSRCDST04B
(7125) DCS[0.00.0.0.0.0] BM[0000..00.00..00.01..000..011...0.0.0..0.0...1.1011...0..0000.0...11.000...001.010.110]
12956
12957 7126: 1(FREE)
12958 KTSRCDST04B;
12959 !THIS WORD NEEDED FOR 1 UWORD DELAY FOR ROM TO SETTLE
12960 P2-T, D_CSPD(C052525), SAVE-D(C), !DATA PATTERN IN SCRATCH PAD ADDRESS [02]
12961 P3, A8BSPHI[02]_D, !BIT<3:0> = (05)
12962 NEXT, BUTR(D(C)-B), !KEY<1> -> READ ASPHI[DF] -> SP
12963 J/KTSRCDST05 !KEY<0> -> READ ASPHI[SF] -> SP
(7126) DCS[0.00.0.0.0.0] BM[1010..10.10..00.00..101..111...0.1.0..0.0...0.111...0..1011.0...10.011...011.011.101]
12964 !ENTER HERE IF D(C) CLEAR, SF SELECTED
12965 7335:
12966 KTSRCDST05;
12967 P2-T, SR_ASPHI[SF], !USE ASP/SF
12968 NEXT, J/BUTSR3=0, !AND NOW GO CHECK WHOM WAS READ
12969 (7335) DCS[0.00.0.0.0.0] BM[1111..00.00..11.11..000..000...0.0.1..0.0...0.0000...0..0000.0...11.000...010.111.110]
12970 !ENTER HERE IF D(C) SET, DF SELECTED
12971 7337:
12972 KTSRCDST06;
12973 P2-T, SR_ASPHI[DF], !USE ASP/DF
12974 NEXT, J/BUTSR3=0, !AND NOW GO CHECK WHOM WAS READ
12975 (7337) DCS[0.00.0.0.0.0] BM[1111..00.00..11.10..000..000...0.0.1..0.0...0.0000...0..0000.0...11.000...010.111.110]
12976
12977 !* WE ALSO NEEDED TWO ENTRY POINTS TO READ BSP SF & DF;
12978
12979 7127: 1(FREE)
12980 KTSRCDST07;
12981 P2-T, SR_BSPHI[SF], !USE BSP/SF
12982 NEXT, J/BUTSR3=0, !AND NOW GO CHECK WHOM WAS READ
12983 (7127) DCS[0.00.0.0.0.0] BM[1010..01.01..00.00..000..000...0.0.1..0.0...0.0000...0..0000.0...11.000...010.111.110]
12984
12985 7130: 1(FREE)
12986 KTSRCDST08;
12987 P2-T, SR_BSPHI[DF], !USE BSP/DF
12988 NEXT, J/BUTSR3=0, !AND NOW GO CHECK WHOM WAS READ
12989 (7130) DCS[0.00.0.0.0.0] BM[1010..01.00..00.00..000..000...0.0.1..0.0...0.0000...0..0000.0...11.000...010.111.110]
12990

```

```

12991
12992 | - - - - -
12993
12994 | *** KI SRC/DST ENTERS HERE ***
12995
12996 | - - - - -
12997
12998
12999 !THESE FIRST TWO WORDS DO SOME PRELIMINAR SETUP OF SCRATCHPAD LOCATIONS (06)/(16)
13000 6545:
13001 SETUP512A;
13002 P2-T, D_CSPD(C125252), D(C)_0, !
13003 P3, A8BSPHI[06]_D, !KENNAL SP LOCATION;
13004 NEXT, J/SETUP512B ! BIT<3:0> = (12) = SP-ADDRESS(06)
(6555) DCS[0.00.0.0.0.0] BM[1010..10.10..00.00..111..000...0.1.0..0.0...0.0110...0..1011.0...11.000...011.000.111]
13005
13006 6307: 1(FREE)
13007 SETUP512B;
13008 P2-T, D_CSPD(C000000), D(C)_0, !
13009 P3, A8BSPHI[16]_D, !USER SP LOCATION;
13010 NEXT, J/TEST512A ! BIT<3:0> = (00) = SP-ADDRESS(16)
(6307) DCS[0.00.0.0.0.0] BM[1010..10.10..00.00..011..000...0.1.0..0.0...0.0100...0..1011.0...11.000...101.100.010]
13011
13012 !A,BSPHI[02] = (052525)
13013 ! BIT<3:0> = (05) = SP-ADDRESS(02)
13014
13015 !A,BSPHI[12] = (000152)
13016 ! BIT<3:0> = (12) = SP-ADDRESS(12)
13017 ! (NOT USED UNLESS ERROR)
13018
13019
13020 | - - - - -
13021
13022 !* TEST 512 A1 *
13023
13024 | TEST 512 A 1-2 SETS UP FOR: ASP-SF-ADDRESS=(02), BSP-DF-ADDRESS=(16)
13025 6542:
13026 TEST512A1;
13027 P0, LOAD-ENUA(2TARGET405), !BIT<3:0> = (05) = SP-ADDRESS(02)
13028 P3, LOAD-ERROR(TEST512A1), !ERROR DIRECTORY KEY
13029 NEXT, DCS-CTR(C11.), !COMPARE AT TARGET
13030 J/SETIR512A1 !
(6542) DCS[1.00.1.0.0.0] BM[10100..00.11..11.00..000..101...0.0.0..0.0...0.0000...0..0000.0...11.000...101.110.100]
13031
13032 6564:
13033 SETIP512A1;
13034 P0, RUMP-VERIFY, !COUNT
13035 P3, CSPD[17]_EMIT, EMIT/172206, !SETUP IR: FLTP/SMO/SR6/DNO/DP6
13036 NEXT, J/SETPSFLAG512A1 !
(6564) DCS[0.00.0.0.0.1] BM[1111..10.01..00.10..000..110...0.0.0..0.0...0.0000...1..0000.0...11.000...011.001.000]
13037
13038 6310: 1(FREE)
13039 SETPSFLAG512A1;
13040 P3, CSPD[16]_EMIT, EMIT/103000, !RIT<15,13> -> PS<15,13>,

```

```
13041 NFXT, J/GOTESTS12A1 ;BIT<10:9> -> FLAGS<2:1>
13042 (6310) DCS[0.00.0.0.0.0] BM[1000..10.01..10.00..0000..0000..0.0.0.0..0.0001...1..0000.0...11.000...011.001.001] ;BIT<00> IS REGISTER KEY (SEE SUBR)
13043 6311: 1(FREE)
13044 GOTESTS12A1:
13045 SETUP, RETURN/TESTS12A2, ;GO EXEC RT SRC/DST TEST SUBR AT
13046 NEXT, CALL[KTSCDST] ; INITIALISATION POINT
13047 (6311) DCS[0.00.0.0.0.0] BM[1010..00.10..11.00..100..111...0.0.0.0..0.0000...0..0000.0...11.100...001.001.100]
13048
13049
13050
13051 ;* TEST 512A2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TESTS12A1, INTO THE SR
13052 6544:
13053 TESTS12A2:
13054 PO, LOAD=ENUA(ZTARGET400), ;BIT<3:0> = (00) = SP-ADDRESS(16)
13055 LOAD=ERROR(TESTS12A2), ;ERROR DIRECTORY KEY
13056 DCS=CTR(C4.), ;COMPARE AT TARGET
13057 NEXT, J/GOTESTS12A2
13058 (6544) DCS[1.00.1.0.0.0] BM[1011..00.11..11.00..0000..0000...0.0.0.0..0.0000...0..0000.0...11.000...011.001.010]
13059 6312: 1(FREE)
13060 GOTESTS12A2:
13061 SETUP, RETURN/SCOPES12A, ;READ REGISTER BSMPI[DF] TO SR,
13062 NEXT, CALL[KTDBSP] ; THEN DO BUT(SR3=0)
13063 (6312) DCS[0.00.0.0.0.0] BM[0110..00.01..10.01..011..111...0.0.0.0..0.0000...0..0000.0...11.100...001.011.000]
13064
13065 6313: 1(FREE)
13066 SCOPES12A:
13067 PO, BUSDIN_EMIT-[1], ;RESET PROC UCON
13068 EN=CLR-IR[15=00],
13069 BUTD[SCOPE], ;NO ERROR: "TESTS12B1" (+1, WORDS)
13070 J/TESTS12B1 ; ERROR: "SETIRS12A1" (-5, WORDS)
13071 (6313) DCS[0.00.0.1.0.0] BM[0000..00.00..00.01..0000..100...0.0.0.0..0.0000...0..0000.0...11.000...101.110.101]
13072
13073
13074
13075 1 - - - - -
13076
13077 ;* TEST 512 B1 *
13078
13079 1 TEST 512 B 1-2 SETS UP FOR: ASP=DF-ADDRESS=(06), BSP=SF-ADDRESS=(16)
13080 6565:
13081 TESTS12B1:
13082 PO, LOAD=ENUA(ZTARGET412), ;BIT<3:0> = (12) = SP-ADDRESS(06)
13083 LOAD=ERROR(TESTS12B1), ;ERROR DIRECTORY KEY
13084 DCS=CTR(C11.), ;COMPARE AT TARGET
13085 NEXT, J/SETIRS12B1
13086 (6565) DCS[1.00.1.0.0.0] BM[0100..00.11..11.00..001..010...0.0.0.0..0.0000...0..0000.0...11.000...101.011.110]
```

```
13087 6516:
13088 SETIRS12B1:
13089 PO, BUMP-VERIFY, ;COUNT
13090 P3, CSPD[17]_EMIT, EMIT/070606, ;SETUP IR: FLTPT/SMO/SR6/DMO/DR6
13091 NFXT, J/SETPSFLAG512B1 ;
13092 (6536) DCS[0.00.0.0.0.0] BM[0111..10.00..01.10..0000..110...0.0.0.0..0.0000...1..0000.0...11.000...011.001.100]
13093 6314: 1(FREE)
13094 SFTPSFLAG512B1:
13095 P3, CSPD[16]_EMIT, EMIT/022001, ;BIT<15,13> -> PS<15,13>,
13096 ;BIT<10:9> -> FLAGS<2:1>
13097 NEXT, J/GOTESTS12B1 ;BIT<00> IS REGISTER KEY (SEE SUBR)
13098 (6314) DCS[0.00.0.0.0.0] BM[0010..10.01..00.00..0000..001...0.0.0.0..0.0001...1..0000.0...11.000...011.001.101]
13099 6315: 1(FREE)
13100 GOTESTS12B1:
13101 SETUP, RETURN/TESTS12B2, ;GO EXEC RT SRC/DST TEST SUBR AT
13102 NEXT, CALL[KTSCDST] ; INITIALISATION POINT
13103 (6315) DCS[0.00.0.0.0.0] BM[0110..00.10..11.00..110..111...0.0.0.0..0.0000...0..0000.0...11.100...001.001.100]
13104
13105
13106 ;* TEST 512B2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TESTS12B1, INTO THE SR
13107 6546:
13108 TESTS12B2:
13109 PO, LOAD=ENUA(ZTARGET400), ;BIT<3:0> = (00) = SP-ADDRESS(16)
13110 LOAD=ERROR(TESTS12B2), ;ERROR DIRECTORY KEY
13111 DCS=CTR(C4.), ;COMPARE AT TARGET
13112 NEXT, J/GOTESTS12B2
13113 (6546) DCS[1.00.1.0.0.0] BM[1011..00.11..11.00..0000..0000...0.0.0.0..0.0000...0..0000.0...11.000...011.001.110]
13114 6316: 1(FREE)
13115 GOTESTS12B2:
13116 SFTUP, RETURN/TESTS12C1, ;READ REGISTER BSMPI[SF] TO SR,
13117 NEXT, CALL[KTSCDST] ; THEN DO BUT(SR3=0)
13118 (6316) DCS[0.00.0.0.0.0] BM[0110..00.10..11.01..110..111...0.0.0.0..0.0000...0..0000.0...11.100...001.010.111]
13119
13120
13121
13122
13123 1 - - - - -
13124
13125 ;* TEST 512 C1 *
13126
13127
13128 1 TEST 512 C 1-2 SETS UP FOR: ASP=DF-ADDRESS=(16), BSP=SF-ADDRESS=(06)
13129 6556:
13130 TESTS12C1:
13131 PO, LOAD=ENUA(ZTARGET400), ;BIT<3:0> = (00) = SP-ADDRESS(16)
13132 LOAD=ERROR(TESTS12C1), ;ERROR DIRECTORY KEY
13133 DCS=CTR(C11.), ;COMPARE AT TARGET
```

```
13134 NEXT, J/SETIRS12C1
(6556) DCS[1.00,1.0,0.0] BM[0100..00.11..11.00..000..000..0.0,0.0,0.0..0.0000..0.0000.0..11.000..011.001.111]
13135
13136 6317: 1(FREE)
13137 SETIRS12C1:
13138 PO, BUMP-VERIFY, ICOUNT
13139 P3, CSPD[17]_EMIT, EMIT/134606, ISETUP IR: FLTP/SMO/SR6/DWO/DR6
13140 NEXT, J/SETPSFLACS12C1
(6317) DCS[0.00,0.0,0.0] BM[1011..10.10..01.10..000..110..0.0,0.0,0.0..0.0000..1.0000.0..11.000..011.010.000]
13141
13142 6320: 1(FREE)
13143 SETPSFLACS12C1:
13144 P3, CSPD[16]_EMIT, EMIT/023001, IBIT<15,13> -> PS<15,13>,
13145 IBIT<10,9> -> FLAGS<2,1>
13146 NEXT, J/GOTESTS12C1 IBIT<00> IS REGISTER KEY (SEE SUBR)
(6320) DCS[0.00,0.0,0.0] BM[0010..10.01..10.00..000..001..0.0,0.0,0.0..0.0000..1.0000.0..11.000..011.010.001]
13147
13148 6321: 1(FREE)
13149 GOTESTS12C1:
13150 SETUP, RETURN/TESTS12C2, IGO EXEC KT SRC/DST TEST SUBR AT
13151 NEXT, CALL[KT SRC/DST] I INITIALIZATION POINT
(6321) DCS[0.00,0.0,0.0] BM[0110..00.10..11.10..110..111..0.0,0.0,0.0..0.0000..0.0000.0..11.100..001.001.100]
13152
13153
13154
13155 1* TEST 512C2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TESTS12C1, INTO THE SR
13156 6566:
13157 TESTS12C2:
13158 PO, LOAD-ENUA(ZTARGET412), IBIT<3,0> = (12) = SP-ADDRESS(06)
13159 LOAD-ERROR(TESTS12C2), IERROR DIRECTORY KEY
13160 DCS-CTR(C4.), ICOMPARE AT TARGET
13161 NEXT, J/GOTESTS12C2
(6566) DCS[1.00,1.0,0.0] BM[1011..00.11..11.00..001..010..0.0,0.0,0.0..0.0000..0.0000.0..11.000..011.010.010]
13162
13163 6322: 1(FREE)
13164 GOTESTS12C2:
13165 SETUP, RETURN/SCOPE512C, IREAD REGISTER BSPHI(SF) TO SR,
13166 NEXT, CALL[KT SRC/DST] I THEN DO BUT(SR3=0)
(6322) DCS[0.00,0.0,0.0] BM[0110..00.01..10.10..011..111..0.0,0.0,0.0..0.0000..0.0000.0..11.100..001.010.111]
13167
13168
13169 6323: 1(FREE)
13170 SCOPE512C:
13171 PO, BUSDIN-EMIT-[I], IRESET PROC UCON
13172 EN-CLK-IR(15=00), I
13173 NEXT, RUTD(SCOPE), I NO ERROR: "TESTS12D1" (+1. WORDS)
13174 J/TESTS12D1 I ERROR: "SETIRS12B1" (-1. WORDS)
(6323) DCS[0.00,0.1,0.0] BM[0000..00.00..00.01..000..100..0.0,0.0,0.0..1.0001..0.0000.0..11.000..101.011.111]
13175
13176
13177
13178
```

```
13179 1 - - - - -
13180
13181 1* TEST 512 D1 *
13182
13183 1 TEST 512 D 1-2 SETS UP FOR: ASP-SF-ADDRESS=(16), BSP-DP-ADDRESS=(02)
13184 6537:
13185 TESTS12D1:
13186 PO, LOAD-ENUA(ZTARGET400), IBIT<3,0> = (00) = SP-ADDRESS(16)
13187 LOAD-ERROR(TESTS12D1), IERROR DIRECTORY KEY
13188 DCS-CTR(C1.), ICOMPARE AT TARGET
13189 NEXT, J/SETIRS12D1
(6537) DCS[1.00,1.0,0.0] BM[0100..00.11..11.00..000..000..0.0,0.0,0.0..0.0000..0.0000.0..11.000..110.010.110]
13190
13191 6626:
13192 SETIRS12D1:
13193 PO, BUMP-VERIFY, ICOUNT
13194 P3, CSPD[17]_EMIT, EMIT/160612, ISETUP IR: FLTP/SMO/SR6/DWO/DR6
13195 NEXT, J/SETPSFLACS12D1
(6626) DCS[0.00,0.0,0.0] BM[1110..10.00..01.10..001..010..0.0,0.0,0.0..0.0000..1.0000.0..11.000..011.010.100]
13196
13197 6324: 1(FREE)
13198 SETPSFLACS12D1:
13199 P3, CSPD[16]_EMIT, EMIT/101000, IBIT<15,13> -> PS<15,13>,
13200 IBIT<10,9> -> FLAGS<2,1>
13201 NEXT, J/GOTESTS12D1 IBIT<00> IS REGISTER KEY (SEE SUBR)
(6324) DCS[0.00,0.0,0.0] BM[1000..10.00..10.00..000..000..0.0,0.0,0.0..0.0001..1.0000.0..11.000..011.010.101]
13202
13203 6325: 1(FREE)
13204 GOTESTS12D1:
13205 SETUP, RETURN/TESTS12D2, IGO EXEC KT SRC/DST TEST SUBR AT
13206 NEXT, CALL[KT SRC/DST] I INITIALIZATION POINT
(6325) DCS[0.00,0.0,0.0] BM[0110..00.10..11.11..110..111..0.0,0.0,0.0..0.0000..0.0000.0..11.100..001.001.100]
13207
13208
13209
13210 1* TEST 512D2 NOW READS THE COMPLEMENTARY REGISTER TO THAT USED IN TESTS12D1, INTO THE SR
13211 6576:
13212 TESTS12D2:
13213 PO, LOAD-ENUA(ZTARGET405), IBIT<3,0> = (05) = SP-ADDRESS(02)
13214 LOAD-ERROR(TESTS12D2), IERROR DIRECTORY KEY
13215 DCS-CTR(C4.), ICOMPARE AT TARGET
13216 NEXT, J/GOTESTS12D2
(6576) DCS[1.00,1.0,0.0] BM[1011..00.11..11.00..000..101..0.0,0.0,0.0..0.0000..0.0000.0..11.000..011.010.110]
13217
13218 6326: 1(FREE)
13219 GOTESTS12D2:
13220 SETUP, RETURN/TESTS12E1, IREAD REGISTER BSPHI[DF] TO SR,
13221 NEXT, CALL[KT DST/BS] I THEN DO BUT(SR3=0)
(6326) DCS[0.00,0.0,0.0] BM[0110..00.10..11.11..111..111..0.0,0.0,0.0..0.0000..0.0000.0..11.100..001.011.000]
13222
13223
13224
```



```
13324      LOAD=ERROR(TEST520A),      |ERROR DIRECTORY KEY
13325      DCS=CTR(C7.),                |COMPARE ENUA;TNUA AT TARGET
13326      NEXT,                          |
(6627) DCS[1.00,1.0,0.0] BM[0111..00,11..11,00..000..011...0,0,0..0..0..0.0000...0..0000,0...11,000...110,010,100]
13327
13328      6624:
13329      UCON520A:
13330      SELECT, UCON=PROC,              |PROCESSOR UCON;
13331      ENABLE, BUSDIN_EMIT[15=00],      |  EMIT ON BUSDIN
13332      EN=CLK=IR[15=80],                |  AND CLOCKING IR
13333      P0,      BUMP=VERIFY,            |COUNT
13334      SET=UCON=CONTROL,                |WRITE CONTROLS
13335      NEXT, J/SETUP520A
(6624) DCS[0.00,0.0,0.0,1] BM[0000..00,00..00,01..000..100...0,0,0..0..0...1,1001...0..0000,0...11,000...011,011,100]
13336
13337      6334: 1(FREE)
13338      SETUP520A:
13339      P0,      BUMP=VERIFY,            |COUNT
13340      EMITC,   EMIT/101004,            |PS<3;0>H="0100"
13341      P2=U,    IR_EMIT,                |
13342      P3,      CSPD[05]_EMIT,          |IR<15,10;08>H="1010"
13343      NEXT, J/GOTEST520A
(6334) DCS[0.00,0.0,0.0,1] BM[1000..10,00..10,00..000..100...0,0,0..0..0...1,1010...1..0000,0...11,000...011,011,101]
13344
13345      6335: 1(FREE)
13346      GOTEST520A:
13347      SETUP,   RETURN/TEST520B,        |RETURN TO START OF NEXT SUBTEST
13348      NEXT,    GOTO=PAGE(7),           |GO TO CODE THAT LOADS PS[CC], AND
13349      J/SUBC8TEST01                     | THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
(6335) DCS[0.00,0.0,0.0,0] BM[0110..00,11..11,10..100..111...0,0,0..0..0...0,0000...0..0000,0...11,100...001,010,010]
13350
13351
13352
13353
13354      | - - - - -
13355
13356      |*** TEST 520B ***
13357      |TEST-520-B SETS UP IR<15,10;08>H="0110", PS<3;0>H="0000",
13358      | AND THEN BUTS ON "INSTR BRANCH L"
13359      6764:
13360      TEST520B:
13361      P0,      LOAD=ENUA(ZTARGET402),    |ASSERTED
13362      LOAD=ERROR(TEST520B),              |ERROR DIRECTORY KEY
13363      DCS=CTR(C7.),                      |COMPARE ENUA;TNUA AT TARGET
13364      NEXT, J/SETUP520B
(6764) DCS[1.00,1.0,0.0] BM[1000..00,11..11,00..000..010...0,0,0..0..0...0,0000...0..0000,0...11,000...011,011,110]
13365
13366      6336: 1(FREE)
13367      SETUP520B:
13368      FMITC,   EMIT/003000,              |PS<3;0>H="0000"
13369      P2=U,    IR_EMIT,                  |
13370      P3,      CSPD[05]_EMIT,            |IR<15,10;08>H="0110"
13371      NEXT, J/GOTEST520B
```

```
(6336) DCS[0.00,0.0,0.0,0] BM[0000..10,01..10,00..000..000...0,0,0..0..0...1,1010...1..0000,0...11,000...011,011,111]
13372
13373      6337: 1(FREE)
13374      GOTEST520B:
13375      SETUP,   RETURN/TEST520C,        |RETURN TO START OF NEXT SUBTEST
13376      NEXT,    GOTO=PAGE(7),           |GO TO CODE THAT LOADS PS[CC], AND
13377      J/SUBC8TEST01                     | THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
(6337) DCS[0.00,0.0,0.0,0] BM[0110..00,11..11,11..010..111...0,0,0..0..0...0,0000...0..0000,0...11,100...001,010,010]
13378
13379
13380
13381
13382      | - - - - -
13383
13384      |*** TEST 520C ***
13385      |TEST-520-C SETS UP IR<15,10;08>H="0101", PS<3;0>H="1110",
13386      | AND THEN BUTS ON "INSTR BRANCH L"
13387      6772:
13388      TEST520C:
13389      P0,      LOAD=ENUA(ZTARGET403),    |NEGATED
13390      LOAD=ERROR(TEST520C),              |ERROR DIRECTORY KEY
13391      DCS=CTR(C7.),                      |COMPARE ENUA;TNUA AT TARGET
13392      BUMP=VERIFY,                        |COUNT
13393      NEXT, J/SETUP520C
(6772) DCS[1.00,1.0,0.0,1] BM[1000..00,11..11,00..000..011...0,0,0..0..0...0,0000...0..0000,0...11,000...011,100,000]
13394
13395      6340: 1(FREE)
13396      SETUP520C:
13397      EMITC,   EMIT/002416,              |PS<3;0>H="1110"
13398      P2=U,    IR_EMIT,                  |
13399      P3,      CSPD[05]_EMIT,            |IR<15,10;08>H="0101"
13400      NEXT, J/GOTEST520C
(6340) DCS[0.00,0.0,0.0,0] BM[0000..10,01..01,00..001..110...0,0,0..0..0...1,1010...1..0000,0...11,000...011,100,001]
13401
13402      6341: 1(FREE)
13403      GOTEST520C:
13404      SETUP,   RETURN/TEST520D,        |RETURN TO START OF NEXT SUBTEST
13405      NEXT,    GOTO=PAGE(7),           |GO TO CODE THAT LOADS PS[CC], AND
13406      J/SUBC8TEST01                     | THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
(6341) DCS[0.00,0.0,0.0,0] BM[0110..00,11..11,10..010..111...0,0,0..0..0...0,0000...0..0000,0...11,100...001,010,010]
13407
13408
13409
13410
13411      | - - - - -
13412
13413      |*** TEST 520D ***
13414      |TEST-520-D SETS UP IR<15,10;08>H="1011", PS<3;0>H="0001",
13415      | AND THEN BUTS ON "INSTR BRANCH L"
13416      6762:
13417      TEST520D:
13418      P0,      LOAD=ENUA(ZTARGET402),    |ASSERTED
```

```
13419      LOAD=ERROR(TESTS20D),      |ERROR DIRECTORY KEY
13420      DCS=CTR(C7.),                |COMPARE ENUA:THUA AT TARGET
13421      NFXT, J/SETUP520D              |
(6762) DCS[1.00,1.0,0,0] BMT1000..00.11..11.00..000..010...0,0,0..0..0,0000...0..0000,0...11,000...011,100,010|
13422
13423      6342: 1(FREF)
13424      SETUP520D:
13425      EMITC, EMIT/101401,            |PS<3:0>H="0001"
13426      P2=U, IR_EMIT,                |
13427      P3, C8PD[05]_EMIT,            |IR<15,10:08>H="1011"
13428      NEXT, J/GOTESTS20D            |
(6342) DCS[0.00,0,0,0,0] BMT1000..10.00..11.00..000..001...0,0,0..0..0,1.010...1..0000,0...11,000...011,100,011|
13429
13430      6343: 1(FREF)
13431      GOTESTS20D:
13432      SETUP, RETURN/TESTS20E,        |RETURN TO START OF NEXT SUBTEST
13433      NFXT, GOTO=PAGE(7),            |GO TO CODE THAT LOADS PS[CC], AND
13434      J/SUCBRTST01                  | THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
(6343) DCS[0.00,0,0,0,0] BMT0110..00.11..11.11..101..111...0,0,0..0..0,0.0000...0..0000,0...11,100...001,010,010|
13435
13436
13437
13438
13439      1 - - - - -
13440
13441      |*** TEST 520E ***
13442      |TEST-520-E SETS UP IR<15,10:08>H="1111", PS<3:0>H="0110",
13443      | AND THEN BUTS ON "INSTR BRANCH L"
13444
13445      6775:
13446      TESTS20E:
13447      PO, LOAD=ENUA(ZTARGET403),      |NEGATED
13448      DCS=CTR(C7.),                |ERROR DIRECTORY KEY
13449      NEXT, J/SETUP520E              |COMPARE ENUA:THUA AT TARGET
13450      (6775) DCS[1.00,1.0,0,0] BMT1000..00.11..11.00..000..011...0,0,0..0..0,0.0000...0..0000,0...11,000...011,100,100|
13451
13452      6344: 1(FREF)
13453      SETUP520E:
13454      PO, BUMP=VERIFY,              |COUNT
13455      EMITC, EMIT/103406,            |PS<3:0>H="0110"
13456      P2=U, IR_EMIT,                |
13457      P3, C8PD[05]_EMIT,            |IR<15,10:08>H="1111"
13458      NEXT, J/GOTESTS20E            |
(6344) DCS[0.00,0,0,0,1] BMT1000..10.01..11.00..000..110...0,0,0..0..0,1.010...1..0000,0...11,000...011,100,101|
13459
13460      6345: 1(FREF)
13461      GOTESTS20E:
13462      SETUP, RETURN/SCOPE520,        |RETURN TO SCOPE LOOP TEST WORD
13463      NEXT, GOTO=PAGE(7),            |GO TO CODE THAT LOADS PS[CC], AND
13464      J/SUCBRTST01                  | THEN BUTS ON THE "INSTR BRANCH" ROM OUTPUT
(6345) DCS[0.00,0,0,0,0] BMT0110..00.01..11.00..110..111...0,0,0..0..0,0.0000...0..0000,0...11,100...001,010,010|
13464
```

```
13465
13466
13467      6346: 1(FREF)
13468      SCOPE520:
13469      PO, BUSDIN_EMIT-[I],           |RESET PROC UCONS
13470      EN=CLK-IR[15=00],             |
13471      NEXT, BUT0[SCOPE],             |NO ERROR: "TESTS33A" [+4, WORDS]
13472      J/TESTS33A                     | ERROR: "UCONS20A" [-15, WORDS]
13473      (6346) DCS[0.00,0,1,0,0] BMT0000..00.00..00.01..000..100...0,0,0..0..0,1.1001...0..0000,0...11,000...110,010,101|
13474
13475
13476
13477      1 - - - - -
13478
13479      | TESTING SUBR (COMMON CODE) FOR ABOVE TESTS
13480
13481      7122: 1(FREF)
13482      SUCBRTST01:
13483      P2=T, D_C8PD[05], D[IC]_0,      |GET PATTERN
13484      NEXT, J/SUCBRTST02            |
(7122) DCS[0.00,0,0,0,0] BMT1010..10.00..00.00..000..000...0,1,0..0..0,0.1010...0..0000,0...11,000...001,011,010|
13485
13486      7132: 1(FREF)
13487      SUCBRTST02:
13488      P2=T, PS[3=0]_D[3=0]-[I],      |INTO PS[CC]
13489      NEXT, J/SUCBRTST03            |
(7132) DCS[0.00,0,0,0,0] BMT1000..00.00..00.01..000..000...0,0,0..0..0,0.1011...0..0000,0...11,000...001,011,011|
13490
13491      7133: 1(FREF)
13492      SUCBRTST03:
13493      SELECT, UCON=PROC,              |LEAVE WITH EMIT ON BUSDIN,
13494      ENABLE, BUSDIN_EMIT[15=00],    |CLOCK IR ENABLED
13495      EN=CLK-IR[15=00],             |
13496      PO, SET=UCON=CONTROL,          |WRITE CONTROLS
13497      NEXT, J/BUTINSTBRANCH          |
(7133) DCS[0.00,0,0,0,0] BMT0000..00.00..00.01..000..100...0,0,0..0..0,1.1001...0..0000,0...11,000...011,101,110|
13498
13499
13500
13501
13502
13503      |PAGE=====
13504
13505      .TnC * TESTS33-537: SHIFT TREE
13506
13507
13508      |*****
13509      |*
13510      |* TESTS: 533A - 537A                      UNWORDS: 170 + 044
13511      |*
13512      |* FUNCTIONS: TESTS 533A - 537A VERIFY THE DATA AND CONTROL PATHS
13513      |* OF THE 3 LEVEL BARREL SHIFTER (SHIFT TREE),
13514      |*
```

```
13515 1*****
13516
13517
13518
13519
13520
13521
13522 1 - - - - -
13523
13524 1*** TEST 533A ***
13525 IREAD D DIRECTLY THRU "D[H1]D[L0]" PORT OF AMUXENI8L0, BMUX-CMUX/DIRECT
13526 IN(0)(052652), OUT(052652)
13527 66251
13528 TEST533A:
13529 PO, LOAD-ENUA(ETARGET402), ISETUP FOR D = ZERO TEST
13530 LOAD-ERROR(TEST533A), IERROR DIRECTORY KEY
13531 DCS-CTR(C6.), ICOMPARE AT TARGET
13532 NEXT, J/INIT533A
13533 (6625) DCS(1.00,1.0,0,0) BM(1001..00,11..11,00..0000..010...0.0.0.0.0...0.0000...0..0000,0...11,000...110,110,110)
13534 66661
13535 INIT533A:
13536 PO, BUMP-VERIFY, ICOUNT
13537 P3, CSPD(17)EMIT, IGET INITIAL PATTERN FOR D
13538 EMIT/052652, I"0101 0101 1010 1010"
13539 NEXT, J/INIT533A
13540 (6666) DCS(0.00,0.0,0,0.1) BM(0101..10,01..01,10..101..010...0.0.0.0.0...0.0000...1..0000,0...11,000...011,100,111)
13541 6347: 1(FREE)
13542 INIT533A:
13543 P2-T, D_CSPD(D17), IINITIAL D =(052652)
13544 D(C)ALU15, ISETUP D(C) FOR SHIFT = "0"
13545 NEXT, J/COMP533A
13546 (6347) DCS(0.00,0.0,0,0.0) BM(1010..10,00..00,00..0000..100...0.1.0.0.0...0.0000...0..0000,0...11,000...011,101,000)
13547 6350: 1(FREE)
13548 COMP533A:
13549 PO, BUMP-VERIFY, ICOUNT
13550 SETUP, D-DIRECT, IAMUX-BMUX-CMUX ALL DIRECT
13551 P2-T, D-D-SHIFTED-XOR-CSPB(B17), ICOMPARE D-SHIFTED:EXPECTED, BITWISE
13552 NEXT, J/GOBUT533A IEXPECTED=(052652)
13553 (6350) DCS(0.00,0.0,0,0.1) BM(0110..11,00..01,01..0000..000...0.1.0.0.0...0.0000...0..0000,0...11,000...011,101,001)
13554 6351: 1(FREE)
13555 GOBUT533A:
13556 SETUP, RETURN/TEST533B, IRETURN TO START OF NEXT SUBTEST
13557 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
13558 J/BUTD-IS-ZERO IGO TEST D IS ALL ZERO
13559 (6351) DCS(0.00,0.0,0,0.0) BM(0110..00,11..00,11..011..111...0.0.0.0.0...0.0000...0..0000,0...11,100...011,100,001)
13560
13561
13562
```

```
13563 1 - - - - -
13564
13565 1*** TEST 533B ***
13566 IREAD D DIRECTLY THRU "D[H1]D[L0]" PORT OF AMUXENI8L0, BMUX-CMUX/DIRECT
13567 IN(0)(125125), OUT(125125)
13568 66331
13569 TEST533B:
13570 PO, LOAD-ENUA(ETARGET402), ISETUP FOR D=ZERO TEST
13571 LOAD-ERROR(TEST533B), IERROR DIRECTORY KEY
13572 DCS-CTR(C6.), ICOMPARE AT TARGET
13573 NEXT, J/INIT533B
13574 (6633) DCS(1.00,1.0,0,0) BM(1001..00,11..11,00..0000..010...0.0.0.0.0...0.0000...0..0000,0...11,000...011,101,010)
13575 6352: 1(FREE)
13576 INIT533B:
13577 P3, CSPD(16)EMIT, IGET INITIAL PATTERN FOR D
13578 EMIT/125125, I"1010 1010 0101 0101"
13579 NEXT, J/INIT533B
13580 (6352) DCS(0.00,0.0,0,0.0) BM(1010..10,10..10,01..010..101...0.0.0.0.0...0.0001...1..0000,0...11,000...011,101,011)
13581 6353: 1(FREE)
13582 INIT533B:
13583 PO, BUMP-VERIFY, ICOUNT
13584 P2-T, D_CSPD(D16), IINITIAL D=(125125)
13585 D(C)ALU07, ISETUP D(C) FOR SHIFT = "0"
13586 NEXT, J/COMP533B
13587 (6353) DCS(0.00,0.0,0,0.1) BM(1010..10,00..00,00..0000..011...0.1.0.0.0...0.0001...0..0000,0...11,000...011,101,100)
13588 6354: 1(FREE)
13589 COMP533B:
13590 SETUP, D-DIRECT, IAMUX-BMUX-CMUX ALL DIRECT
13591 P2-T, D-D-SHIFTED-XOR-CSPB(B16), ICOMPARE D-SHIFTED:EXPECTED, BITWISE
13592 NEXT, J/GOBUT533B IEXPECTED=(125125)
13593 (6354) DCS(0.00,0.0,0,0.0) BM(0110..11,01..01,01..0000..000...0.1.0.0.0...0.0000...0..0000,0...11,000...011,101,101)
13594 6355: 1(FREE)
13595 GOBUT533B:
13596 SETUP, RETURN/SCOPE533B, IRETURN TO SCOPE LOOP TEST WORD
13597 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
13598 J/BUTD-IS-ZERO IGO TEST D IS ALL ZERO
13599 (6355) DCS(0.00,0.0,0,0.0) BM(0110..00,01..11,01..110..111...0.0.0.0.0...0.0000...0..0000,0...11,100...011,100,001)
13600 6356: 1(FREE)
13601 SCOPE533B:
13602 P3, CSPD(14)EMIT, EMIT/000377, ICONSTANT FOR USE BELOW
13603 NEXT, BUTD(SCOPE), INO ERROR: "TEST534A" (+1,WORDS)
13604 J/TEST534A I ERROR: "INIT533A" (-9,WORDS)
13605 (6356) DCS(0.00,0.1,0,0) BM(0000..10,00..00,11..111..111...0.0.0.0.0...0.0011...1..0000,0...11,000...110,110,111)
13606
13607
13608
13609 1 - - - - -
```

```
13610
13611 |*** TEST 534A ***
13612 |READ D THRU "D[LO]D[HI]" PORT OF AMUX[HI,LO], BMUX=CMUX/DIRECT
13613 |IN(0)(052652), OUT(125125)
13614 6667:
13615 TEST534A:
13616 PO, LOAD=ENUA(ZTARGET402), |SETUP FOR D=ZERO TEST
13617 LOAD=ERROR(TEST534A), |ERROR DIRECTORY KEY
13618 DCS=CTR(C5,), |COMPARE AT TARGET
13619 BUTA(CLR=FLAG=RES=UCON), |EMIT ON BUGDIN
13620 NEXT, J/INITD534A
13621 (6667) DCS(1.00,1.0,0.0) BM(1010,00,11,11,00,000,010...0,0,0,0,0,0,0,0000...0,0000,0...11,010,110,100)
13622 6664:
13623 INITD534A:
13624 PO, BUMP=VERIFY, |COUNT
13625 P2=T, D_CSPD(D17), |INITIAL D=(052652)
13626 D(C)_ALU15, |SETUP D(C) FOR SHIFT = "0"
13627 NEXT, J/COMP534A
13628 (6664) DCS(0.00,0.0,0.0,1) BM(1010,10,00,00,00,000,100...0,1,0,0,0,0,0,0000...0,0000,0...11,000,011,101,111)
13629 6357: 1(FREE)
13630 COMP534A:
13631 SETUP, D=SWAB, |AMUX/SWAB, BMUX=CMUX/DIRECT
13632 P2=T, D_D=SHIFTED-XOR-CSPB(B16), |COMPARE D=SHIFTED, EXPECTED, BITWISE
13633 NEXT, J/GOBUT534A |EXPECTED=(125125)
13634 (6357) DCS(0.00,0.0,0.0) BM(0110,11,01,01,01,000,000...0,1,0,0,0,0,0,0101...0,0000,0...11,000,011,110,000)
13635 6360: 1(FREE)
13636 GOBUT534A:
13637 SETUP, RETURN/TEST534B, |RETURN TO START OF NEXT SUBTEST
13638 NEXT, GOTO=PAGE(7), |BUT TABLE IS ON PAGE 7
13639 J/BUTD=IS=ZERO |GO TEST D IS ALL ZERO
13640 (6360) DCS(0.00,0.0,0.0) BM(0110,00,11,00,11,100,111...0,0,0,0,0,0,0,0000...0,0000,0...11,100,011,100,001)
13641
13642
13643
13644 | - - - - -
13645
13646 |*** TEST 534B ***
13647 |READ D THRU "D[LO]D[HI]" PORT OF AMUX[HI,LO], BMUX=CMUX/DIRECT
13648 |IN(0)(125125), OUT(052652)
13649 6634:
13650 TEST534B:
13651 PO, LOAD=ENUA(ZTARGET402), |SETUP FOR D=ZERO TEST
13652 LOAD=ERROR(TEST534B), |ERROR DIRECTORY KEY
13653 DCS=CTR(C5,), |COMPARE AT TARGET
13654 BUMP=VERIFY, |COUNT
13655 NEXT, J/INITD534B
13656 (6634) DCS(1.00,1.0,0.0,1) BM(1010,00,11,11,00,000,010...0,0,0,0,0,0,0,0000...0,0000,0...11,000,011,110,001)
13657 6361: 1(FREE)
```

```
13658 INITD534B:
13659 P2=T, D_CSPD(D16), |INITIAL D=(125125)
13660 D(C)_ALU07, |SETUP D(C) FOR SHIFT = "0"
13661 NEXT, J/COMP534B
13662 (6361) DCS(0.00,0.0,0.0) BM(1010,10,00,00,00,000,011...0,1,0,0,0,0,0,0001...0,0000,0...11,000,011,110,010)
13663 6362: 1(FREE)
13664 COMP534B:
13665 PO, BUMP=VERIFY, |COUNT
13666 SETUP, D=SWAB, |AMUX/SWAB, BMUX=CMUX/DIRECT
13667 P2=T, D_D=SHIFTED-XOR-CSPB(B17), |COMPARE D=SHIFTED, EXPECTED, BITWISE
13668 NEXT, J/GOBUT534B |EXPECTED=(052652)
13669 (6362) DCS(0.00,0.0,0.0,1) BM(0110,11,00,01,01,000,000...0,1,0,0,0,0,0,0101...0,0000,0...11,000,011,110,011)
13670 6363: 1(FREE)
13671 GOBUT534B:
13672 SETUP, RETURN/SCOPE534B, |RETURN TO SCOPE LOOP TEST WORD
13673 NEXT, GOTO=PAGE(7), |BUT TABLE IS ON PAGE 7
13674 J/BUTD=IS=ZERO |GO TEST D IS ALL ZERO
13675 (6363) DCS(0.00,0.0,0.0) BM(0110,00,01,11,10,100,111...0,0,0,0,0,0,0,0000...0,0000,0...11,100,011,100,001)
13676 6364: 1(FREE)
13677 SCOPE534B:
13678 P1, CSPD[15], EMIT, EMIT/177400, |CONSTANT FOR USE BELOW
13679 NEXT, BUTD[SCOPE], |NO ERROR: "TEST534C" (+1,WORDS)
13680 J/TEST534C |ERROR: "INITD534A" (-7,WORDS)
13681 (6364) DCS(0.00,0.1,0.0) BM(1111,10,11,11,00,000,000...0,0,0,0,0,0,0,0010...1,0000,0...11,000,110,110,101)
13682
13683
13684
13685 | - - - - -
13686
13687 |*** TEST 534C ***
13688 |READ D THRU "8+D(C)+D[HI]" PORT OF AMUX[HI,LO], BMUX=CMUX/DIRECT
13689 |IN(1)(000000), OUT(177400)
13690 6665:
13691 TEST534C:
13692 PO, LOAD=ENUA(ZTARGET402), |SETUP FOR D=ZERO TEST
13693 LOAD=ERROR(TEST534C), |ERROR DIRECTORY KEY
13694 DCS=CTR(C5,), |COMPARE AT TARGET
13695 NEXT, J/INITD534C
13696 (6665) DCS(1.00,1.0,0.0) BM(1010,00,11,11,00,000,010...0,0,0,0,0,0,0,0000...0,0000,0...11,000,110,110,010)
13697 6662:
13698 INITD534C:
13699 P2=T, D_ASPI(C000000), |INITIAL D=(000000)
13700 D(C)_1, |SETUP D(C) FOR SHIFT = "1"
13701 NEXT, J/COMP534C
13702 (6662) DCS(0.00,0.0,0.0) BM(1111,00,00,11,01,100,000...0,1,0,0,0,0,0,0000...0,0000,0...11,000,011,110,101)
13703 6365: 1(FREE)
13704 COMP534C:
```

```
13705          SETUP, D=RIGHT=0,          |CSP(15)=(177400)="1111 1111 0000 0000"  
13706          P2-T, D=D-SHIFTED-XOR-CSPB(B15), |AMUX/0=D(C)0<HI>, BMUX=CMUX/DIRECT  
13707          NEXT, J/GOBUT534C          |COMPARE D-SHIFTED; EXPECTED, BITWISE  
13708          (6365) DCS(0.00,0.0,0.0) BM(0110..11.10..01.01..000..000...0.1.0..0.0...0.0110...0..0000.0...11.000...011.110.110)|  
13709          6366: I(FREE)  
13710          GOBUT534C:  
13711          SETUP, RETURN/TEST534D,      |RETURN TO START OF NEXT SUBTEST  
13712          NEXT, GOTO-PAGE(7),          |BUT TABLE IS ON PAGE 7  
13713          J/BUTD-IS-ZERO              |GO TEST D IS ALL ZERO  
13714          (6366) DCS(0.00,0.0,0.0) BM(0110..00.11..00.11..101..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.100.001)|  
13715  
13716  
13717  
13718  
13719  
13720  
13721          |*** TEST 534D ***  
13722          IREAD D THRU "0=D(C)0<LO>" PORT OF AMUX(HI0LO), BMUX=CMUX/DIRECT  
13723          LIN(0)(17777), OUT(000377)  
13724          6635:  
13725          TEST534D:  
13726          PO, LOAD=ENUA(2TARGET402),      |SETUP FOR D=ZERO TEST  
13727          LOAD=ERROR(TEST534D),          |ERROR DIRECTORY KEY  
13728          DCS=CTR(C5),                  |COMPARE AT TARGET  
13729          NEXT, J/INITD534D  
13730          (6635) DCS(1.00,1.0,0.0) BM(1010..00.11..11.00..000..010...0.0.0..0.0...0.0000...0..0000.0...11.000...011.110.111)|  
13731          6367: I(FREE)  
13732          INITD534D:  
13733          P2-T, D=D&PHI(C17777),          |INITIAL D=(17777)  
13734          D(C)=0,                    |SETUP D(C) FOR SHIFT = "0"  
13735          NEXT, J/COMP534D  
13736          (6367) DCS(0.00,0.0,0.0) BM(1010..01.11..00.00..101..000...0.1.0..0.0...0.0000...0..0000.0...11.000...011.111.000)|  
13737          6370: I(FREE)  
13738          COMP534D:  
13739          |CSP(14)=(000377)="0000 0000 1111 1111"  
13740          SETUP, D=SIGNEXT,          |AMUX/0=D(C)0<LO>, BMUX=CMUX/DIRECT  
13741          P2-T, D=D-SHIFTED-XOR-CSPB(B14), |COMPARE D-SHIFTED; EXPECTED, BITWISE  
13742          NEXT, J/GOBUT534D          |EXPECTED=(000377)  
13743          (6370) DCS(0.00,0.0,0.0) BM(0110..11.11..01.01..000..000...0.1.0..0.0...0.0010...0..0000.0...11.000...011.111.001)|  
13744          6371: I(FREE)  
13745          GOBUT534D:  
13746          PO, BUMP=VERIFY,            |COUNT  
13747          SETUP, RETURN/SCOPE534D,      |RETURN TO SCOPE LOOP TEST WORD  
13748          NEXT, GOTO-PAGE(7),          |BUT TABLE IS ON PAGE 7  
13749          J/BUTD-IS-ZERO              |GO TEST D IS ALL ZERO  
13750          (6371) DCS(0.00,0.0,0.0) BM(0110..00.01..11.11..010..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.100.001)|
```

```
13751          6372: I(FREE)  
13752          SCOP534D:  
13753          NEXT, BUTD[SCOPE],            |NO ERROR: "TEST534E" (+1,WORDS)  
13754          J/TEST534E                    |ERROR: "EXPEC534C" (-9,WORDS)  
13755          (6372) DCS(0.00,0.1,0.0) BM(0000..00.00..00.00..000..000...0.0.0..0.0...0.0000...0..0000.0...11.000...110.110.011)|  
13756  
13757  
13758  
13759  
13760  
13761          |*** TEST 534E ***  
13762          IREAD D THRU "COUNTER=D[LO]" PORT OF AMUX(HI0LO), BMUX=CMUX/DIRECT  
13763          LIN(0)(000125), OUT(125125), CTR(252)  
13764          6663:  
13765          TEST534E:  
13766          PO, LOAD=ENUA(2TARGET402),      |SETUP FOR D=ZERO TEST  
13767          LOAD=ERROR(TEST534E),          |ERROR DIRECTORY KEY  
13768          DCS=CTR(C6),                  |COMPARE AT TARGET  
13769          NEXT, J/LODCNTR534E  
13770          (6663) DCS(1.00,1.0,0.0) BM(1001..00.11..11.00..000..010...0.0.0..0.0...0.0000...0..0000.0...11.000...110.110.000)|  
13771          6660:  
13772          LODCNTR534E:  
13773          PO, BUMP=VERIFY,            |COUNT  
13774          P2, COUNTER_B&PHI(C125252),    |PUT A (252) IN BM COUNTER  
13775          NEXT, J/INITD534E            |GETS D-BUS(7:0)  
13776          (6660) DCS(0.00,0.0,0.0) BM(0000..01.11..00.00..110..000...0.0.0..0.0...0.0000...0..0010.1...11.000...011.111.011)|  
13777          6373: I(FREE)  
13778          INITD534E:  
13779          PO, BUMP=VERIFY,            |COUNT  
13780          P2-T, D=CSPD(14)=AND=ASPHI(C052525), |INITIAL D=(000125), CSP(14)=(000377)  
13781          D(C)=ALUI5,                  |SETUP D(C) FOR SHIFT = "0"  
13782          NEXT, J/COMP534E  
13783          (6373) DCS(0.00,0.0,0.0) BM(1011..10.00..11.01..111..100...0.1.0..0.0...0.0011...0..0000.0...11.000...011.111.100)|  
13784          6374: I(FREE)  
13785          COMP534E:  
13786          SETUP, COUNT=D[LO],          |AMUX/COUNTER=D<LO> BMUX=CMUX/DIRECT  
13787          P2-T, D=D-SHIFTED-XOR-CSPB(B16), |COMPARE D-SHIFTED; EXPECTED, BITWISE  
13788          NEXT, J/GOBUT534E          |EXPECTED=(125125)  
13789          (6374) DCS(0.00,0.0,0.0) BM(0110..11.01..01.01..000..000...0.1.0..0.0...0.0011...0..0000.0...11.000...011.111.101)|  
13790          6375: I(FREE)  
13791          GOBUT534E:  
13792          SETUP, RETURN/TEST534F,      |RETURN TO START OF NEXT SUBTEST  
13793          NEXT, GOTO-PAGE(7),          |BUT TABLE IS ON PAGE 7  
13794          J/BUTD-IS-ZERO              |GO TEST D IS ALL ZERO  
13795          (6375) DCS(0.00,0.0,0.0) BM(0110..00.11..00.11..110..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.100.001)|  
13796  
13797
```

```
13798
13799 1 - - - - -
13800
13801 *** TEST 534F ***
13802 IREAD D THRU "COUNTERS[D(HI)]" PORT OF AMUX(HISLO), BMUX-CMUX/DIRECT
13803 IIN(0)(125000), OUT(052652), CTR(125)
13804 66361
13805 TEST534F:
13806     PO,      LOAD=ENUA(ZTARGET402),      ISETUP FOR D=ZERO TEST
13807     LOAD=ERROR(TEST534F),      IERROR DIRECTORY KEY
13808     DCS=CTR(C6.),      ICOMPARE AT TARGET
13809     NEXT,     J/LDCNTRS34F
13810 (6636) DCS[1.00,1.0,0,0] BM[1001.00,11.11,00.000,010...0,0,0,0...0.0000...0.0000,0...11,000...100,000,000]
13811 64001 1(FREE)
13812 LDCNTRS34F:
13813     P2,      COUNTER=ASPHI(C052525),      IPUT A (125) IN BM COUNTER
13814     NEXT,     J/INITD534F      I(GETS B-BUS(7-0))
13815 (6400) DCS[0.00,0.0,0,0] BM[0000.01,11.00,00.111,000...0,0,0,0...0.0000...0.0010,1...11,000...100,000,001]
13816 64011 1(FREE)
13817 INITD534F:
13818     PO,      BUMP=VERIFY,      ICOUNT
13819     P2=T,     D_CSPD[15]=AND=ASPHI(C125252),      IINITIAL D=(125000), CSP(15)=(177400)
13820     D[0]_ALU00,      ISETUP D[0] FOR SHIFT = "0"
13821     NEXT,     J/COMPS34F
13822 (6401) DCS[0.00,0.0,0,1] BM[1011.10,00.11,01.110,010...0,1,0,0...0.0010...0.0000,0...11,000...100,000,010]
13823 64021 1(FREE)
13824 COMPS34F:
13825     SETUP,     COUNT=D(HI),      IANUX/COUNTERS[D(HI)], BMUX-CMUX/DIRECT
13826     P2=T,     D_D-SHIFTED=XOR-CSPB(B17),      ICOMPARE D-SHIFTED/EXPECTED, BITWISE
13827     P3,      BUTA(LAST),      ICLEAR CTR TO (000) DURING P3
13828     NEXT,     J/GOBUT534F      IEXPECTED=(052652)
13829 (6402) DCS[0.00,0.0,0,0] BM[0110.11,00.01,01.000,000...0,1,0,0...0.0111...0.0000,0...10,000...100,000,011]
13830 64031 1(FREE)
13831 GOBUT534F:
13832     SETUP,     RETURN/TEST534G,      IRETURN TO START OF NEXT SUBTEST
13833     NEXT,     GOTO=PAGE(7),      IPUT TABLE IS ON PAGE 7
13834     J/BUTD=IS=ZERO      IGO TEST D IS ALL ZERO
13835 (6403) DCS[0.00,0.0,0,0] BM[0110.10,11.11,01.010,111...0,0,0,0...0.0000...0.0000,0...11,100...011,100,001]
13836
13837
13838 1 - - - - -
13839
13840 *** TEST 534G ***
13841 IREAD D THRU "COUNTERS[D(LO)]" PORT OF AMUX(HISLO), BMUX-CMUX/DIRECT
13842 ICOUNTER AND D SHOULD READ ALL ZERO, AFTER BEING ZAPPED IN TEST-534-F
13843 67521
13844 TEST534G:
13845     PO,      LOAD=ENUA(ZTARGET402),      ISETUP FOR D=ZERO TEST
```

```
13846     LOAD=ERROR(TEST534G),      IERROR DIRECTORY KEY
13847     DCS=CTR(C4.),      ICOMPARE AT TARGET
13848     NEXT,     J/COUNTER534G
13849 (6752) DCS[1.00,1.0,0,0] BM[1011.00,11.11,00.000,010...0,0,0,0...0.0000...0.0000,0...11,000...100,000,100]
13850 64041 1(FREE)
13851 COUNTER534G:
13852     P2=T,     D_COUNT=D(LO),      IREAD THE MUX INTO D
13853     NEXT,     J/GOBUT534G
13854 (6404) DCS[0.00,0.0,0,0] BM[1111.00,00.01,01.000,000...0,1,0,0...0.0011...0.0000,0...11,000...100,000,101]
13855 64051 1(FREE)
13856 GOBUT534G:
13857     PO,      BUMP=VERIFY,      ICOUNT
13858     SETUP,     RETURN/SCOPE534G,      IRETURN TO SCOPE LOOP TEST WORD
13859     NEXT,     GOTO=PAGE(7),      IPUT TABLE
13860     J/BUTD=IS=ZERO      IGO TEST D IS ALL ZERO
13861 (6405) DCS[0.00,0.0,0,1] BM[0110.10,10.00,00.110,111...0,0,0,0...0.0000...0.0000,0...11,100...011,100,001]
13862
13863
13864
13865 64061 1(FREE)
13866 SCOPE534G:
13867     NEXT,     BUID(SCOPE),      INO ERROR: "TEST535A" (+1,WORDS)
13868     J/TEST535A      IERROR: "LDCNTRS34F" (-11,WORDS)
13869 (6406) DCS[0.00,0.1,0,0] BM[0000.00,00.00,00.000,000...0,0,0,0...0.0000...0.0000,0...11,000...110,110,001]
13870
13871
13872
13873 1 - - - - -
13874
13875 *** TEST 535A ***
13876 IREAD D THRU "4*D[C]#AMUX<15:04>" PORT OF BMUX, AMUX-CMUX/DIRECT
13877 IIN(1)(122645), OUT(175132)
13878 66611
13879 TEST535A:
13880     PO,      LOAD=ENUA(ZTARGET402),      ISETUP FOR D=ZERO TEST
13881     LOAD=ERROR(TEST535A),      IERROR DIRECTORY KEY
13882     DCS=CTR(C7.),      ICOMPARE AT TARGET
13883     BUMP=VERIFY,      ICOUNT
13884     NEXT,     J/INIT535A
13885 (6661) DCS[1.00,1.0,0,1] BM[1000.00,11.11,00.000,010...0,0,0,0...0.0000...0.0000,0...11,000...110,101,110]
13886 66561
13887 INIT535A:
13888     P3,      CSPD[15]_EMIT,      IGET INITIAL PATTERN FOR D
13889     EMIT/122645,      I"1010 0101 1010 0101"
13890     NEXT,     J/EXPEC535A
13891 (6656) DCS[0.00,0.0,0,0] BM[1010.10,01.01,10.100,101...0,0,0,0...0.0010...1.0000,0...11,000...100,000,111]
13892 64071 1(FREE)
```

```
13893 EXEC535A1
13894 P3, CSPD[14]_EMIT, IGET EXPECTED PATTERN AFTER SHIFT
13895 EMIT/175132, I"1111 1010 0101 1010"
13896 NEXT, J/INITD535A
(6407) DCS[0.00,0.0,0.0] BM[1111..10.10..10.01..011..010...0.0,0.0,0.0...0.0011...1..0000,0...11,000...100,001,000]
13897
13898 6410: 1(FREE)
13899 INITD535A1
13900 P0, BUMP-VERIFY, ICOUNT
13901 P2-T, D_CSPD(D15), IINITIAL D=(122645)
13902 D[C]_ALU15, ISETUP D(C) FOR SHIFT = "1"
13903 NEXT, J/COMP535A
(6410) DCS[0.00,0.0,0.1] BM[1010..10.00..00.00...000...100...0.1,0.0,0.0...0.0010...0..0000,0...11,000...100,001,001]
13904
13905 6411: 1(FREE)
13906 COMP535A1
13907 SETUP, D-RIGHT-4, IANUX/DIRECT, BMUX/RIGHT-4, CMUX/DIRECT
13908 P2-T, D_D-SHIFTED-XOR-CSPB(B14), ICOMPARE D-SHIFTED:EXPECTED, BITWISE
13909 NEXT, J/GOBUT535A IEXPECTED=175132
(6411) DCS[0.00,0.0,0.0] BM[0110..11.11..01.01..000...000...0.1,0.0,0.0...0.1000...0..0000,0...11,000...100,001,010]
13910
13911 6412: 1(FREE)
13912 GOBUT535A1
13913 SETUP, RETURN/TEST535B, IRETURN TO START OF NEXT SUBTEST
13914 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
13915 J/BUTD-IS-ZERO IGO TEST D IS ALL ZERO
(6412) DCS[0.00,0.0,0.0] BM[0110..00.11..00.11..111..111...0.0,0.0,0.0...0.0000...0..0000,0...11,100...011,100,001]
13916
13917
13918
13919
13920 | - - - - -
13921
13922 I*** TEST 535B ***
13923 IREAD D THRU "4*D[C]#ANUX<15:04>" PORT OF BMUX, ANUX-CMUX/DIRECT
13924 IIN(0)(045132), OUT(002645)
13925 6637:
13926 TEST535B1
13927 P0, LOAD-ENVA(ZTARGET402), ISETUP FOR D=ZERO TEST
13928 LOAD-ERROR(TEST535B), IERROR DIRECTORY KEY
13929 DCS-CTR(C6), ICOMPARE AT TARGET
13930 BUMP-VERIFY, ICOUNT
13931 NEXT, J/EXEC535B
(6637) DCS[1.00,1.0,0.1] BM[1001..00.11..11.00..000...010...0.0,0.0,0.0...0.0000...0..0000,0...11,000...100,001,011]
13932
13933 6413: 1(FREE)
13934 EXEC535B1
13935 P3, CSPD[14]_EMIT, IGET EXPECTED PATTERN AFTER SHIFT
13936 EMIT/002645, I"0000 0101 1010 0101"
13937 NEXT, J/INITD535B
(6413) DCS[0.00,0.0,0.0] BM[0000..10.01..01.10..100...101...0.0,0.0,0.0...0.0011...1..0000,0...11,000...100,001,100]
13938
13939 6414: 1(FREE)
```

```
13940 INITD535B1
13941 P2-T, D_NOT-CSPB(B15), IINITIAL D=(055132)
13942 D[C]_ALU15, ISETUP D(C) FOR SHIFT = "0"
13943 NEXT, J/COMP535B
(6414) DCS[0.00,0.0,0.0] BM[0111..11.10..11.01..101...100...0.1,0.0,0.0...0.0000...0..0000,0...11,000...100,001,101]
13944
13945 6415: 1(FREE)
13946 COMP535B1
13947 SETUP, D-RIGHT-4, IANUX/DIRECT, BMUX/RIGHT-4, CMUX/DIRECT
13948 P2-T, D_D-SHIFTED-XOR-CSPB(B14), ICOMPARE D-SHIFTED:EXPECTED, BITWISE
13949 NEXT, J/GOBUT535B IEXPECTED=002645
(6415) DCS[0.00,0.0,0.0] BM[0110..11.11..01.01..000...000...0.1,0.0,0.0...0.1000...0..0000,0...11,000...100,001,110]
13950
13951 6416: 1(FREE)
13952 GOBUT535B1
13953 P0, BUMP-VERIFY, ICOUNT
13954 SETUP, RETURN/SCOPE535B, IRETURN TO SCOPE LOOP TEST WORD
13955 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
13956 J/BUTD-IS-ZERO IGO TEST D IS ALL ZERO
(6416) DCS[0.00,0.0,0.1] BM[0110..00.10..00.01..111..111...0.0,0.0,0.0...0.0000...0..0000,0...11,100...011,100,001]
13957
13958 6417: 1(FREE)
13959 SCOPE535B1
13960 P3, CSPD[16]_EMIT, EMIT/114631, ICONSTANT FOR USE BELOW
13961 NEXT, BUTD[SCOPE], INO ERROR: "TEST536A" (+1,WORDS)
13962 J/TEST536A I ERROR: "INIT535A" (-10,WORDS)
(6417) DCS[0.00,0.1,0.0] BM[1001..10.10..01.10..011..001...0.0,0.0,0.0...0.0001...1..0000,0...11,000...110,101,111]
13963
13964
13965
13966
13967 | - - - - -
13968
13969 I*** TEST 536A ***
13970 IREAD D THRU "2*D[C]#BMUX<15:02>" PORT OF CMUX, ANUX-BMUX/DIRECT
13971 IIN(1)(016161), OUT(143434)
13972 6647:
13973 TEST536A1
13974 P0, LOAD-ENVA(ZTARGET402), ISETUP FOR DD=ZERO TEST
13975 LOAD-ERROR(TEST536A), IERROR DIRECTORY KEY
13976 DCS-CTR(C7), ICOMPARE AT TARGET
13977 NEXT, J/INIT536A
(6647) DCS[1.00,1.0,0.0] BM[1000..00.11..11.00..000...010...0.0,0.0,0.0...0.0000...0..0000,0...11,000...110,101,100]
13978
13979 6654:
13980 INTT536A1
13981 P3, CSPD[15]_EMIT, IGET INITIAL PATTERN FOR D
13982 EMIT/016161, I"0001 1100 0111 0001"
13983 NEXT, J/EXEC536A
(6654) DCS[0.00,0.0,0.0] BM[0001..10.11..00.01..110...001...0.0,0.0,0.0...0.0010...1..0000,0...11,000...100,010,000]
13984
13985 6420: 1(FREE)
13986 EXEC536A1
```

```
13987      PO,      BUMP-VERIFY,      |COUNT
13988      P3,      CSPD[14]_EMIT,      |GET EXPECTED PATTERN AFTER SHIFT
13989      NEXT,      J/INITDS36A      |"1100 0111 0001 1100"
13990      (6420) DCS[0.00,0.0,0.0,1] BM[1100..10.01..11.00..011..100...0.0,0.0,0.0,0.0011...1..0000,0...11.000...100,010,001]
13991      6421: 1(FREE)
13992      TMTDS36A:
13993      P2-T,      D_CSPD(D15),      |INITIAL D=016161
13994      D[C]_ALU00,      |SETUP D[C] FOR SHIFT = "1"
13995      NEXT,      J/COMPS36A      |
13996      (6421) DCS[0.00,0.0,0.0,0] BM[1010..10.00..00.00..000..010...0.1,0.0,0.0,0.0010...0..0000,0...11.000...100,010,010]
13997      6422: 1(FREE)
13998      COMPS36A:
14000      SETUP,      D=RIGHT-2,      |ANUX=BMUX/DIRECT, CMUX/RIGHT-2
14001      P2-T,      D=D-SHIFTED-XOR-CSPB(B14),      |COMPARE D-SHIFTED/EXPECTED, BITWISE
14002      NEXT,      J/GOBUTS36A      |EXPECTED = (143434)
14003      (6422) DCS[0.00,0.0,0.0,0] BM[0110..11.11..01.11..000..000...0.1,0.0,0.0,0.0000...0..0000,0...11.000...100,010,011]
14004      6423: 1(FREE)
14005      GOBUTS36A:
14006      SETUP,      RETURN/TESTS36B,      |RETURN TO START OF NEXT SUBTEST
14007      NEXT,      GOTO-PAGE(7),      |BUT TABLE IS ON PAGE 7
14008      J/BUTD-IS-ZERO      |GO TEST D IS ALL ZERO
14009      (6423) DCS[0.00,0.0,0.0,0] BM[0110..00.11..01.00..000..111...0.0,0.0,0.0,0.0000...0..0000,0...11.100...011,100,001]
14010
14011
14012
14013
14014
14015      | - - - - -
14016      |*** TEST 536B ***
14017      |READ D THRU "D[C]BMUX<15:02>" PORT OF CMUX, ANUX=BMUX/DIRECT
14018      |IN(0)(161616), OUT(034343)
14019      6640:
14020      TESTS36B:
14021      PO,      LOAD=ENUA(ZTARGET402),      |SETUP FOR D=ZERO TEST
14022      LOAD=ERROR(TESTS36B),      |ERROR DIRECTORY KEY
14023      DCS=CTR(C6.),      |COMPARE AT TARGET
14024      NEXT,      J/EXPCS36B      |
14025      (6640) DCS[1.00,1.0,0.0,1] BM[1001..00.11..11.00..000..010...0.0,0.0,0.0,0.0000...0..0000,0...11.000...100,010,100]
14026      6424: 1(FREE)
14027      EXPCS36B:
14028      PO,      BUMP-VERIFY,      |COUNT
14029      P3,      CSPD[14]_EMIT,      |GET EXPECTED PATTERN AFTER SHIFT
14030      NEXT,      J/INITDS36B      |"1110 0011 1000 1110"
14031      (6424) DCS[0.00,0.0,0.0,1] BM[0011..10.10..00.11..100..011...0.0,0.0,0.0,0.0011...1..0000,0...11.000...100,010,101]
14032      6425: 1(FREE)
14033      TMTDS36A:
```

```
14034      P2-T,      D=NOT-CSPB(B15),      |INITIAL D=(161616)
14035      D[C]_ALU00,      |SETUP D[C] FOR SHIFT = "0"
14036      NEXT,      J/COMPS36B      |
14037      (6425) DCS[0.00,0.0,0.0,0] BM[0111..11.10..11.01..101..010...0.1,0.0,0.0,0.0000...0..0000,0...11.000...100,010,110]
14038      6426: 1(FREE)
14039      COMPS36B:
14040      PO,      BUMP-VERIFY,      |COUNT
14041      SETUP,      D=RIGHT-2,      |ANUX=BMUX/DIRECT, CMUX/RIGHT-2
14042      P2-T,      D=D-SHIFTED-XOR-CSPB(B14),      |COMPARE D-SHIFTED/EXPECTED, BITWISE
14043      NEXT,      J/GOBUTS36B      |EXPECTED=(034343)
14044      (6426) DCS[0.00,0.0,0.0,1] BM[0110..11.11..01.11..000..000...0.1,0.0,0.0,0.0000...0..0000,0...11.000...100,010,111]
14045      6427: 1(FREE)
14046      GOBUTS36B:
14047      SETUP,      RETURN/SCOPE536B,      |RETURN TO SCOPE LOOP TEST WORD
14048      NEXT,      GOTO-PAGE(7),      |BUT TABLE IS ON PAGE 7
14049      J/BUTD-IS-ZERO      |GO TEST D IS ALL ZERO
14050      (6427) DCS[0.00,0.0,0.0,0] BM[0110..00.10..00.11..000..111...0.0,0.0,0.0,0.0000...0..0000,0...11.100...011,100,001]
14051      6430: 1(FREE)
14052      SCOPE536B:
14053      PO,      BUMP-VERIFY,      |COUNT
14054      P3,      CSPD[17]_EMIT, EMIT/031463,      |CONSTANT FOR USE BELOW
14055      NEXT,      BUTD[SCOPE],      |NO ERROR: "TESTS36C" (+1,WORDS)
14056      J/TESTS36C      |ERROR: "INITS36A" (-10,WORDS)
14057      (6430) DCS[0.00,0.1,0.0,1] BM[0011..10.00..11.00..110..011...0.0,0.0,0.0,0.0000...1..0000,0...11.000...110,101,101]
14058
14059
14060
14061
14062
14063      | - - - - -
14064      |*** TEST 536C ***
14065      |READ D THRU "D[C]BMUX<15:01>" PORT OF CMUX, ANUX=BMUX/DIRECT
14066      |IN(1)(031463), OUT(114631)
14067      6655:
14068      TESTS36C:
14069      PO,      LOAD=FNUA(ZTARGET402),      |SETUP FOR D=ZERO TEST
14070      LOAD=EPORR(TESTS36C),      |ERROR DIRECTORY KEY
14071      DCS=CTR(C5.),      |COMPARE AT TARGET
14072      NEXT,      J/INITDS36C      |COUNT
14073      (6655) DCS[1.00,1.0,0.0,1] BM[1010..00.11..11.00..000..010...0.0,0.0,0.0,0.0000...0..0000,0...11.000...110,101,010]
14074      6657:
14075      TMTDS36C:
14076      PO,      BUMP-VERIFY,      |COUNT
14077      P2-T,      D_CSPD(D17),      |INITIAL D=(031463)
14078      D[C]_ALU00,      |SETUP D[C] FOR SHIFT = "1"
14079      NEXT,      J/COMPS36C      |"0011 0011 0011 0011"
14080      (6657) DCS[0.00,0.0,0.0,1] BM[1010..10.00..00.00..000..010...0.1,0.0,0.0,0.0000...0..0000,0...11.000...100,011,001]
```

```
14081 6431: 1(FREE)
14082 COMPS36C:
14083 SETUP, D=RIGHT-1, |ANUX=BMUX/DIRECT, CMUX/RIGHT-1
14084 P2-T, D=D-SHIFTED-XOR-CSPB(B16), |COMPARE D-SHIFTED;EXPECTED, BITWISE
14085 NEXT, J/GOBUTS36C |EXPECTEDD(14631)
(6431) DCS[0.00,0.0,0,0] BM[0110..11.01..01.10..0000..0000...0.1.0..0..0.0000...0..0000,0...11.000...100.011,010]
|"1001 1001 1001 1001"
14086
14087 6432: 1(FREE)
14088 GORUTS36C:
14089 SETUP, RETURN/TESTS36D, |RETURN TO START OF NEXT SUBTEST
14090 NEXT, GOTO-PAGE(7), |BUT TABLE IS ON PAGE 7
14091 J/BUTD-IS-ZERO |GO TEST D IS ALL ZERO
14092 (6432) DCS[0.00,0.0,0,0] BM[0110..00.11..01.00..001..111...0.0,0..0..0.0000...0..0000,0...11.100...011.100,001]
14093
14094
14095
14096
14097
14098
14099 |*** TEST 536D ***
14100 |READ D THRU "D(C)BMUX<15:01>" PORT OF CMUX, ANUX=BMUX/DIRECT
14101 |IN(0)(146314), OUT(063146)
14102 6641:
14103 TESTS36D:
14104 P0, LOAD=ENUA(2TARGET402), |SETUP FOR D=ZERO TEST
14105 LOAD=ERROR(TESTS36D), |ERROR DIRECTORY KEY
14106 DCS=CTR(C7.), |COMPARE AT TARGET
14107 NEXT, J/INIT536D
(6441) DCS[1.00,1.0,0,0] BM[1000..00.11..11.00..000..010...0.0,0..0..0.0000...0..0000,0...11.000...100.011,011]
14108
14109 6433: 1(FREE)
14110 INIT536D:
14111 P3, CSPD[15]_EMIT, |GET INITIAL PATTERN FOR D
14112 EMIT/146314, |"1100 1100 1100 1100"
14113 NEXT, J/EXPEC536D
(6433) DCS[0.00,0.0,0,0] BM[1100..10.11..00.11..001..100...0.0,0..0..0.0010...1..0000,0...11.000...100.011,100]
14114
14115 6434: 1(FREE)
14116 EXPEC536D:
14117 P3, CSPD[14]_EMIT, |GET EXPECTED PATTERN AFTER SHIFT
14118 EMIT/063146, |"0110 0110 0110 0110"
14119 NEXT, J/INITD536D
(6434) DCS[0.00,0.0,0,0] BM[0110..10.01..10.01..100..110...0.0,0..0..0.0011...1..0000,0...11.000...100.011,101]
14120
14121 6435: 1(FREE)
14122 INITD536D:
14123 P0, BUMP-VERIFY, |COUNT
14124 P2-T, D_CSPD(D15), |INITIAL D=(146314)
14125 D(C)_0, |SETUP D(C) FOR SHIFT = "0"
14126 NEXT, J/COMPS36D
(6435) DCS[0.00,0.0,0,0] BM[1010..10.00..00.00..000..000...0.1,0..0..0.0010...0..0000,0...11.000...100.011,110]
14127
```

```
14128 6436: 1(FREE)
14129 COMPS36D:
14130 SETUP, D=RIGHT-1, |ANUX=BMUX/DIRECT, CMUX/RIGHT-1
14131 P2-T, D=D-SHIFTED-XOR-CSPB(B14), |COMPARE D-SHIFTED;EXPECTED, BITWISE
14132 NEXT, J/GOBUTS36D |EXPECTEDD(063146)
(6436) DCS[0.00,0.0,0,0] BM[0110..11.11..01.10..0000..0000...0.1.0..0..0.0000...0..0000,0...11.000...100.011,111]
14133
14134 6437: 1(FREE)
14135 GORUTS36D:
14136 SETUP, RETURN/SCOPE536D, |RETURN TO SCOPE LOOP TEST WORD
14137 NEXT, GOTO-PAGE(7), |BUT TABLE IS ON PAGE 7
14138 J/BUTD-IS-ZERO |GO TEST D IS ALL ZERO
(6437) DCS[0.00,0.0,0,0] BM[0110..00.10..01.00..000..111...0.0,0..0..0.0000...0..0000,0...11.100...011.100,001]
14139
14140 6440: 1(FREE)
14141 SCOPE536D:
14142 NEXT, BUTD[SCOPE], |NO ERROR: "TESTS36E" (+1,WORDS)
14143 J/TESTS36E |ERROR: "INIT536C" (-11,WORDS)
(6440) DCS[0.00,0.1,0,0] BM[0000..00.00..00.00..000..000...0.0,0..0..0.0000...0..0000,0...11.000...110.101,011]
14144
14145
14146
14147
14148
14149
14150 |*** TEST 536E ***
14151 |READ D THRU BMUX<14:00>=SENDMUX(=SR15) PORT OF CMUX, ANUX=BMUX/DIRECT
14152 |IN(0)(146314), OUT(114631), SR=(100000)
14153 6653:
14154 TESTS36E:
14155 P0, LOAD=ENUA(2TARGET402), |SETUP FOR D=ZERO TEST
14156 LOAD=ERROR(TESTS36E), |ERROR DIRECTORY KEY
14157 DCS=CTR(C6.), |COMPARE AT TARGET
14158 P3, BUTA(CLR-FLAG-RES-UCON), |SELECT SR-LOAD, SENDMUX PORTS 0123, BUSDIN_EMIT
14159 NEXT, J/LOADSR536E
(6653) DCS[1.00,1.0,0,0] BM[1001..00.11..11.00..000..010...0.0,0..0..0.0000...0..0000,0...11.010...110.101,000]
14160
14161 6650:
14162 LOADSR536E:
14163 P0, BUMP-VERIFY, |COUNT
14164 P2-T, SR_BSPH(C100000), |1(100000) IN SR<15:00>
14165 NEXT, J/INITD536E
(6650) DCS[0.00,0.0,0,0] BM[1010..01.11..00.00..001..000...0.0,0..1..0..0.0000...0..0000,0...11.000...100.100,001]
14166
14167 6441: 1(FREE)
14168 INITD536E:
14169 P0, BUMP-VERIFY, |COUNT
14170 P2-T, D_CSPD(D15), |INITIAL D=(146314)
14171 D(C)_0, |SETUP D(C) FOR SHIFT = "0"
14172 NEXT, J/COMPS36E
(6441) DCS[0.00,0.0,0,0] BM[1010..10.00..00.00..000..000...0.1,0..0..0.0010...0..0000,0...11.000...100.100,010]
14173
14174 6442: 1(FREE)
```

```

14175 COMP536F;
14176 SETUP, D-LEFT-1,
14177
14178 P2-T, D-D-SHIFTED-XOR-CSPB(B16),
14179 NEXT, J/GOBUT536F
14180
14181 6442: DCS(0,00,0,0,0,0) BN(0110..11,01..01,00..0000,0000..0,1,0,0,0,0..0,0000...0,0000,0...11,000...100,100,011)
14182
14183 6443: I(FREF)
14184 GOBUT536F;
14185 SETUP, RETURN/TEST536F,
14186 NEXT, GOTO-PAGE(??),
14187
14188 J/BUFD-IS-ZERO
14189
14190 6443: DCS(0,00,0,0,0,0) BN(0110..00,11..01,00..010,111...0,0,0,0,0,0..0,0000...0,0000,0...11,100...011,100,001)
14191
14192
14193
14194
14195
14196
14197
14198
14199
14200
14201
14202
14203
14204
14205
14206
14207
14208
14209
14210
14211
14212
14213
14214
14215
14216
14217
14218
14219
14220
14221
14222
14223
14224
14225
14226
14227
14228
14229
14230
14231
14232
14233
14234
14235
14236
14237
14238
14239
14240
14241
14242
14243
14244
14245
14246
14247
14248
14249
14250
14251
14252
14253
14254
14255
14256
14257
14258
14259
14260
14261
14262
14263
14264
14265
14266
14267
14268
14269
14270
14271
14272
14273
14274
14275
14276
14277
14278
14279
14280
14281
14282
14283
14284
14285
14286
14287
14288
14289
14290
14291
14292
14293
14294
14295
14296
14297
14298
14299
14300
14301
14302
14303
14304
14305
14306
14307
14308
14309
14310
14311
14312
14313
14314
14315
14316
14317
14318
14319
14320
14321
14322
14323
14324
14325
14326
14327
14328
14329
14330
14331
14332
14333
14334
14335
14336
14337
14338
14339
14340
14341
14342
14343
14344
14345
14346
14347
14348
14349
14350
14351
14352
14353
14354
14355
14356
14357
14358
14359
14360
14361
14362
14363
14364
14365
14366
14367
14368
14369
14370
14371
14372
14373
14374
14375
14376
14377
14378
14379
14380
14381
14382
14383
14384
14385
14386
14387
14388
14389
14390
14391
14392
14393
14394
14395
14396
14397
14398
14399
14400
14401
14402
14403
14404
14405
14406
14407
14408
14409
14410
14411
14412
14413
14414
14415
14416
14417
14418
14419
14420
14421
14422
14423
14424
14425
14426
14427
14428
14429
14430
14431
14432
14433
14434
14435
14436
14437
14438
14439
14440
14441
14442
14443
14444
14445
14446
14447
14448
14449
14450
14451
14452
14453
14454
14455
14456
14457
14458
14459
14460
14461
14462
14463
14464
14465
14466
14467
14468
14469
14470
14471
14472
14473
14474
14475
14476
14477
14478
14479
14480
14481
14482
14483
14484
14485
14486
14487
14488
14489
14490
14491
14492
14493
14494
14495
14496
14497
14498
14499
14500
14501
14502
14503
14504
14505
14506
14507
14508
14509
14510
14511
14512
14513
14514
14515
14516
14517
14518
14519
14520
14521
14522
14523
14524
14525
14526
14527
14528
14529
14530
14531
14532
14533
14534
14535
14536
14537
14538
14539
14540
14541
14542
14543
14544
14545
14546
14547
14548
14549
14550
14551
14552
14553
14554
14555
14556
14557
14558
14559
14560
14561
14562
14563
14564
14565
14566
14567
14568
14569
14570
14571
14572
14573
14574
14575
14576
14577
14578
14579
14580
14581
14582
14583
14584
14585
14586
14587
14588
14589
14590
14591
14592
14593
14594
14595
14596
14597
14598
14599
14600
14601
14602
14603
14604
14605
14606
14607
14608
14609
14610
14611
14612
14613
14614
14615
14616
14617
14618
14619
14620
14621
14622
14623
14624
14625
14626
14627
14628
14629
14630
14631
14632
14633
14634
14635
14636
14637
14638
14639
14640
14641
14642
14643
14644
14645
14646
14647
14648
14649
14650
14651
14652
14653
14654
14655
14656
14657
14658
14659
14660
14661
14662
14663
14664
14665
14666
14667
14668
14669
14670
14671
14672
14673
14674
14675
14676
14677
14678
14679
14680
14681
14682
14683
14684
14685
14686
14687
14688
14689
14690
14691
14692
14693
14694
14695
14696
14697
14698
14699
14700
14701
14702
14703
14704
14705
14706
14707
14708
14709
14710
14711
14712
14713
14714
14715
14716
14717
14718
14719
14720
14721
14722
14723
14724
14725
14726
14727
14728
14729
14730
14731
14732
14733
14734
14735
14736
14737
14738
14739
14740
14741
14742
14743
14744
14745
14746
14747
14748
14749
14750
14751
14752
14753
14754
14755
14756
14757
14758
14759
14760
14761
14762
14763
14764
14765
14766
14767
14768
14769
14770
14771
14772
14773
14774
14775
14776
14777
14778
14779
14780
14781
14782
14783
14784
14785
14786
14787
14788
14789
14790
14791
14792
14793
14794
14795
14796
14797
14798
14799
14800
14801
14802
14803
14804
14805
14806
14807
14808
14809
1
```

```

14222 GOBUT536F;
14223 SETUP, RETURN/SCOPE536F, ;RETURN TO SCOPE TEST WORD
14224 NEXT, GOTO-PAGE(7), ;BUT TABLE IS ON PAGE 7
14225 J/RUTD-IS-ZERO ;GO TEST D IS ALL ZERO
(6447) DCS(0.00,0.0,0.0) BM(0110.00,10.01,01.000.111.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.100.001)

14226
14227 6450: I(FREE)
14228 SCOPE536F;
14229 P0, BUMP-VERIFY, ;ICOUNT
14230 P3, CSPD(02).EMIT, ;RES-CON #1: (FOR USE BELOW)
14231 EMITC, SFNDMUX-0123-BEL, ; SELECT SENDMUX PORTS 0-3
14232 SR-LEFT, GUARD-DIS, ; SR GOES LEFT, SR<00>-DIC
14233 NEXT, RUTD(SCOPE), ;NO ERROR: "SETEMIT537A" (+1,WORDS)
14234 J/SETEMIT537A ;ERROR: "LOADSR536E" (-9,WORDS)
(6450) DCS(0.00,0.1,0.0) BM(0101.10,00.00,00.000.000.0.0.0.0.0.0.0.1101.1.0000.0.0.11.000.110.101.001)

14235
14236
14237
14238
14239
14240
14241 *** TEST 537A ***
14242 THIS TEST VALIDATES THE "SENDMUX" INPUTS TO THE SHIFT-TREE.
14243 EACH "SENDMUX" OUTPUT IS SET TO 1/0 VALUES, AND THEN READ OUT INTO THE "SR" WHERE THEY
14244 ARE ALL SAVED TO CHECK AT ONCE.
14245 6651:
14246 SEITEMIT537A;
14247 P0, DCS-CTR(C15.), ;HOLD UP FOR NOW
14248 NEXT, GOTO-PAGE(4), ;XFER
14249 J/LOAD16537A ;
(6651) DCS(0.00,1.0,0.0) BM(0000.00,00.00,00.000.100.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.100.101.111.100)

14250
14251 4574:
14252 LOAD16537A;
14253 P3, CSPD(16).EMIT, ;BIT03 ONLY SET
14254 EMIT/000010, ;
14255 BUTA(CLR-FLAG-RES-UCON), ;BUSBIN <- EMIT, SR LOAD
14256 NEXT, J/LOAD14537A ;
(4574) DCS(0.00,0.0,0.0) BM(0000.10,00.00,00.001.000.0.0.0.0.0.0.0.0001.1.0.0000.0.0.11.010.0.010.111.110)

14257
14258 4276: I(FREE)
14259 LOAD14537A;
14260
14261 P3, CSPD(14).EMIT, ;BIT11 ONLY SET
14262 EMIT/004000, ;
14263 NEXT, J/LOAD03537A ;
(4276) DCS(0.00,0.0,0.0) BM(0000.10,10.00,00.000.000.000.0.0.0.0.0.0.0.0011.1.0.0000.0.0.11.000.0.010.111.111)

14264
14265 4277: I(FREE)
14266 LOAD03537A;
14267 P3, CSPD(03).EMIT, ;WHAT THE "SR" SHOULD BE
14268 EMIT/177242, ;AFTER THIS TEST

```

```
14269 NEXT, J/LOAD01537A
(4277) DCS(0.00,0.0,0.0) BM(1111..10.11..10.10..101..010..0.0,0.0,0.0,0.0,0.1100...1..0000,0...11,000...011,000,000)
14270
14271 4300: 1(FREE)
14272 LOAD01537A:
14273 P3, CSPD(011_EMIT, IRES-CON #2:
14274 ENITC, SENDMUX-4567-SEL, I SELECT SENDMUX PORTS 4-7
14275 SR-LEFT, GUARD-DIS, I SR QUES LEFT, SR<00>-D(C)
14276 NEXT, GOTO-PAGE(7), I XFER
14277 J/INITRS537A
(4300) DCS(0.00,0.0,0.0) BM(0001..10.00..00.00..000..111..0.0,0.0,0.0,0.0,0.1110...1..0000,0...11,100...001,011,001)
14278
14279 7131: 1(FREE)
14280 INITRS537A:
14281 P2-T, SR-ALL-ONES, I START 'SR' WITH ALL ONES
14282 NEXT, J/SETRESA537A
(7131) DCS(0.00,0.0,0.0) BM(1111..00.00..11.01..101..000..0.0,0.1,0.0,0.0,0.0000...0..0000,0...11,000...001,011,101)
14283
14284 7135: 1(FREE)
14285 SETRESA537A:
14286 P2, RES_CSPD(D02), I LOAD RES W/ RES-CON#1 (SENDMUX-0123)
14287 NEXT, GOTO-PAGE(6), I XFER
14288 J/NEWCTRS537A
(7135) DCS(0.00,0.0,0.0) BM(0000..10.00..00.00..000..110..0.0,0.0,0.0,0.0,0.1101...0..1000,1...11,100...110,100,011)
14289
14290 6643:
14291 NEWCTRS537A:
14292 P0, LOAD-ENUA(SETRESB537A), I COMPARE POINT #2
14293 LOAD-ERROR(NEWCTRS537A), I ERROR DIRECTORY KEY
14294 DCS-CTR(C14), I COMPARE BELOW
14295 NEXT, J/AR3-537A
(6643) DCS(1.00,1.0,0.0) BM(0001..00.11..10.11..111..000..0.0,0.0,0.0,0.0,0.0000...0..0000,0...11,000...100,101,001)
14296
14297 6451: 1(FREE)
14298 AR3-537A:
14299 P0, BUMP-VERIFY, I COUNT
14300 P2-T, D_NOT-CSPD(D16), I ID=(177767), BIT03="0"
14301 NEXT, GOTO-PAGE(7), I XFER
14302 J/BR3-537A
(6451) DCS(0.00,0.0,0.0) BM(0111..10.00..11.01..101..111...0.0,0.0,0.0,0.0,0.0001...0..0000,0...11,100...001,011,100)
14303
14304 7134: 1(FREE)
14305 AR3-537A:
14306 P2-T, D-D-RIGHT-3, I USE SENDMUX PORT 1 = AMUX03 = "0"
14307 D(C)_ALU00, I INTO D(C) FOR SR
14308 NEXT, J/CR3-537A
(7134) DCS(0.00,0.0,0.0) BM(1111..00.00..01.00..000..010...0.0,0.0,0.0,0.0,0.1000...0..0000,0...11,000...001,011,111)
14309
14310 7137: 1(FREE)
14311 CR3-537A:
14312 P2-T, SR-SR-LEFT-1, I SENDMUX OUTPUT INTO SR<00>
```

```
14313 D_CSPB(B16), SAVE-D(C), I D=(000010), BIT03="1"
14314 NEXT, J/DR3-537A
(7137) DCS(0.00,0.0,0.0) BM(1010..11.01..00.00..000..111...0.0,0.1,0.0,0.0,0.0000...0..0000,0...11,000...001,100,000)
14315
14316 7140: 1(FREE)
14317 DR3-537A:
14318 P2-T, D-D-RIGHT-3, I USE SENDMUX PORT 1 = AMUX03 = "1"
14319 D(C)_ALU00, I INTO D(C) FOR SR
14320 NEXT, J/AR7-537A
(7140) DCS(0.00,0.0,0.0) BM(1111..00.00..01.00..000..010...0.0,0.0,0.0,0.0,0.1000...0..0000,0...11,000...001,100,001)
14321
14322 7141: 1(FREE)
14323 AR7-537A:
14324 P2-T, SR-SR-LEFT-1, I SENDMUX OUTPUT INTO SR<00>
14325 D_NOT-ASPHI(C000200), SAVE-D(C), I D=(177577), BIT07="0"
14326 NEXT, J/BR7-537A
(7141) DCS(0.00,0.0,0.0) BM(0000..00.00..11.01..010..111...0.0,0.1,0.0,0.0,0.0000...0..0000,0...11,000...001,100,010)
14327
14328 7142: 1(FREE)
14329 RP7-537A:
14330 P2-T, D-D-RIGHT-7, I USE SENDMUX PORT 2 = D07 = "0"
14331 D(C)_ALU00, I INTO D(C) FOR SR
14332 NEXT, J/CR7-537A
(7142) DCS(0.00,0.0,0.0) BM(1111..00.00..01.00..000..010...0.0,0.0,0.0,0.0,0.0110...0..0000,0...11,000...001,100,011)
14333
14334 7143: 1(FREE)
14335 CR7-537A:
14336 P2-T, SR-SR-LEFT-1, I SENDMUX OUTPUT INTO SR<00>
14337 D-ASPHI(C000200), SAVE-D(C), I D=(000200), BIT07="1"
14338 NEXT, J/DR7-537A
(7143) DCS(0.00,0.0,0.0) BM(1111..00.00..11.01..010..111...0.0,0.1,0.0,0.0,0.0000...0..0000,0...11,000...001,100,100)
14339
14340 7144: 1(FREE)
14341 RP7-537A:
14342 P2-T, D-D-RIGHT-7, I USE SENDMUX PORT 2 = D07 = "1"
14343 D(C)_ALU00, I INTO D(C) FOR SR
14344 NEXT, J/AR11-537A
(7144) DCS(0.00,0.0,0.0) BM(1111..00.00..01.00..000..010...0.0,0.0,0.0,0.0,0.0110...0..0000,0...11,000...001,100,101)
14345
14346 7145: 1(FREE)
14347 AR11-537A:
14348 P2-T, SR-SR-LEFT-1, I SENDMUX OUTPUT INTO SR<00>
14349 D_NOT-CSPB(B14), SAVE-D(C), I D=(173777), BIT11="0"
14350 NEXT, J/BR11-537A
(7145) DCS(0.00,0.0,0.0) BM(0111..11.11..11.01..101..111...0.0,0.1,0.0,0.0,0.0000...0..0000,0...11,000...001,100,110)
14351
14352 7146: 1(FREE)
14353 RP11-537A:
14354 P2-T, D-D-RIGHT-11, I USE SENDMUX PORT 3 = AMUX03 = "0"
14355 D(C)_ALU00, I INTO D(C) FOR SR
14356 NEXT, J/CR11-537A
(7146) DCS(0.00,0.0,0.0) BM(1111..00.00..01.00..000..010...0.0,0.0,0.0,0.0,0.1110...0..0000,0...11,000...001,100,111)
14357
14358 7147: 1(FREE)
```

```
14359 CR11-537A1
14360 P2-T, SR_SR-LEFT-1, |SENDMUX OUTPUT INTO SR<00>
14361 D_CSPB(B14), SAVE-D(C), |D=(004000), BIT11="1"
14362 NEXT, J/DR11-537A |
(7147) DCS(0.00,0.0,0.0) BM(1010.11,11.00,00.0000.111.0.1.0.0.0.0000.0.0000.0.11.000.001.101.000)

14363 7150: 1(FREE)
14364 DR11-537A1
14365 P2-T, D_D-RIGHT-11, |USE SENDMUX PORT 3 = AMUX03 = "1"
14366 D(C)_ALU00, |INTO D(C) FOR SR
14367 NEXT, J/FR3-537A |
(7150) DCS(0.00,0.0,0.0) BM(1111.00,00.01.00.0000.010.0.0.1.0.0.0.0.1110.0.0000.0.11.000.001.101.001)

14369 7151: 1(FREE)
14370 FR3-537A1
14371 P2-T, SR_SR-LEFT-1, |SENDMUX OUTPUT INTO SR<00>
14372 D_NOT-CSPB(B16), SAVE-D(C), |D=(177767), BIT03="0"
14373 NEXT, J/SETRESB537A |
(7151) DCS(0.00,0.0,0.0) BM(0111.11.01.11.01.101.111.0.0.1.1.0.0.0.0000.0.0.0000.0.11.000.001.111.000)

14375 7370:
14376 SETRESB537A1
14377 P2, RES_CSPD(D01), |LOAD RES W/RES-CON#2 (SENDMUX-4567)
14378 NEXT, J/FR3-537A |
(7370) DCS(0.00,0.0,0.0) BM(0000.10.00.00.00.0000.000.0.0.0.0.0.0.1110.0.0.1000.1.11.000.001.101.010)

14380 7152: 1(FREE)
14381 FR3-537A1
14382 P2-T, D_D-RIGHT-3, |USE SENDMUX PORT 5 = AMUX03 = "0"
14383 D(C)_ALU00, |INTO D(C) FOR SR
14384 NEXT, J/GR3-537A |
(7152) DCS(0.00,0.0,0.0) BM(1111.00,00.01.00.0000.010.0.0.1.0.0.0.0.1000.0.0.0000.0.11.000.001.101.011)

14386 7153: 1(FREE)
14387 GR3-537A1
14388 P2-T, SR_SR-LEFT-1, |SENDMUX OUTPUT INTO SR<00>
14389 D_CSPB(B16), SAVE-D(C), |D=(000010), BIT03="1"
14390 NEXT, J/HR3-537A |
(7153) DCS(0.00,0.0,0.0) BM(1010.11.01.00.00.0000.111.0.0.1.1.0.0.0.0000.0.0.0000.0.11.000.001.101.100)

14392 7154: 1(FREE)
14393 HR3-537A1
14394 P2-T, D_D-RIGHT-3, |USE SENDMUX PORT 5 = AMUX03 = "1"
14395 D(C)_ALU00, |INTO D(C) FOR SR
14396 NEXT, J/AL1-537A |
(7154) DCS(0.00,0.0,0.0) BM(1111.00,00.01.00.0000.010.0.0.1.0.0.0.0.1000.0.0.0000.0.11.000.001.101.101)

14398 7155: 1(FREE)
14399 AL1-537A1
14400 P2-T, SR_SR-LEFT-1, |SENDMUX OUTPUT INTO SR<00>
14401 D_ALL-ONES, |D=(177777)
14402 NEXT, GOTO-PAGE(4), |XFER
14403 J/BL1-537A |
14404
```

```
(7155) DCS(0.00,0.0,0.0) BM(1111.00,00.11.01.101.100.0.0.1.1.0.0.0.0.0000.0.0.0000.0.11.100.010.100.010)

14405 4242: 1(FREE)
14406 RL1-537A1
14407 P2-T, D_D-LEFT-1, |USE SENDMUX PORT 4 = HARD "0"
14408 D(C)_ALU00, |INTO D(C) FOR SR
14409 NEXT, J/TESTS37A |
(7147) DCS(0.00,0.0,0.0) BM(1111.00,00.01.00.0000.010.0.0.1.0.0.0.0.0000.0.0.0000.0.11.000.001.111.010)

14411 4572:
14412 TEST537A1
14413 P0, LOAD-ENUA(ZTARGET402), |SETUP FOR D-IS-ZERO COMPARE
14414 LOAD-ERROR(TESTS37A), |ERROR DIRECTORY KEY
14415 DCS-CTR(C4), |COMPARE AT TARGET
14416 BUMP-VERIFY, |COUNT
14417 P2-T, SR_SR-LEFT-1, |SENDMUX OUTPUT INTO SR<00>
14418 NEXT, J/COMPS37A |
(4572) DCS(1.00,1.0,0.0) BM(1011.00,11.11.00.0000.010.0.0.0.1.0.0.0.0.0000.0.0.0000.0.11.000.001.000.010)

14420 4302: 1(FREE)
14421 COMPS37A1
14422 P2-T, D_SR-XOR-CSPD(D03), |COMPARE RECEIVED:EXPECTED BITWISE
14423 P3, RUTA(CLR-FLAG-RES-UCON), |RESET SR TO LOAD/GUARD-DIS
14424 NEXT, J/GOBUT537A |
(4302) DCS(0.00,0.0,0.0) BM(1010.10.00.00.00.0000.000.0.0.1.0.0.0.0.1100.0.0.0000.0.11.010.011.000.011)

14426 4303: 1(FREE)
14427 GOBUT537A1
14428 SETUP, RETURN/SCOPE537A, |RETURN TO SCOPE LOOP TEST WORD
14429 NEXT, GOTO-PAGE(7), |PUT TABLE IS ON PAGE 7
14430 J/BUTD-IS-ZERO |GO TEST D IS ALL ZERO
(4303) DCS(0.00,0.0,0.0) BM(0100.00,01.10.00.100.111.0.0.0.0.0.0.0.0.0000.0.0.0000.0.11.100.001.100.001)

14432 4304: 1(FREE)
14433 SCOPE537A1
14434 P0, BUMP-VERIFY, |COUNT
14435 RUSDIN-EMIT-11, |RESET PROC UCON
14436 EN-CLK-IR(15-00), |
14437 NEXT, BUTD(SCOPE), |NO ERROR: "TEST551A" (+1,WORD)
14438 J/TEST551A |ERROR: "LOAD16537A" (-30,WORDS)
(4304) DCS(0.00,0.1,0.1) BM(0000.00,00.01.0000.100.0.0.0.0.0.0.1.1001.0.0.0000.0.11.000.001.111.101)

14440
14441
14442
14443
14444
14445 1.PAGE=====
14446
14447
14448 .TnC * TEST551: BASE MACHINE DATAPATH COUNTER CAN COUNT
14449
14450
14451 |=====
```

```

14452 1*
14453 1* TESTS: 551 A = C UMWORDS: 015 + 035
14454 1*
14455 1* FUNCTIONS:
14456 1*
14457 1* THE FOLLOWING THREE TESTS USE A COMMON SUBROUTINE TO TEST THE COUNTING
14458 1* ABILITY OF THE BASE MACHINE DATAPATH COUNTER, USING THE THREE ACTIVE
14459 1* BITS THAT ENABLE COUNTING. ADMITTEDLY THIS SEEMS LIKE OVERKILL, BUT
14460 1* THIS METHOD WAS THE LEAST EXPENSIVE IN TERMS OF NUMBER OF MICROWORDS
14461 1* USED FOR THE TESTING.
14462 1*
14463 1*****
14464
14465
14466
14467 1THE FIRST TEST USES THE ACTIVE BUT(#13) "SR<1:0>COUNT-IS-377" TO CHECK THE COUNTER
14468 4575:
14469 TEST551A:
14470 PO, LOAD-ERROR(TEST551A), 1ERROR DIRECTORY KEY
14471 DCS-CTR(C15.), 1HOLD UP FOR NOW
14472 BUMP-VERIFY, 1BUMP DCS COUNTER
14473 NEXT, J/LOADIN551A
(4575) DCS[1.00.1.0.0.1] BM[0000..00.00..00.00..000..0.0.0..0.0000...0..0000.0...11.000...011.000.101]
14474
14475 4305: 1(FREE)
14476 LOADIN551A:
14477 P3, CSPO[17]_EMIT, EMIT/400, 1INCREMENT FOR D, TO MATCH COUNTER
14478 NEXT, J/SETSR551A
(4305) DCS[0.00.0.0.0.0] BM[0000..10.00..01.00..000..0.0.0..0.0000...1..0000.0...11.000...011.000.110]
14479
14480 4306: 1(FREE)
14481 SETSR551A:
14482 PO, BUMP-VERIFY, 1COUNT
14483 P2-T, SR_ZERO, 1KEY IN SR<1:0>=(00) FOR SELECT THIS BUT
14484 NEXT, J/GOTEST551A
(4306) DCS[0.00.0.0.0.1] BM[0011..00.00..00.00..000..0.0.1..0.0000...0..0000.0...11.000...101.110.100]
14485
14486 4564:
14487 GOTEST551A:
14488 SETUP, RETURN/TEST551B, 1GO TO TEST SUBROUTINE
14489 NEXT, CALL(COUNT-TEST) 1 (SEE DESCRIP, FOLLOWING)
(4564) DCS[0.00.0.0.0.0] BM[0100..00.10..11.11..110..100..0.0.0..0.0000...0..0000.0...11.100...011.000.001]
14490
14491
14492
14493
14494 1 - - - - -
14495
14496 1THE SECOND TEST USES THE ACTIVE BUT(#25) "COUNT-IS-377" TO CHECK THE COUNTER
14497 4576:
14498 TEST551B:
14499 PO, LOAD-ENUA(ZTARGET400), 1BIT<1:0> CLEAR
14500 LOAD-ERROR(TEST551B), 1ERROR DIRECTORY KEY
14501 DCS-CTR(C4.), 1COMPARE AT TARGET

```

```

14502 NEXT, BUTD[SCOPE], 1NO ERROR: "SETSR551B" (+1, WORDS)
14503 1ERROR: "GOTEST551A" (-1, WORDS) REPEAT PREV TEST
(4576) DCS[1.00.1.1.0.0] BM[1011..00.11..11.00..000..000..0.0.0..0.0000...0..0000.0...11.000...101.110.101]
14504
14505 4565:
14506 SFTSR551B:
14507 P2-T, SR_SR-PLUS-1, 1KEY IN SR<1:0>=(01) FOR SELECT THIS BUT
14508 NEXT, J/GOTEST551B
(4565) DCS[0.00.0.0.0.0] BM[1001..01.11..00.00..000..000..0.0.1..0.0000...0..0000.0...11.000...101.110.000]
14509
14510 4560:
14511 GOTEST551B:
14512 SETUP, RETURN/GOTEST551B, 1RETURN TO START OF NEXT SUBTEST
14513 NEXT, GOTO-PAGE(7), 1BUT TABLE
14514 J/BUTCOUNT-IS-377 1FAKE A TEST
(4560) DCS[0.00.0.0.0.0] BM[0111..00.00..11.01..111..111..0.0.0..0.0000...0..0000.0...11.100...001.011.110]
14515
14516 7136: 1(FREE)
14517 BUTCOUNT-IS-377:
14518 NEXT, BUT(COUNT-IS-377), 1SHOULD TARGET INTO *** ** *00
14519 J/ZTARGET400 1BIT<1:0> BOTH CLEAR
(7136) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..000..0.0.0..0.0000...0..0000.0...10.101...100.000.000]
14520
14521
14522 1 - - - - -
14523
14524
14525 1 AND NOW TEST THAT THE ACTUAL BUT/COUNTER WORKS
14526
14527 7157: 1(PREF)
14528 GOTEST551C:
14529 SETUP, RETURN/TEST551C, 1GO TO TEST SUBROUTINE
14530 NEXT, CALL(COUNT-TEST) 1 (SEE DESCRIP, FOLLOWING)
(7157) DCS[0.00.0.0.0.0] BM[0100..00.10..11.10..110..100..0.0.0..0.0000...0..0000.0...11.100...011.000.001]
14531
14532
14533
14534
14535 1 - - - - -
14536
14537 1THE THIRD TEST USES THE ACTIVE BUT(#17) "COUNT-IS-377#D[C]" TO CHECK THE COUNTER
14538 4566:
14539 TEST551C:
14540 PO, LOAD-ERROR(TEST551C), 1ERROR DIRECTORY KEY
14541 DCS-CTR(C15.), 1HOLD UP FOR NOW
14542 NEXT, BUTD[SCOPE], 1NO ERROR: "SETSR551C" (+1, WORDS)
14543 J/SETSR551C 1 ERROR: "GOTEST551B" (-1, WORDS) REPEAT PREV TEST
(4566) DCS[1.00.1.1.0.0] BM[0000..00.00..00.00..000..000..0.0.0..0.0000...0..0000.0...11.000...101.110.001]
14544
14545 4561:
14546 SETSR551C:
14547 P2-T, SR_SR-PLUS-1, 1KEY IN SR<1:0>=(10) FOR SELECT THIS BUT

```

```

14548 NEXT, GOTO=PAGE(5), XFER
14549 J/GOTEST551C
(4561) DCS(0,00,0,0,0,0) BM(1001,01,11,00,00,0000,101,0,0,1,0,0,0,0000,0,0000,0,0,11,100,110,010,1101)
14550
14551 56261
14552 GOTEST551C
14553 SETUP, RETURN/SCOPE551, GO TO TEST SUBROUTINE
14554 NFXT, CALL(COUNT-TEST) (SEE DESCRIP, FOLLOWING)
(5626) DCS(0,00,0,0,0,0) BM(0101,00,01,10,00,101,100,0,0,0,0,0,0,0000,0,0000,0,0,11,100,011,000,0011)
14555
14556
14557
14558 53051 1(FREE)
14559 SCOPE551:
14560 P0, BUSDIN_EXIT=11, RESET PROC UCON
14561 EN-CLK-IR(15=00),
14562 NEXT, BUTD(SCOPE), NO ERROR: "TEST610A1" (NEXT SECTION)
14563 J/TEST610A1 ERROR: "GOTEST551C" (-1, WORDS) REPEAT PREV TEST
(5305) DCS(0,00,0,1,0,0) BM(0000,00,00,00,01,000,100,0,0,0,0,0,0,1,1001,0,0000,0,0,11,000,110,010,1111)
14564
14565
14566
14567 | - - - - -
14568
14569 |*** B.M. COUNTER TESTING ROUTINE ***
14570
14571 | THIS ROUTINE USES THE B.M. COUNTER TO COUNT FROM 000-377, WAITING FOR IT
14572 | TO OVERFLOW AT THE RIGHT MOMENT. AT THE SAME TIME THE D REGISTER HAS BEEN
14573 | TRACKING THE PROGRESS OF THE COUNTER, WAITING FOR IT TO OVERFLOW, COMPARING
14574 | COUNT FOR COUNT THE INCREMENTED VALUES, AND GUARANTEEING WE WILL EXIT THE
14575 | LOOP IF THE COUNTER SOMEHOW NEVER OVERFLOWS.
14576
14577
14578 4301: 1(FREE)
14579 COUNTER01:
14580 P2, COUNTER_BSPHI(C000000), LOAD COUNTER WITH (000) TO START
14581 NFXT, J/COUNTER02
(4301) DCS(0,00,0,0,0,0) BM(0000,01,11,00,00,100,000,0,0,0,0,0,0,0,0000,0,0,0010,1,11,000,011,001,0001)
14582
14583 4310: 1(FREE)
14584 COUNTER02:
14585 P2-T, D_ZERO, D(C)_ALU15, ZERO D, D(C)
14586 P3, A#BSPHI(17)_D, ASPHI(17) WILL TRACK THE COUNTER
14587 NFXT, J/COUNTER03
(4310) DCS(0,00,0,0,0,0) BM(0011,00,11,00,01,011,100,0,0,1,0,0,0,0,0000,0,0,1011,0,11,000,101,010,1101)
14588
14589
14590 |*** THE LOOP FOR THE COUNTER TEST ENTERS HERE ***
14591 45261
14592 COUNTER03:
14593 P2-T, D_A-XOR-B, SAVE=D(C), COMPARE COUNTER:TRACKER
14594 BUS-A, COUNT#D(L0), I=(COUNTER)*000

```

```

14595 BUS-B_BSPHI(R17), I=(TRACKER)*000
14596 NEXT, BUTR(D(C)-B), (D(C) HOLDS THE CARRYOUT FROM THE TRACKING REGISTER
14597 J/COUNTER04 IF (0), GOTO(COUNTER04), CONTINUE TESTING
14598 (4576) DCS(0,00,0,0,0,0) BM(0110,01,11,01,01,011,111,0,0,1,0,0,0,0,0011,0,0,0000,0,0,10,011,101,100,1011)
14599
14600
14601 |ENTER HERE FOR CONTINUE TEST
14602 45451
14603 COUNTER04:
14604 P0, DCS-CTR(C5,), HOLD UP ON DCS COMPARE
14605 NEXT, BUTR(D14=00-E0=0), COMPARE COUNTER:TRACKER, GENERATED ABOVE
14606 J/COUNTER12 IF EQUAL, GOTO(COUNTER05), CONTINUE TEST
14607 (4545) DCS(0,00,1,0,0,0) BM(1010,00,00,00,00,000,000,0,0,0,0,0,0,0,0000,0,0,0000,0,0,01,101,101,101,1011)
14608
14609
14610 |ENTER HERE FOR CONTINUE TEST
14611 45571
14612 COUNTER05:
14613 P2-T, D_ZERO, D(C)_ALU15, TRY TO FORCE EXIT
14614 NEXT, BUTP(CASE), DETERMINE WHO TO TEST:
14615 ISR0 -> COUNTER06
14616 J/COUNTER06 ISR1 -> COUNTER07
14617 (4557) DCS(0,00,0,0,0,0) BM(0011,00,00,00,00,000,100,0,0,1,0,0,0,0,0000,0,0,0000,0,0,00,000,101,011,1001)
14618
14619 |ENTER HERE FOR SR=0
14620 45341
14621 COUNTER06:
14622 P2-T, D_A-PLUS-B, D(C)_COUT15, INCREMENT TRACKING REGISTER
14623 BUS-A_ASPHI(R17),
14624 BUS-B_CSPD(D17),
14625 P3, A#BSPHI(17)_D,
14626 NEXT, BUTA(SR1=09-COUNT-15-377),
14627 (4534) DCS(0,00,0,0,0,0) BM(1001,10,11,11,01,011,110,0,0,1,0,0,0,0,0000,0,0,1011,0,0,01,011,101,010,1101)
14628
14629 |ENTER HERE FOR SR=1
14630 45351
14631 COUNTER07:
14632 P2-T, D_A-PLUS-B, D(C)_COUT15, INCREMENT TRACKING REGISTER
14633 BUS-A_ASPHI(R17),
14634 BUS-B_CSPD(D17),
14635 P3, A#BSPHI(17)_D,
14636 NEXT, BUTA(COUNT-TS-377),
14637 (4535) DCS(0,00,0,0,0,0) BM(1001,10,11,11,01,011,110,0,0,1,0,0,0,0,0000,0,0,1011,0,0,10,101,101,010,1101)
14638
14639 |ENTER HERE FOR SR=2
14640 45361
14641 COUNTER08:
14642 P2-T, D_A-PLUS-B, D(C)_COUT15, INCREMENT TRACKING REGISTER
14643 BUS-A_ASPHI(R17),

```

```

14644      BUS-B_CPD(017),          ICONSTANT (400)
14645      ADBSPH[17]_D,             ISAVE NEW
14646      P3,                        IIF SET, GOTO(COUNTER10), B.M. COUNTER OVERFLOWED
14647      NEXT, BUTA(COUNT-15-377#D[C]), IIF CLEAR, GOTO(COUNTER09), NEXT PASS THRU TEST
14648      J/COUNTER09
(4536) DCS[0.00,0.0,0.0] BM[1001..10.11..11.01..011..110...0.1,0.,0.,0...0.0000...0..1011,0...01,111...101,010,101]
14649
14650      IINTERMEDIATE WORD FOR NEXT PASS
14651      4525:
14652      COUNTER09:
14653      NEXT, J/COUNTER03          INEXT PASS
(4525) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.,0.,0...0.0000...0..0000,0...11,000...101,010,110]
14654
14655
14656
14657      | - - - - -
14658
14659      IENTER THIS SECTION WHEN HIT AN END CONDITION;
14660
14661      IENTER HERE WHEN COUNTER SIGNALS IT WAS OVERFLOWN;
14662      4527:
14663      COUNTER10:
14664      PO, DCS-CTR(C15.),          I HOLD UP
14665      NEXT, BUTR(DIC)=B,          INON TEST THAT D WAS ALSO OVERFLOWED, AT THE SAME TIME
14666      J/COUNTER10B                IIF TRUE, GOTO(COUNTER10A), ALL IS OK, END THE TEST
14667      IIF FALSE, GOTO(COUNTER10B), ERROR, COUNTER OVERFLOWED TOO SOON
(4527) DCS[0.00,1.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.,0.,0...0.0000...0..0000,0...10,011...101,011,001]
14668
14669      4533:
14670      COUNTER10A:
14671      PO, DCS-CTR(C15.),          IALL, END THE TEST WITH NO ERRORS
14672      NEXT, BUTA(RETURN),          |
14673      J/BUTERROR4                  |
(4533) DCS[0.00,1.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.,0.,0...0.0000...0..0000,0...11,111...011,111,110]
14674
14675      4531:
14676      COUNTER10B:
14677      PO, DCS-CTR(C0.),          IFORCE ERROR NOW,
14678      NEXT, BUTA(RETURN),          I B.M. COUNTER OVERFLOWED TOO SOON
14679      J/BUTERROR4                  |
(4531) DCS[0.00,1.0,0.0] BM[1111..00.00..00.00..000..000...0.0,0.,0.,0...0.0000...0..0000,0...11,111...011,111,110]
14680
14681      IENTER HERE WHEN WE GET A "D OVERFLOWED BEFORE COUNTER" CONDITION
14682      4547:
14683      COUNTER11:
14684      PO, DCS-CTR(C0.),          IFORCE ERROR NOW,
14685      NEXT, BUTA(RETURN),          I B.M. COUNTER DIDN'T OVERFLOW ON TIME
14686      J/BUTERROR4                  |
(4547) DCS[0.00,1.0,0.0] BM[1111..00.00..00.00..000..000...0.0,0.,0.,0...0.0000...0..0000,0...11,111...011,111,110]
14687
14688      IENTER HERE WHEN WE GET A "TRACKER : COUNTER" MISCOMPARE
14689      4555:
14690      COUNTER12:
14691      PO, DCS-CTR(C0.),          IFORCE ERROR NOW,

```

```

14692 NEXT, BUTA(RETURN), ! B.M. COUNTER DIDN'T INCREMENT IN STEP WITH TRACKER
14693 J/BUTERROR4 !
(4555) DCS(0.00,1.0,0.0) BM[1111.00,00.00,00.00,000.000.00,0.0,0.0,0.0,0.0,0.0000.0.0.0000.0.0.11.111.011.111,110]
14694
14695
14696
14697
14698
14699 I.PAGE=====
14700
14701
14702 .TOC * TEST610: CONDITION CODE LOGIC
14703
14704
14705
14706 |*****
14707 |*
14708 |* TESTS: 610 A - D UNWORDS: 074 + 070
14709 |*
14710 |* FUNCTIONS:
14711 |*
14712 |* THE FOLLOWING EIGHT TESTS EXERCISE THE PSW CC LOGIC (THE NZVC BITS)
14713 |* AND THE ASSOCIATED ROMS/MUXES ETC. TO VERIFY THE ABSENCE OF STUCK
14714 |* ONE/ZERO CONDITIONS ON ALL LOGIC LINES.
14715 |*
14716 |*****
14717 |
14718 | SUMMARY OF CC-LOGIC TESTS:
14719 |
14720 |
14721 | TEST ROM IR CC MODIFY PREV GENERATED
14722 | NUMR ADR DATA CLASS BYTE-H [DIC]/D-REG A-SIDE B-SIDE VBIT-H NZVC NZVC
14723 | --- -- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
14724 | A1 665 105200 010 1 1-100000 177700 100100 1 1010 0101
14725 | A2 145 105200 010 1 0-100000 077600 000200 0 1001 0110
14726 |
14727 | R1 132 105300 101 1 1-000200 100000 000200 0 0100 1011
14728 | B2 253 005300 101 0 0-000000 100200 000200 1 1001 0110
14729 |
14730 | C1 437 072000 111 0 1-000000 000200 000200 0 1010 0111
14731 | C2 037 072000 111 0 1-077776 037777 037777 0 1100 0001
14732 |
14733 | D1 216 072000 111 0 0-177400 077600 077600 1 0101 1000
14734 | D2 116 072000 111 0 0-100000 000200 100200 0 0101 1000
14735 |
14736 |
14737
14738
14739 | - - - - -
14740
14741 |*** TEST 610A1 ***
14742
14743 5627:
14744 TEST610A1:

```

```
14745      PO,      LOAD=ENUA(ZTARGET405),      INZVC AFTER = "0101"
14746      LOAD=ERROR(TEST610A1),      ERROR DIRECTORY KEY
14747      DCS=CTR(C15.),      COMPARE AT TARGET
14748      BUMP=VERIFY,      COUNT
14749      NEXT,      J/LOADIR610A1
(5627) DCS(1.00,1.0,0.1) BM(0000.00,11.11.00.000.101...0.0,0.0,0.0,0.0000...0.0000,0...11,000...101,011,010)
14750
14751      5332:
14752      LOADIR610A1:
14753      PO,      RUMP=VERIFY,      COUNT
14754      P2=U,      IP_EMIT, EMIT/105200,      (105200)INCB, CC-CLASS="010", BYTE-H
14755      NEXT,      J/LOAD01-610A1
(5332) DCS(0.00,0.0,0.1) BM(1000.00,10.10.10.000.000...0.0,0.0,0.0,1.1010...0.0000,0...11,000...011,000,110)
14756
14757      5306: 1(FREE)
14758      LOAD01-610A1:
14759      PO,      BUMP=VERIFY,      COUNT
14760      P3,      CSPD(01)_EMIT, EMIT/100012,      FOR DIC(=1), PS(INZVC)="010" PREVIOUSLY
14761      NEXT,      J/LOAD05-610A1
(5306) DCS(0.00,0.0,0.1) BM(1000.10,00.00.00.001.010...0.0,0.0,0.0,0.1110...1.0000,0...11,000...011,000,111)
14762
14763      5307: 1(FREE)
14764      LOAD05-610A1:
14765      P3,      CSPD(05)_EMIT, EMIT/177700,      A-SIDE DATA
14766      NEXT,      J/LOAD06-610A1
(5307) DCS(0.00,0.0,0.0) BM(1111.10,11.11.11.000.000...0.0,0.0,0.0,0.1010...1.0000,0...11,000...011,001,000)
14767
14768      5310: 1(FREE)
14769      LOAD06-610A1:
14770      P3,      CSPD(06)_EMIT, EMIT/100100,      B-SIDE DATA
14771      NEXT,      J/PSCC-DC610A1
(5310) DCS(0.00,0.0,0.0) BM(1000.10,00.00.01.000.000...0.0,0.0,0.0,0.1001...1.0000,0...11,000...011,001,001)
14772
14773      5311: 1(FREE)
14774      PSSC-DC610A1:
14775      SETUP,      RETURN/SETBUSA610A1,      EXEC SUBR WHICH:
14776      NEXT,      CALL(SETUPPSSC#DC)      (1) CSP(10)<310> -> PS(INZVC)
14777      (5311) DCS(0.00,0.0,0.0) BM(0101.00,01.10,01.010.111...0.0,0.0,0.0,0.0000...0.0000,0...11,100...001,101,110)
14778
14779      5312: 1(FREE)
14780      SETBUSA610A1:
14781      P2=T,      SR_CSPD(D05),      GET CONSTANT FOR A-SIDE WHEN CC SET
14782      NEXT,      J/DOIT610A1
(5312) DCS(0.00,0.0,0.0) BM(1010.10,00.00.00.000.000...0.0,1.0,0.0,0.1010...0.0000,0...11,000...011,001,011)
14783
14784      5313: 1(FREE)
14785      DOIT610A1:
14786      SETUP,      SET=CC,      FOR CLOCKING CC=5 IN NEXT WORD
14787      P2=T,      MODIFY=VBIT,      EXTRA CC ROM INPUT
14788      BUS=A=SR,      D=A=PLUS=B, SAVE=D(C),      D=(100000), D(C)=1
14789      BUS=B_CSPD(D06),      A=(177700)
14790      B=(100100)
```

```
14791      NEXT,      J/GETIT610A1
(5313) DCS(0.00,0.0,0.0) BM(1001.10,00.00.10.000.111...0.1,0.0,1.1,0.1001...0.0000,0...11,000...011,001,100)
14792
14793      5314: 1(FREE)
14794      GETIT610A1:
14795      P2=T,      CLK=CC,      PS(INZVC) GENERATED ABOVE LATCHED HERE
14796      SETUP,      RETURN/TEST610A2,      EXEC SUBR WHICH:
14797      NEXT,      CALL(PSCCTOSR3=0)      (1) PS<310> -> SR<310>, J/BUT(SR3=0)
(5314) DCS(0.00,0.0,0.0) BM(1010.00,11.00,10.010.111...0.0,0.0,0.0,0.0000...0.0000,0...11,100...001,110,010)
14798
14799
14800
14801
14802
14803      *** TEST 610A2 ***
14804
14805      5622:
14806      TEST610A2:
14807      PO,      LOAD=ENUA(ZTARGET406),      INZVC AFTER = "0110"
14808      LOAD=ERROR(TEST610A2),      ERROR DIRECTORY KEY
14809      DCS=CTR(C13.),      COMPARE AT TARGET
14810      NEXT,      J/LOAD06-610A2
(5627) DCS(1.00,1.0,0.0) BM(0010.00,11.11.00.000.110...0.0,0.0,0.0,0.0000...0.0000,0...11,000...011,001,101)
14811
14812      5315: 1(FREE)
14813      LOAD06-610A2:
14814      PO,      BUMP=VERIFY,      COUNT
14815      P3,      CSPD(06)_EMIT, EMIT/077600,      A-SIDE DATA
14816      NEXT,      J/LOAD01-610A2
(5315) DCS(0.00,0.0,0.1) BM(0111.10,11.11.10.000.000...0.0,0.0,0.0,0.1001...1.0000,0...11,000...011,001,110)
14817
14818      5316: 1(FREE)
14819      LOAD01-610A2:
14820      P3,      CSPD(01)_EMIT, EMIT/000011,      FOR DIC(=0), PS(INZVC)="1001" PREVIOUSLY
14821      NEXT,      J/PSCC-DC610A2
(5316) DCS(0.00,0.0,0.0) BM(0000.10,00.00.00.001.001...0.0,0.0,0.0,0.1110...1.0000,0...11,000...011,001,111)
14822
14823      5317: 1(FREE)
14824      PSSC-DC610A2:
14825      SETUP,      RETURN/SETBUSA610A2,      EXEC SUBR WHICH:
14826      NEXT,      CALL(SETUPPSSC#DC)      (1) CSP(10)<310> -> PS(INZVC)
14827      (5317) DCS(0.00,0.0,0.0) BM(0101.00,01.10,10.000.111...0.0,0.0,0.0,0.0000...0.0000,0...11,100...001,101,110)
14828
14829      5320: 1(FREE)
14830      SETBUSA610A2:
14831      P2=T,      SR_CSPD(D06),      GET CONSTANT FOR A-SIDE WHEN CC SET
14832      NEXT,      J/DOIT610A2
(5320) DCS(0.00,0.0,0.0) BM(1010.10,00.00.00.000.000...0.0,1.0,0.0,0.1001...0.0000,0...11,000...011,010,001)
14833
14834      5321: 1(FREE)
14835      DOIT610A2:
```

```

14836          SETUP,          SETUP-CC          IFOR CLOCKING CC=8 IN NEXT UNWORD
14837          NOT-MODIFY-VBIT,  EXTRA CC ROM INPUT
14838          P2-T,             D_A=PLUS-B, SAVE-D[C], IDW(100000), D[C]=0)
14839          BUS-A-SR,          BUS-B-SR,          IS=(077600)
14840          BUS-B-SRPHI(C000200), IS=(000200)
14841          NEXT,             J/GETIT610A2
(5321) DCS[0.00,0.0,0.0] BM[1001.01,11.00,00.010.111...0.1,0.0.1...0.0000...0.0000,0...11,000...011,010,010]
14842          5322: 1(FREE)
14843          GETIT610A2:
14844          P2-T,             CLK=CC,             !PS[NEVC] GENERATED ABOVE LATCHED HERE
14845          SETUP,          RETURN/SCOPE610A,      !EXEC SUBR WHICH:
14846          NEXT,          CALL(PSCCT0SR3=0)        ! PS[SR3] => SR43[0], J/BUT(SR3=0)
14847          (5322) DCS[0.00,0.0,0.0] BM[0101.00,01.10,10.011.111...0.0,0.0.0...0.0000...0.0000,0...11,100...001,110,010]
14848
14849          5323: 1(FREE)
14850          SCOPE610A:
14851          PO,              BUMP-VERIFY,          !COUNT
14852          NEXT,          BUTD[SCOPE],            !NO ERROR: "TEST610B1" (+1, WORDS)
14853          J/TEST610B1      ! ERROR: "LOADIR610A1" (-15, WORDS)
14854          (5323) DCS[0.00,0.1,0.1] BM[0000.00,00.00,00.000.000...0.0,0.0.0...0.0000...0.0000,0...11,000...101,011,011]
14855
14856          1 - - - - -
14857
14858          !*** TEST 610B1 ***
14859
14860          5331:
14861          TEST610B1:
14862          PO,              LOAD=ENUA(ZTARGET413), !NEVC AFTER "1011"
14863          LOAD=ERROR(TEST610B1),                !ERROR DIRECTORY KEY
14864          DCS=CTR(C13.),                        !COMPARE AT TARGET
14865          NEXT,          J/LOADIR610B1
14866          (5331) DCS[1.00,1.0,0.0] BM[0010.00,11.11,00.001.011...0.0,0.0.0...0.0000...0.0000,0...11,000...101,011,100]
14867
14868          5334:
14869          LOADIR610B1:
14870          PO,              BUMP-VERIFY,          !COUNT
14871          P2-U,             IR_EMIT, EMIT/105300, !!(105300)=DECB, CC=CLASS="101", BYTE-H
14872          NEXT,          J/LOAD01-610B1
14873          (5334) DCS[0.00,0.0,0.1] BM[1000.00,10.10,11.000.000...0.0,0.0.0...1.1010...0.0000,0...11,000...011,010,100]
14874
14875          5334: 1(FREE)
14876          LOAD01-610B1:
14877          PO,              CSPH[0]_EMIT, EMIT/100004, !FOR D[C]=1, PS[NEVC]="0100" PREVIOUSLY
14878          P2,
14879

```

```

14884      NEXT,      J/PRCC=DC610B1
(5324) DCS[0.00,0.0,0.0] BM[1000..10.00..00.00..0000..100..0.0,0.0,0.0...0.110...1..0000,0...11,000...011,010,101]

14885      5325: 1(FREE)
14886      PRCC=DC610B1
14888      SETUP,     RETURN/SETBUSA610B1,
14889
14890      NEXT,      CALL[SETUPPRCC=DC]
(5325) DCS[0.00,0.0,0.0] BM[0101..00.01..10.10..110..111..0.0,0.0,0.0...0.0000..0..0000,0...11,100...001,101,110]

14891      5326: 1(FREE)
14893      SETBUSA610B1
14894      P2-T,      SR_BSPHI(C100000),
14895      NEXT,      J/DOIT610B1
(5326) DCS[0.00,0.0,0.0] BM[1010..01.11..00.00..001..000..0.0,0.1,0.0...0.0000..0..0000,0...11,100...011,010,111]

14896      5327: 1(FREE)
14898      DOIT610B1
14899      SETUP,     SET-CC,
14900      NOT-MODIFY-VBIT,
14901      P2-T,      D_NOT-A-AND-B, SAVE=D[C],
14902      BUS-A-SR,   D=CC,
14903      BUS-B-BSPHI(C000200),
14904      NEXT,      J/GETIT610B1
(5327) DCS[0.00,0.0,0.0] BM[0010..01.11..00.00..010..111..0.0,1.0,0.1...0.0000..0..0000,0...11,100...011,011,000]

14905      5330: 1(FREE)
14907      GETIT610B1
14908      P2-T,      CLK-CC,
14909      SETUP,     RETURN/TEST610B2,
14910      NEXT,      CALL[PRCCCTOSR3=0]
(5330) DCS[0.00,0.0,0.0] BM[0101..00.11..00.01..010..111..0.0,0.0,0.0...0.0000..0..0000,0...11,100...001,110,010]

14911      1 - - - - -
14912
14913
14914
14915
14916      *** TEST 610B2 ***
14917
14918      5612:
14919      TEST610B2:
14920      PO,        LOAD=ENUA(ZTARGET406),
14921      LOAD=ERRR(TEST610B2),
14922      DCS=CTR(C14.),
14923      NEXT,      J/LOADR610B2
(5612) DCS[1.00,1.0,0.0] BM[0001..00.11..11.00..000..110..0.0,0.0,0.0...0.0000..0..0000,0...11,100...011,011,001]

14924      5331: 1(FREE)
14926      LOADR610B2:
14927      PO,        RUMP-VERIFY,
14928      P2-U,      IR_EMIT, EMIT/005300,
14929      NEXT,      J/LOAD07-610B2
(5331) DCS[0.00,0.0,0.0] BM[0000..00.11..11.00..000..110..0.0,0.0,0.0...0.0000..0..0000,0...11,100...011,011,001]

```

```
(5331) DCS[0.00,0.0,0.1] BM[0000.00,10.10,11.000.000...0.0,0.0,0.0...1.1010...0.0000,0...11.000...011.011.010]
14930
14931 5332: 1(FREE)
14932 LOAD07-610B2:
14933 P3, CSPD[07]_EMIT, EMIT/100200, |A-SIDE DATA
14934 NEXT, J/LOAD01-610B2 |
(5332) DCS[0.00,0.0,0.0] BM[1000.10,00.00,10.000.000...0.0,0.0,0.0...0.1000...1.0000,0...11.000...011.011.011]
14935
14936 5333: 1(FREE)
14937 LOAD01-610B2:
14938 P3, CSPD[01]_EMIT, EMIT/000011, |FOR D[C]=(0), PS[NZVC]="1001" PREVIOUSLY
14939 NEXT, J/PSCC-DC610B2 |
(5333) DCS[0.00,0.0,0.0] BM[0000.10,00.00,00.001.001...0.0,0.0,0.0...0.1110...1.0000,0...11.000...011.011.100]
14940
14941 5334: 1(FREE)
14942 PSSC-DC610B2:
14943 SFTUP, RETURN/SETBUSA610B2, |EXEC SUBR WHICH:
14944 NEXT, CALL[SETUPPSSC#DC] | (1) CSP[10]<3:0> -> PS[NZVC]
14945 (5334) DCS[0.00,0.0,0.0] BM[0101.00,01.10,11.01.111...0.0,0.0,0.0...0.0000...0.0000,0...11.100...001.101.110] | (2) CSP[10]<15> -> D[C]
14946
14947 5335: 1(FREE)
14948 SFTBUSA610B2:
14949 P2-T, SP_CSPD(D07), |GET CONSTANT FOR A-SIDE WHEN CC SET
14950 NEXT, J/DOIT610B2 |
(5335) DCS[0.00,0.0,0.0] BM[1010.10,00.00,00.000.000...0.0,1.0,0.0...0.1000...0.0000,0...11.000...011.011.110]
14951
14952 5336: 1(FREE)
14953 DOIT610B2:
14954 SFTUP, SET-CC, |FOR CLOCKING CC-S IN NEXT UWORD
14955 MODIFY-VBIT, |EXTRA CC ROM INPUT
14956 P2-T, D_NOT-A-AND-B, SAVE-D[C], |D=(000000), D[C]=(0)
14957 BUS-A-SP, |A=(100000)
14958 BUS-B-BSPHI(C000200), |B=(000200)
14959 NEXT, J/GETIT610B2 |
(5336) DCS[0.00,0.0,0.0] BM[0010.01,11.00,10.010.011...0.0,1.0,0.1...0.0000...0.0000,0...11.000...011.011.111]
14960
14961 5337: 1(FREE)
14962 GETIT610B2:
14963 P2-T, CLK-CC, |PS[NZVC] GENERATED ABOVE LATCHED HERE
14964 SETUP, RETURN/SCOPE610B, |EXEC SUBR WHICH:
14965 NEXT, CALL[PSCTOSR3-0] | PS<3:0> -> SR<3:0>, J/BUT(SR3-0)
(5337) DCS[0.00,0.0,0.0] BM[0101.00,01.11,00.000.011...0.0,0.0,0.0...0.0000...0.0000,0...11.100...001.110.010]
14966
14967
14968
14969 5340: 1(FREE)
14970 SCOPE610B:
14971 P0, BUMP-VERIFY, |COUNT
14972 NEXT, BUT[SCOPE], |NO ERROR: "TEST610C1" (+1, WORDS)
14973 J/TEST610C1 | ERROR: "LOADIR610B1" (-14, WORDS)
(5340) DCS[0.00,0.1,0.1] BM[0000.00,00.00,00.000.000...0.0,0.0,0.0...0.0000...0.0000,0...11.000...101.011.101]
```

```
14974
14975
14976
14977
14978
14979
14980
14981
14982 | - - - - -
14983
14984 *** TEST 610C1 ***
14985
14986 5535:
14987 TEST610C1:
14988 P0, LOAD-ENVA(ZTARGET407), |NZVC AFTER = "0111"
14989 LOAD-ERROR(TEST610C1), |ERROR DIRECTORY KEY
14990 DCS-CTR(C13.), |COMPARE AT TARGET
14991 NEXT, J/LOADIR610C1 |
(5535) DCS[1.00,1.0,0.0] BM[0010.00,11.11,00.000.011...0.0,0.0,0.0...0.0000...0.0000,0...11.000...101.100.100]
14992
14993 5544:
14994 LOADIR610C1:
14995 P0, BUMP-VERIFY, |COUNT
14996 P2-U, IR_EMIT, EMIT/072000, |(072000)=ASH, CC-CLASS="111", NOT-BYTE-H
14997 NEXT, J/LOAD01-610C1 |
(5544) DCS[0.00,0.0,0.1] BM[0111.00,01.00,00.000.000...0.0,0.0,0.0...1.1010...0.0000,0...11.000...011.100.001]
14998
14999 5341: 1(FREE)
15000 LOAD01-610C1:
15001 P3, CSPD[01]_EMIT, EMIT/100012, |FOR D[C]=(1), PS[NZVC]="1010" PREVIOUSLY
15002 NEXT, J/PSCC-DC610C1 |
(5341) DCS[0.00,0.0,0.0] BM[1000.10,00.00,00.001.010...0.0,0.0,0.0...0.1110...1.0000,0...11.000...011.100.010]
15003
15004 5342: 1(FREE)
15005 PSSC-DC610C1:
15006 RETUP, RETURN/SETBUSA610C1, |EXEC SUBR WHICH:
15007 NEXT, CALL[SETUPPSSC#DC] | (1) CSP[10]<3:0> -> PS[NZVC]
15008 (5342) DCS[0.00,0.0,0.0] BM[0101.00,01.11,00.011.111...0.0,0.0,0.0...0.0000...0.0000,0...11.100...001.101.110] | (2) CSP[10]<15> -> D[C]
15009
15010 5343: 1(FREE)
15011 RETBUSA610C1:
15012 P2-T, SP_BSPHI(C000200), |GET CONSTANT FOR A-SIDE WHEN CC SET
15013 NEXT, J/DOIT610C1 |
(5343) DCS[0.00,0.0,0.0] BM[1010.01,11.00,00.010.000...0.0,1.0,0.0...0.0000...0.0000,0...11.000...011.100.100]
15014
15015 5344: 1(FREE)
15016 DOIT610C1:
15017 SFTUP, SET-CC, |FOR CLOCKING CC-S IN NEXT UWORD
15018 NOT-MODIFY-VBIT, |EXTRA CC ROM INPUT
15019 P2-T, D_A-XOR-B, SAVE-D[C], |D=(000000), D[C]=(1)
15020 BUS-A-SP, |A=(000200)
15021 BUS-B-BSPHI(C000200), |B=(000200)
```

```
15022 NEXT, J/GETIT610C1
(5344) DCS(0.00,0.0,0.0) BM(0110.01.11.00.00.010.111.0.0.1.0.0.0.0.0000...0.0000.0...11.000...011.100.101)
15023
15024 5345: 1(FREE)
15025 GETIT610C1:
15026 P2-T, CLK-CC,
15027 SETUP, RETURN/TEST610C2,
15028 NEXT, CALL(PSCTOSR3=0)
(5345) DCS(0.00,0.0,0.0) BM(0101.00.11.00.00.010.111.0.0.0.0.0.0.0.0000...0.0000.0...11.100...001.110.010)
15029
15030
15031
15032
15033
15034
15035 *** TEST 610C2 ***
15036
15037 5602:
15038 TEST610C2:
15039 PO, LOAD-ENUA(ZTARGET401),
15040 LOAD-ERROR(TEST610C2),
15041 DCS-CTR(C13.),
15042 NEXT, J/LOAD05-610C2
(5602) DCS(1.00,1.0,0.0) BM(0010.00.11.11.00.000.001.0.0.0.0.0.0.0.0000...0.0000.0...11.000...011.100.110)
15043
15044 5346: 1(FREE)
15045 LOAD05-610C2:
15046 PO, BUMP-VERIFY,
15047 P3, CSPD(05)_EMIT, EMIT/037777,
15048 NEXT, J/LOAD01-610C2
(5346) DCS(0.00,0.0,0.1) BM(0011.10.11.11.11.111.111.0.0.0.0.0.0.0.1010...1.0000.0...11.000...011.100.111)
15049
15050 5347: 1(FREE)
15051 LOAD01-610C2:
15052 P3, CSPD(01)_EMIT, EMIT/100014,
15053 NEXT, J/PSCC-DC610C2
(5347) DCS(0.00,0.0,0.0) BM(1000.10.00.00.00.001.100.0.0.0.0.0.0.0.1110...1.0000.0...11.000...011.101.000)
15054
15055 5350: 1(FREE)
15056 PRCC-DC610C2:
15057 SETUP, RETURN/SETBUSA610C2,
15058 NEXT, CALL(SETUPPSCC=DC)
(5350) DCS(0.00,0.0,0.0) BM(0101.00.01.11.01.001.111.0.0.0.0.0.0.0.0000...0.0000.0...11.100...001.101.110)
15059
15060 5351: 1(FREE)
15061 SETBUSA610C2:
15062 P2-T, SR_CSPD(D05),
15063 NEXT, J/DOIT610C2
(5351) DCS(0.00,0.0,0.0) BM(1010.10.00.00.00.000.000.0.0.1.0.0.0.0.1010...0.0000.0...11.000...011.101.010)
15064
15065
15066 5352: 1(FREE)
```

```
15067 DOIT610C2:
15068 SETUP, SET-CC,
15069 NOT-MODIFY-VBIT,
15070 P2-T, D_A-PLUS-R, SAVE-D(C),
15071 BUS-A_SR,
15072 BUS-B_CSPD(D05),
15073 NEXT, J/GETIT610C2
(5352) DCS(0.00,0.0,0.0) BM(1001.10.00.00.00.000.111.0.0.1.0.0.1.0.1010...0.0000.0...11.000...011.101.011)
15074
15075 5353: 1(FREE)
15076 GETIT610C2:
15077 P2-T, CLK-CC,
15078 SETUP, RETURN/SCOPE610C,
15079 NEXT, CALL(PSCTOSR3=0)
(5353) DCS(0.00,0.0,0.0) BM(0101.00.01.11.01.100.111.0.0.0.0.0.0.0.0000...0.0000.0...11.100...001.110.010)
15080
15081
15082
15083 5354: 1(FREE)
15084 SCOPE610C:
15085 PO, BUMP-VERIFY,
15086 NEXT, BUTD(SCOPE),
15087 J/TEST610D1
(5354) DCS(0.00,0.1,0.1) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000...0.0000.0...11.000...101.100.101)
15088
15089
15090
15091
15092
15093
15094
15095
15096
15097
15098 *** TEST 610D1 ***
15099
15100 5545:
15101 TEST610D1:
15102 PO, LOAD-ENUA(ZTARGET410),
15103 LOAD-ERROR(TEST610D1),
15104 DCS-CTR(C12.),
15105 NEXT, J/LOAD01-610D1
(5545) DCS(1.00,1.0,0.0) BM(0011.00.11.11.00.001.000.0.0.0.0.0.0.0.0000...0.0000.0...11.000...101.001.000)
15106
15107 5510:
15108 LOAD01-610D1:
15109 P3, CSPD(01)_EMIT, EMIT/000005,
15110 NEXT, J/PSCC-DC610D1
(5510) DCS(0.00,0.0,0.0) BM(0000.10.00.00.00.000.101.0.0.0.0.0.0.0.1110...1.0000.0...11.000...011.101.101)
15111
15112 5355: 1(FREE)
15113 PSCC-DC610D1:
15114 SETUP, RETURN/SETBUSA610D1,
15115 EXEC SUBR WHICH:
```

```

15115      NEXT, CALL(SETUPPSCC+DC1)      I(1) CSP(10)<310> -> PS(NZVC)
15116      (5355) DCS(0,00,0,0,0,0) BM(0101,00,01,11,01,110,111,00,0,0,0,0,0,0,0000,0,0,0,0,0000,0,0,11,100,0001,101,110)
15117      5356: 1(FREE)
15118      SETBUSA610D1:
15119      P2-T, SR_CSPD(D06),      IGET CONSTANT FOR A-SIDE WHEN CC SET
15120      NEXT, J/DOIT610D1
15121      (5356) DCS(0,00,0,0,0,0) BM(1010,10,00,00,00,000,000,0,0,1,0,0,0,0,1001,0,0,0000,0,0,11,000,0011,101,111)
15122      5357: 1(FREE)
15123      DOIT610D1:
15124      SETUP, SET-CC,      IFOR CLOCKING CC-S IN NEXT UNWORD
15125      MODIFY-VBIT,      IEXTRA CC ROM INPUT
15126      D_A-PLUS-B, SAVE-D(C),      ID=(177400), D(C)=0)
15127      P2-T, BUS-A-SR,      IAN(077600)
15128      BUS-B_CSPD(D06),      IBN(077600)
15129      NEXT, J/GETIT610D1
15130      (5357) DCS(0,00,0,0,0,0) BM(1001,10,00,00,10,000,0,111,0,0,1,0,0,0,1,1001,0,0,0000,0,0,11,000,0011,110,000)
15131      5360: 1(PREF)
15132      GETIT610D1:
15133      P2-T, CLK-CC,      IPS(NZVC) GENERATED ABOVE LATCHED HERE
15134      SETUP, RETURN/TEST610D2,      IEXEC SUBR WHICH:
15135      NFXT, CALL(PSCTOSR3=0)      I PS<310> -> SR<310>, J/BUT(SR3=0)
15136      (5360) DCS(0,00,0,0,0,0) BM(0101,00,11,00,10,100,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0001,110,010)
15137
15138
15139
15140      I - - - - -
15141
15142      I*** TEST 610D2 ***
15143
15144      5624:
15145      TEST610D2:
15146      PO, LOAD-ENVA(ZTARGET410),      INZVC AFTER = "1000"
15147      LOAD-FRROR(TEST610D2),      IERROR DIRECTORY KEY
15148      DCS-CTR(C11,),      ICOMPARE AT TARGET
15149      NFXT, J/PSCC-DC610D2
15150      (5624) DCS(1,00,1,0,0,0) BM(0100,00,11,11,00,001,000,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,0011,110,001)
15151      5361: 1(FREE)
15152      PSSC-DC610D2:
15153      SETUP, RETURN/SETBUSA610D2,      IEXEC SUBR WHICH:
15154      NEXT, CALL(SETUPPSCC+DC1)      I(1) CSP(10)<310> -> PS(NZVC)
15155      (5361) DCS(0,00,0,0,0,0) BM(0101,00,01,11,10,010,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0001,101,110)
15156
15157      5362: 1(FREE)
15158      SFTBUSA610D2:
15159      P2-T, SR_BSPHI(C000200),      IGET CONSTANT FOR A-SIDE WHEN CC SET

```

```

15160      NEXT, J/DOIT610D2
15161      (5362) DCS(0,00,0,0,0,0) BM(1010,01,11,00,00,010,000,0,0,1,0,0,0,0,0000,0,0,0000,0,0,11,000,0011,110,011)
15162      5363: 1(FREE)
15163      DOIT610D2:
15164      SETUP, SET-CC,      IFOR CLOCKING CC-S IN NEXT UNWORD
15165      NOT-MODIFY-VBIT,      IEXTRA CC ROM INPUT
15166      D_NOT-A-AND-B, SAVE-D(C),      ID=(100000), D(C)=0)
15167      BUS-A-SR,      IAN(000200)
15168      BUS-B_CSPD(D07),      IBN(100200)
15169      NEXT, J/GETIT610D2
15170      (5363) DCS(0,00,0,0,0,0) BM(0010,10,00,00,00,000,0,111,0,0,1,0,0,0,1,1000,0,0,0000,0,0,11,000,0011,110,100)
15171      5364: 1(FREE)
15172      GETIT610D2:
15173      P2-T, CLK-CC,      IPS(NZVC) GENERATED ABOVE LATCHED HERE
15174      SETUP, RETURN/SCOPE610D,      IEXEC SUBR WHICH:
15175      NEXT, CALL(PSCTOSR3=0)      I PS<310> -> SR<310>, J/BUT(SR3=0)
15176      (5364) DCS(0,00,0,0,0,0) BM(0101,00,01,11,10,101,111,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,0001,110,010)
15177
15178
15179      5365: 1(PREF)
15180      SCOPE610D:
15181      PO, BUSDIN_EMIT-[I],      IRESET PROC UCONS
15182      FN-CLK-IR(15=00),      I
15183      NEXT, RUTD(SCOPE),      INO ERROR: "TEST620A" (+6. WORDS)
15184      J/TEST620A      I ERROR: "LOAD01-610D1" (-10. WORDS)
15185      (5365) DCS(0,00,0,1,0,0) BM(0000,00,00,00,01,000,0,100,0,0,0,0,0,0,1,1001,0,0,0000,0,0,11,000,0011,001,001)
15186
15187
15188
15189
15190
15191
15192      I - - - - -
15193
15194      I THREE TWO SUBROUTINES ARE USED IN THE ABOVE CC-LOGIC TESTS:
15195
15196      7156: 1(FREE)
15197      SETUPPSCC+DC1:
15198      P2-T, D_CSPD(D01), D(C)=ALU15,      ISET D(C) FROM BIT<15>
15199      NEXT, J/SETUPPSCC+DC02
15200      (7156) DCS(0,00,0,0,0,0) BM(1010,10,00,00,00,000,0,100,0,0,0,0,0,0,1,1101,0,0,0000,0,0,11,000,0001,110,001)
15201
15202      7161: 1(FREE)
15203      SETUPPSCC+DC02:
15204      P2, PS(3=0)-D(3=0)-[I],      ISET PS(CC) FROM BIT<03100>
15205      BUSDIN_PS-[I],      ISETUP BUSDIN TO READ PS
15206      BUTA(RETURNS),      IAND RETURN
15207      J/BUTERROR7      IERROR IF HERE
15208      (7161) DCS(0,00,0,0,0,0) BM(1000,00,00,10,01,000,0,000,0,0,0,0,0,0,1,1011,0,0,0000,0,0,11,111,0011,111,110)

```

```
15207
15208
15209
15210
15211 7162: 1(FREE)
15212 PSCCTOSR3-0:
15213 P3, CSPD[03]_BUSDIN, IGET ENABLED PS INTO CSP
15214 NEXT, J/PSCCTOSR3-0AA
(7162) DCS[0.00,0.0,0,0] BM[0000..10.00..00.00..000..000..0.0,0..0.0..0.0..0.1100...1.0000,0...11.000...001,110,011]
15215
15216 7163: 1(FREE)
15217 PSCCTOSR3-0AA:
15218 P2-T, SR_CSPD[03], INOVE PS[CC1 TO SR<3>0>
15219 NEXT, J/PSCCTOSR3-0BB
(7163) DCS[0.00,0.0,0,0] BM[1010..10.00..00.00..000..000..0.0,0.1..0.0..0.0..0.1100...0.0000,0...11.000...001,110,100]
15220
15221 7164: 1(FREE)
15222 PSCCTOSR3-0BB:
15223 P0, BUSDIN_EMIT-[I], IRESET EMIT FOR CONSTANTS
15224 EN-CLK-IR[15-00], IAND IR LOADING
15225 NEXT, J/BUTSR3-0 IAND GO TEST THE BITS
(7164) DCS[0.00,0.0,0,0] BM[0000..00.00..00.01..000..100..0.0,0.0..0.0..0.0..1.1001...0.0000,0...11.000...010,111,110]
15226
15227
15228
15229
15230
15231 1.PAGE=====
15232
15233
15234 .TOC * TEST620-624: TESTING UBREAK AND JAMUPP
15235
15236 1*****
15237 1*
15238 1* TESTS: 620 - 624 UNWORDS: 101 + 133
15239 1*
15240 1* FUNCTIONS:
15241 1*
15242 1* THE FOLLOWING GROUP OF TEST FORCES A JAMUPP CONDITION VIA THE UBREAK FACILITY.
15243 1* TESTS ARE THEN PERFORMED ON JAMUPP RELATED SIDE EFFECTS:
15244 1*
15245 1* ALL JAM CONDITION BITS CAN BE RESET; ACTIVE BUT ROM, CSP WRITE BIT ARE CLEARED
15246 1* IN WORD WHICH JAM OCCURS; NON-INTERNAL-JAM IS SET; CUA IS LOCKED, UNLOCKED BY
15247 1* BUTA(TRACK); PREFETCH=JAM SET/CLEARED.
15248 1*
15249 1*****
15250
15251
15252
15253 1 - - - - -
15254
15255 1*** TEST 620 ***
15256
15257 1 - - - - -
```

```
15258
15259 1* PART A ***
15260 1TEST-620-A DOES A 'CLR-JAM-ERRORS' UCON-I-O FUNCTION, THEN CHECKS 'INIT JAM'=0
15261 5511:
15262 TEST620A:
15263 P0, LOAD=ENUA(ZTARGET406), IEXPECTED ADDRESS AFTER 'BUT'
15264 LOAD=ERROR(TEST620A), IERROR DIRECTORY KEY
15265 DCS-CTR(C3), ICOMPARE AT TARGET
15266 NEXT, J/BCERC620A
(5511) DCS[1.00,1.0,0,0] BM[1010..00.11..11.00..000..110...0.0,0..0.0..0.0..0.0000...0..0000,0...11.000...101,111,000]
15267
15268 5570:
15269 BCERC620A:
15270 P0, BUMP-VERIFY, ICOUNT
15271 P2, CLR-JAM-ERRORS-[I], ICLEAR OUT ERROR REGISTERS
15272 NEXT, J/GOBUT620A
(5570) DCS[0.00,0.0,0,0] BM[0100..00.00..10.00..000..000...0.0,0..0.0..0.0..1.1011...0..0000,0...11.000...011,110,110]
15273
15274 5366: 1(FREE)
15275 GOBUT620A:
15276 SETUP, RETURN/TEST620B, IRETURN TO START OF NEXT SUBTEST
15277 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7,
15278 J/BUTINITJAM IGO DO MULTIPLE 'BUT' ON INIT JAM
(5366) DCS[0.00,0.0,0,0] BM[0101..00.11..00.10..011..111...0.0,0..0.0..0.0..0.0000...0..0000,0...11.100...011,011,110]
15279
15280
15281 1 - - - - -
15282
15283 1* PART B *
15284 1TEST-620-B CHECKS THAT 'OTHER-JAM-H'=0 ALSO
15285 5623:
15286 TEST620B:
15287 P0, LOAD=ENUA(ZTARGET401), IBIT<01> CLEAR
15288 LOAD=ERROR(TEST620B), IERROR DIRECTORY KEY
15289 DCS-CTR(C3), ICOMPARE AT TARGET
15290 NEXT, J/GOBUT620B
(5623) DCS[1.00,1.0,0,0] BM[1100..00.11..11.00..000..001...0.0,0..0.0..0.0..0.0000...0..0000,0...11.000...011,110,111]
15291
15292 5367: 1(FREE)
15293 GOBUT620B:
15294 SETUP, RETURN/TEST620C, IRETURN TO START OF NEXT SUBTEST
15295 NEXT, GOTO-PAGE(7), IBUT TABLE IS ON PAGE 7
15296 J/BUTOTHERJAM IGO DO BUT ON OTHER-JAM-H SIGNAL
(5367) DCS[0.00,0.0,0,0] BM[0101..00.11..00.10..001..111...0.0,0..0.0..0.0..0.0000...0..0000,0...11.100...011,101,010]
15297
15298
15299 1 - - - - -
15300
15301 1* PART C *
15302 1TEST-620-C CHECKS THAT STATUS MUX PORT 2 (JAM REG) READS (001000) WHEN RESET
15303 1 'CLEAR-JAM-ERRORS' FUNCTION CLEARS BITS<15,13,11,8-2,0> OF JAM-REG
15304 1 'CLR-YELLOW-ZONE' FUNCTION CLEARS BIT<12> OF JAM-REG
```

```

15305 1 BIT<9> IS ACTIVE LOW, READS AS "1"
15306 1 BITS<10,1> READ "0" SINCE NO WCS PRESENT
15307 1 BIT<14> IS "0" ALWAYS
15308 5621:
15309 TEST620C:
15310 P0, LOAD=ENUA(ZTARGET402), [SETUP FOR D<16:00>=0 TEST RESULT
15311 LOAD=ERROR(TEST620C), [ERROR DIRECTORY KEY
15312 DCS-CTR(C11.), [COMPARE AT TARGET
15313 P3, BUTA(CLR-FLAG-RES-UCON), [PUT ENIT ONTO BUSDIN, CLEAR OUT I=0 UCON
15314 NEXT, J/MASK620C
(5621) DCS[1.00,1.0,0,0] BM[0100.00,11.11,00.00,010...0,0,0,0...0,0000...0,0000,0...11,010...011,111,000]
15315
15316 5370: 1(FREE)
15317 MASK620C:
15318 P0, BUMP-VERIFY, [COUNT
15319 P3, CSPD[04]_EMIT, [DON'T NEED TO MASK ANYTHING
15320 EMITC, EMIT/17777,
15321 NEXT, J/GETJAM620C
(5370) DCS[0.00,0,0,0,1] BM[1111.10,11.11,11.11...0,0,0,0...0,1011...1,0000,0...11,000...011,111,001]
15322
15323 5371: 1(FREE)
15324 GETJAM620C:
15325 SETUP, RETURN/TEST621A, [GO EXECUTE SUBR WHICH:
15326 P0, BUMP-VERIFY, [COUNT
15327 NEXT, CALL(CLRJAM0D) [ (JAMRES)-XOR-CSP(02)/(001000) -> D, BUT(D=ZERO)
(5371) DCS[0.00,0,0,0,1] BM[0101.00,11.00,01.11...0,0,0,0...0,0000...0,0000,0...11,100...010,101,101]
15328
15329
15330
15331
15332
15333 1 - - - - -
15334
15335 *** TEST 621 ***
15336 [CAUSE A MICROBREAK JAM, AND CHECK ALL THE APPROPRIATE SIGNALS ARE SET
15337
15338 1 - - - - -
15339
15340 1- PART A -
15341 [TEST-621-A CAUSES A MICROBREAK JAM AT A SPECIFIC MICROADDRESS, AND CHECKS
15342 [THAT THE MICROCODE JAMS TO LOCATION (4777) IMMEDIATELY
15343 5617:
15344 TEST621A:
15345 P0, LOAD=ENUA(4777), [SETUP JAMUPP ADDRESS
15346 LOAD=ERROR(TEST621A), [ERROR DIRECTORY KEY
15347 DCS-CTR(C9.), [COMPARE AT JAMUPP WORD
15348 P3, BUTA(CUA-TRACK), [RESET CUA TRACKING IF HASN'T BEEN
15349 NEXT, J/CSPIL621A
(5617) DCS[1.00,1,0,0,0] BM[0110.00,10.01,11.11...0,0,0,0...0,0000...0,0000,0...11,001...011,111,010]
15350
15351 5372: 1(FREE)
15352 CSPIL621A:
15353 P0, BUMP-VERIFY, [COUNT

```

```

15354 EMITC, EMIT/100377,
15355 P2-U, IR_EMIT,
15356 P3, CSPD[05]_EMIT,
15357 NEXT, J/SETBRK621A
(5372) DCS[0.00,0,0,0,1] BM[1000.10,00.00,11.11...0,0,0,0...1,1010...1,0000,0...11,000...011,111,011]
15358 [NOTE CSPADDR(05)/UCON-OPERATION BIT OVERLAP
15359
15360 5373: 1(FREE)
15361 SETBRK621A:
15362 SELECT, UCON-PROC, [PROCESSOR UCON:
15363 ENABLF, EN-CLK-UBREAK[11-00], [ FOR UBREAK REG LOAD
15364 BUSDIN_EMIT[15-00], [ AND KEEP EMIT ON BUSDIN
15365 P0, SET-UCON-CONTROL, [WRITE UCON REGISTER
15366 NEXT, J/LODBRK621A
(5373) DCS[0.00,0,0,0,0] BM[0000.01,00.00,01.00...0,0,0,0...1,1001...0,0000,0...11,000...011,111,100]
15367
15368 5374: 1(FREE)
15369 LODBRK621A:
15370 P2-T, SR_ZERO, [ZERO SR<00>, TO PREVENT SPURIOUS UBREAKS FROM GETTING THRU
15371 UBREAK_RUSDIN[11-00], [LOAD MICROBREAK REGISTER
15372 EMITML/5522, [ WITH SELECTED ADDRESS FROM EMIT
15373 NEXT, J/SETRET621A
(5374) DCS[0.00,0,0,0,0] BM[0011.00,10.11,01.01...0,0,0,0...1,1010...0,0000,0...11,000...011,111,101]
15374
15375 5375: 1(FREE)
15376 SETRET621A:
15377 P0, BUMP-VERIFY, [COUNT
15378 P3, CSPD[00]_EMIT, RETURN/TEST621B, [RETURN ADDRESS FOR AFTER JAMUPP
15379 NEXT, J/SETFLG621A
(5375) DCS[0.00,0,0,0,1] BM[0101.10,11.00,01.01...0,0,0,0...0,1111...1,0000,0...11,000...011,111,111]
15380
15381 5377: 1(FREE)
15382 SETFLG621A:
15383 P2-T, D_CSPD(D05), D[C]_0, [GET VALUES TO LOAD INTO FLAGS
15384 P3, ASPH[16]_D, [ AND SAVE IN ASP FOR COMPARE LATER
15385 NEXT, J/LODFLG621A
(5377) DCS[0.00,0,0,0,0] BM[1010.10,00.00,00.01...0,0,0,0...0,1010...0,1001,0...11,000...100,000,000]
15386
15387 5400: 1(FREE)
15388 LODFLG621A:
15389 P0, BUSDIN_EMIT-[I], [KEEP IT ON
15390 P3, BUMP-VERIFY, [COUNT
15391 P3, FLAG[8-0]_D[15-8]-[I], [ENABLE U-BREAK, FLAG 8; SET EXPLAGS FOR LATER
15392 NEXT, J/SETSR621A
(5400) DCS[0.00,0,0,0,1] BM[0000.00,00.00,01.00...0,0,0,0...1,1011...0,0000,0...11,000...100,000,001]
15393
15394 5401: 1(FREE)
15395 SETSR621A:
15396 P2-T, SR_CSPD(D05), [MAKE SR<00>=1, FOR JAMUPP EXPECTED
15397 NEXT, J/UBRK621A [ (FOR UBREAK JAM EXPECTED)
(5401) DCS[0.00,0,0,0,0] BM[1010.10,00.00,00.00...0,0,0,0...0,1010...0,0000,0...11,000...101,010,010]
15398
15399 5522:
15400 UBRK621A: [*** MICROBREAK HERE ***

```

```
15401 P3=T, D_ZERO, D[C]_ALU15, UNWORD LATCH FOR CLK-D NOT CLEARED, CLK-D SHOULD HAPPEN
15402 BUTA(CLR=FLAG-RES-UCON), JAMUPP CLEAR SHOULD ZAP ACTIVE BUT LATCH, BUTA(CLR=...) SHOULDNT H
15403
15404 P3, CSPP[05]_EMIT, EMIT/030004, [CUA GETS LOCKED WITH UADDR OF THIS UNWORD (5522) ON JAMUPP
15405 [PREFETCH=JAM(1)H GETS PREFETCH=N (=0) AT JAMUPP
15406 [LATCH HOLDING "WRCSP" BIT SHOULD GET ZAPPED ON JAMUPP,
15407 [THIS WRITE SHOULDNT HAPPEN
15408 [THIS OFF SHOULD NOT BE USED
(5522) DCS[0,00,0,0,0,0] BM[0011..10,00..00,00..000..100...1,1,0..0..0,1010...1..0000,0...11,010...110,001,101]
15409
15410 [(4777) JAMUPP001: ***COMPARE ENABLED ABOVE DONE HERE***
15411 THIS WORD TESTS SR<00>, WHICH SHOULD BE SET
15412 IF SR<00>=1, GOTO(JAMUPP002B), IF SR<00>=0, GOTO(JAMUPP003) [ERROR]
15413 P3=T, SR=D SAVE OLD D IN SR, FOR NOW
15414
15415 [(4757) JAMUPP002B: P2=T, D_CSPP(00) GET RETURN ADDRESS INTO D
15416
15417 [(4XXX) JAMUPP002C: P0, RETURN_D LOAD RETURN ADDRESS
15418 P2=T, D_SR RESTORE OLD D FROM SR
15419
15420 [(7XXX) JAMUPP002D: P2=T, SR_ZERO, ZERO OUT SR, JAMUPP8 NOW ILLEGAL
15421 NEXT, BUTA(RETURN) AND NOW RETURN
15422
15423
15424 [EXECUTE THE FOLLOWING WORD ONLY IF NO JAMUPP OCCURRED:
15425 5615:
15426 ERROR621A:
15427 P0, LOAD=ENUA(0000), [FORCE ERROR
15428 LOAD=ERROR(ERROR621A), [ERROR DIRECTORY KEY
15429 DCS=CTR(C0,), [FORCE COMPARE AT P3-T
15430 BUMP=VERIFY, [COUNT
15431 NEXT, J/TEST621B
(5615) DCS[1,00,1,0,0,1] BM[1111..00,00..00,00..000..000...0,0,0,0,0...0,0000...0,0000,0...11,000...110,001,011]
15432
15433
15434
15435
15436
15437 [* PART A *
15438 [TEST-621-A CHECKS THAT D WAS ZEROED, IE, P3 PULSE WAS SUPPRESSED IN JAM WORD,
15439 [BUT UNWORD LATCH FOR CLK-D BIT NOT ZAPPED, SO BIT WAS SAVED FOR EXECUTE LATER
15440 5613:
15441 TEST621B:
15442 P0, LOAD=ENUA(ZTARGET402), [SETUP FOR D = ZERO TEST
15443 LOAD=ERROR(TEST621B), [ERROR DIRECTORY KEY
15444 DCS=CTR(C0,), [COMPARE AT TARGET
15445 NEXT, J/GOBUT621B
(5613) DCS[1,00,1,0,0,0] BM[1100..00,11..11,00..000..010...0,0,0,0,0...0,0000...0,0000,0...11,000...100,000,010]
15446
15447 5402: 1(FREF)
15448 GOBUT621B:
15449 SETUP, RETURN/TEST621C, [RETURN TO START OF NEXT SUBTEST
15450 NEXT, GOTO=PAGE(7), [BUT TABLE ON PAGE 7
15451 J/BUTD-IS-ZERO [BUT ON D CONTENTS
```

```
(5472) DCS[0,00,0,0,0,0] BM[0101..00,11..00,01..001..111...0,0,0,0,0...0,0000...0,0000,0...11,100...011,100,001]
15451
15452
15453
15454
15455 [* PART C *
15456 [TEST-621-C CHECKS THAT 'INIT JAM' STILL OFF
15457 5611:
15458 TEST621C:
15459 P0, LOAD=ENUA(ZTARGET406), [BIT<00> CLEAR
15460 LOAD=ERROR(TEST621C), [ERROR DIRECTORY KEY
15461 DCS=CTR(C4,), [COMPARE AT TARGET
15462 NEXT, J/GOBUT621C
(5611) DCS[1,00,1,0,0,0] BM[1011..00,11..11,00..000..110...0,0,0,0,0...0,0000...0,0000,0...11,000...100,000,011]
15463
15464 5403: 1(FREF)
15465 GOBUT621C:
15466 SETUP, RETURN/TEST621D, [RETURN TO START OF NEXT SUBTEST
15467 NEXT, GOTO=PAGE(7), [BUT TABLE ON PAGE 7
15468 J/BUTMINITJAM [GO TEST INIT JAM
(5403) DCS[0,00,0,0,0,0] BM[0101..00,11..00,00..111..111...0,0,0,0,0...0,0000...0,0000,0...11,100...011,011,110]
15469
15470
15471
15472
15473 [* PART D *
15474 [TEST-621-D CHECKS THAT 'OTHER-JAM-H'=1, WAS SET FOR UBREAK
15475 5607:
15476 TEST621D:
15477 P0, LOAD=ENUA(ZTARGET403), [BIT<01> SET
15478 LOAD=ERROR(TEST621D), [ERROR DIRECTORY KEY
15479 DCS=CTR(C3,), [COMPARE AT TARGET
15480 NEXT, J/GOBUT621D
(5607) DCS[1,00,1,0,0,0] BM[1100..00,11..11,00..000..011...0,0,0,0,0...0,0000...0,0000,0...11,000...100,000,100]
15481
15482 5404: 1(FREF)
15483 GOBUT621D:
15484 SETUP, RETURN/TEST621E, [RETURN TO START OF NEXT SUBTEST
15485 P0, BUMP=VERIFY, [COUNT
15486 NEXT, GOTO=PAGE(7), [BUT TABLE ON PAGE 7
15487 J/BUTOTHERJAM [GO TEST 'OTHER-JAM-H' IS SET
(5404) DCS[0,00,0,0,0,1] BM[0101..00,11..00,00..101..111...0,0,0,0,0...0,0000...0,0000,0...11,100...011,101,010]
15488
15489
15490
15491
15492
15493 [* PART E *
15494 [TEST-621-E CHECKS UBPAK ONLY BIT SET FROM UBREAK JAMUPP IN STATUS MUX PORT 2 (JAM REC)
15495 5605:
15496 TEST621E:
15497 P0, LOAD=ENUA(ZTARGET407), [SETUP FOR D = ZERO COMPARE
15498 LOAD=ERROR(TEST621E), [ERROR DIRECTORY KEY
```

```
15498 DCS-CTR(C10.), [COMPARE AT TARGET
15499 NEXT, J/EXPEC621E
(5605) DCS(1.00,1.0,0.0) BM(0101.00.11.11.00.000.010...0.0.0.0.0...0.0000...0.0000,0...11.000...100.000,101)
15500
15501 5405: 1(FREE)
15502 EXPEC621E:
15503 P3, CSPD(02) EMIT, [WHAT WE EXPECT TO SEE IN JAM IS:
15504 EMITC, EMIT/001001, [ (001001), UBREAK H IN BIT00
15505 NEXT, J/GETJAM621E
(5405) DCS(0.00,0.0,0.0) BM(0000.10.00.10.00.000.001...0.0.0.0.0...0.1101...1.0000,0...11.000...100.000,110)
15506
15507 5406: 1(FREE)
15508 GETJAM621E:
15509 SFTUP, RETURN/TEST621F, [GO TO SUBR WHICH:
15510 NEXT, CALL(JAMTOD) [ (JAMREG)-XOR-CSP(02) -> D, BUT(D=ZERO)
(5406) DCS(0.00,0.0,0.0) BM(0101.00.11.00.00.011.111...0.0.0.0.0...0.0000...0.0000,0...11.100...010.101,110)
15511
15512
15513
15514 | - - - - -
15515
15516 [* PART F *
15517 1TEST-621-F CHECKS THAT THE RIGHT CUA WAS LOCKED, AND SHORT TERM FLAGS NOT CLEARED (IE,
15518 1 BUTA(CLR-FLAG-...) IN UBREAK WORD DIDN'T CLEAR EXFLAG1), INDICATING JAMUPP CLEAR L
15519 1 DID IN FACT ZAP THE ACTIVE BUT ROM LATCH. ALSO PREFETCH=JAM(1)H GETS PREFETCH-H=0".
15520 5603:
15521 TEST621F:
15522 PO, LOAD-ENUA(ZTARGET402), [SETUP FOR D = ZERO COMPARE
15523 LOAD-ERROR(TEST621F), [ERROR DIRECTORY KEY
15524 DCS-CTR(C10.), [COMPARE AT TARGET
15525 NEXT, J/EXPEC621F
(5603) DCS(1.00,1.0,0.0) BM(0101.00.11.11.00.000.010...0.0.0.0.0...0.0000...0.0000,0...11.000...100.000,111)
15526
15527 5407: 1(FREE)
15528 EXPEC621F:
15529 PO, BUMP-VERIFY, [COUNT
15530 P3, CSPD(02) EMIT, [(USE MASK OF ALL 1'S FROM BEFORE)
15531 EMITC, EMIT/055226, [WHAT THE CUA-EXFLAG-FOVP PORT OF HMMUX SHOULD BE
15532 NEXT, J/GETCUA621F [CUA=(5522), EXFLAG<21>="1", PREFETCH=JAM(1)H="0"
(5407) DCS(0.00,0.0,0.0) BM(0101.10.10.10.10.010.110...0.0.0.0.0...0.1101...1.0000,0...11.000...100.001,000)
15533
15534 5410: 1(FREE)
15535 GETCUA621F:
15536 SFTUP, RETURN/TEST621G, [GO TO SUBR WHICH:
15537 PO, BUMP-VERIFY, [COUNT
15538 NEXT, CALL(CUATOD) [ (CUA)-XOR-CSP(02) -> D, BUT(D=ZERO)
(5410) DCS(0.00,0.0,0.0) BM(0101.00.11.11.10.110.111...0.0.0.0.0...0.0000...0.0000,0...11.100...010.101,111)
15539
15540
15541
15542
15543
```

```
15544
15545 | - - - - -
15546
15547 [* PART G *
15548 1TEST-621-G CHECKS THAT 'PREFETCH=JAM(1)H' GOT 'PREFETCH-H'=0" AFTER JAMUPP
15549 5766:
15550 TEST621G:
15551 PO, LOAD-ENUA(ZTARGET401), [BIT01 CLEAR
15552 LOAD-ERROR(TEST621G), [ERROR DIRECTORY KEY
15553 DCS-CTR(C3), [COMPARE AT TARGET
15554 NEXT, J/GOBUT621G
(5766) DCS(1.00,1.0,0.0) BM(1100.00.11.11.00.000.001...0.0.0.0.0...0.0000...0.0000,0...11.000...100.001,001)
15555
15556 5411: 1(FREE)
15557 GOBUT621G:
15558 SETUP, RETURN/TEST621H, [RETURN TO START OF NEXT SUBTEST
15559 PO, BUMP-VERIFY, [COUNT
15560 NEXT, GOTO-PAGE(7), [BUT TABLE ON PAGE 7
15561 J/BUTPREFETCHJAM [PREFETCH=JAM(1)H IN BIT01
(5411) DCS(0.00,0.0,0.0) BM(0101.00.11.11.01.100.111...0.0.0.0.0...0.0000...0.0000,0...11.100...011.110,000)
15562
15563
15564
15565
15566 | - - - - -
15567
15568 [* PART H *
15569 1TEST-621-H CHECKS THAT CSP(05) DID NOT GET WRITTEN IN THE JAMMED WORD ABOVE,
15570 1 AND IS IN FACT, EQUAL TO THE SAVED COPY OF ITS CONTENTS, IN ASPHI(16)
15571 5754:
15572 TEST621H:
15573 PO, LOAD-ENUA(ZTARGET402), [SETUP FOR D=ZERO COMPARE
15574 LOAD-ERROR(TEST621H), [ERROR DIRECTORY KEY
15575 DCS-CTR(C4), [COMPARE AT TARGET
15576 BUMP-VERIFY, [COUNT
15577 NEXT, J/COMP621H
(5754) DCS(1.00,1.0,0.0) BM(1011.00.11.11.00.000.010...0.0.0.0.0...0.0000...0.0000,0...11.000...100.001,010)
15578
15579 5412: 1(FREE)
15580 COMP621H:
15581 P2=T, D_CSPD(05)-XOR-ASPHI(R16), [COMPARE CURRENT; SAVED, SHOULD BE SAME
15582 NEXT, GOTO-PAGE(7), [XFER
15583 J/GOBUT621H
(5412) DCS(0.00,0.0,0.0) BM(0110.10.00.11.00.011.111...0.1.0.0.0...0.1010...0.0000,0...11.100...001.110,000)
15584
15585 7160: 1(FREE)
15586
15587 GOBUT621H:
15588 SFTUP, RETURN/TEST622A, [RETURN TO START OF NEXT SUBTEST
15589 NEXT, GOTO-PAGE(7), [BUT TABLE ON PAGE 7
15590 J/BUTD-18-ZERO [TEST FOR EQUALITY
(7160) DCS(0.00,0.0,0.0) BM(0101.00.11.00.00.001.111...0.0.0.0.0...0.0000...0.0000,0...11.100...011.100,001)
15590
```

```

15591
15592
15593
15594
15595
15596
15597
15598
15599
15600
15601
15602
15603
15604
15605
15606
15607
15608
15609
15610
15611
15612
15613
15614
15615
15616
15617
15618
15619
15620
15621
15622
15623
15624
15625
15626
15627
15628
15629
15630
15631
15632
15633
15634
15635
15636
15637
15638
15639
15640
15641
15642
15643
15644
15645
15646
15647
15648
15649
15650
15651
15652
15653
15654
15655
15656
15657
15658
15659
15660
15661
15662
15663
15664
15665
15666
15667
15668
15669
15670
15671
15672
15673
15674
15675
15676
15677
15678
15679
15680
15681
15682
15683
15684
15685
15686
15687
15688
15689
15690
15691
15692
15693
15694
15695
15696
15697
15698
15699
15700
15701
15702
15703
15704
15705
15706
15707
15708
15709
15710
15711
15712
15713
15714
15715
15716
15717
15718
15719
15720
15721
15722
15723
15724
15725
15726
15727
15728
15729
15730
15731
15732
15733
15734
15735
15736
15737
15738
15739
15740
15741
15742
15743
15744
15745
15746
15747
15748
15749
15750
15751
15752
15753
15754
15755
15756
15757
15758
15759
15760
15761
15762
15763
15764
15765
15766
15767
15768
15769
15770
15771
15772
15773
15774
15775
15776
15777
15778
15779
15780
15781
15782
15783
15784
15785
15786
15787
15788
15789
15790
15791
15792
15793
15794
15795
15796
15797
15798
15799
15800
15801
15802
15803
15804
15805
15806
15807
15808
15809
15810
15811
15812
15813
15814
15815
15816
15817
15818
15819
15820
15821
15822
15823
15824
15825
15826
15827
15828
15829
15830
15831
15832
15833
15834
15835
15836
15837
15838
15839
15840
15841
15842
15843
15844
15845
15846
15847
15848
15849
15850
15851
15852
15853
15854
15855
15856
15857
15858
15859
15860
15861
15862
15863
15864
15865
15866
15867
15868
15869
15870
15871
15872
15873
15874
15875
15876
15877
15878
15879
15880
15881
15882
15883
15884
15885
15886
15887
15888
15889
15890
15891
15892
15893
15894
15895
15896
15897
15898
15899
15900
15901
15902
15903
15904
15905
15906
15907
15908
15909
15910
15911
15912
15913
15914
15915
15916
15917
15918
15919
15920
15921
15922
15923
15924
15925
15926
15927
15928
15929
15930
15931
15932
15933
15934
15935
15936
15937
15938
15939
15940
15941
15942
15943
15944
15945
15946
15947
15948
15949
15950
15951
15952
15953
15954
15955
15956
15957
15958
15959
15960
15961
15962
15963
15964
15965
15966
15967
15968
15969
15970
15971
15972
15973
15974
15975
15976
15977
15978
15979
15980
15981
15982
15983
15984
15985
15986
15987
15988
15989
15990
15991
15992
15993
15994
15995
15996
15997
15998
15999
16000
16001
16002
16003
16004
16005
16006
16007
16008
16009
16010
16011
16012
16013
16014
16015
16016
16017
16018
16019
16020
16021
16022
16023
16024
16025
16026
16027
16028
16029
16030
16031
16032
16033
16034
16035
16036
16037
16038
16039
16040
16041
16042
16043
16044
16045
16046
16047
16048
16049
16050
16051
16052
16053
16054
16055
16056
16057
16058
16059
16060
16061
16062
16063
16064
16065
16066
16067
16068
16069
16070
16071
16072
16073
16074
16075
16076
16077
16078
16079
16080
16081
16082
16083
16084
16085
16086
16087
16088
16089
16090
16091
16092
16093
16094
16095
16096
16097
16098
16099
16100
16101
16102
16103
16104
16105
16106
16107
16108
16109
16110
16111
16112
16113
16114
16115
16116
16117
16118
16119
16120
16121
16122
16123
16124
16125
16126
16127
16128
16129
16130
16131
16132
16133
16134
16135
16136
16137
16138
16139
16140
16141
16142
16143
16144
16145
16146
16147
16148
16149
16150
16151
16152
16153
16154
16155
16156
16157
16158
16159
16160
16161
16162
16163
16164
16165
16166
16167
16168
16169
16170
16171
16172
16173
16174
16175
16176
16177
16178
16179
16180
16181
16182
16183
16184
16185
16186
16187
16188
16189
16190
16191
16192
16193
16194
16195
16196
16197
16198
16199
16200
16201
16202
16203
16204
16205
16206
16207
16208
16209
16210
16211
16212
16213
16214
16215
16216
16217
16218
16219
16220
16221
16222
16223
16224
16225
16226
16227
16228
16229
16230
16231
16232
16233
16234
16235
16236
16237
16238
16239
16240
16241
16242
16243
16244
16245
16246
16247
16248
16249
16250
16251
16252
16253
16254
16255
16256
16257
16258
16259
16260
16261
16262
16263
16264
16265
16266
16267
16268
16269
16270
16271
16272
1
```

```

15639      54161: 1(FREE)
15640      PCERC622B1
15641      P2,      CLR-JAM-ERRORS=11,
15642      NEXT,    J/COBUT622B,
15643      IDO IT
(5416) DCS(0.00,0.0,0.0,0) BM(0100..00.00..10.00..0000..0000..0.0,0.0,0.0...1,1011...0..0000,0...11,000...100,001,111)
15644
15645      54171: 1(FREE)
15646      COBUT622B1
15647      SETUP,   RETURN/TEST622C,
15648      NEXT,    GOTO-PAGE(7),
15649      J/BUTOTHERJAM
(5417) DCS(0.00,0.0,0.0,0) BM(0101..00.10..11,11..101..111...0.0,0.0,0.0...0,0000...0..0000,0...11,100...011,101,010)
15650
15651      I - - - - -
15652
15653      I* PART C *
15654      TEST-622-C TESTS THAT STATUS MUX PORT 2 (JAM REG) READS (001000) WHEN RESET
15655      55751
15656      TEST622C:
15657      P0,      LOAD-ENUA(ZITARGET402),
15658      LOAD-ERROR(TEST622C),
15659      DCS-CTR(C10.),
15660      P3,      BUTA(CLR-FLAG-RES=UCON),
15661      NEXT,    J/GETJAM622C
15662      ISETUP FOR D = ZERO TEST
15663      ERROR DIRECTORY KEY
15664      ICOMPARE AT TARGET
15665      IPUT ENIT ON BUSDIN, CLEAR SHORT TERM FLAGS
15666      I
(5575) DCS(1.00,1.0,0.0,0) BM(0101..00.11..11,00..0000..010...0.0,0.0,0.0...0,0000...0..0000,0...11,010...100,010,000)
15667
15668      54201: 1(FREE)
15669      GETJAM622C:
15670      SETUP,   RETURN/TEST623,
15671      NEXT,    CALL(CLRJAMTOD)
15672      IGO TO SUBR WHICH:
15673      I (JAMREG)-XOR-CSP(02)/(001000) -> D, BUT(D=ZERO)
15674      I
(5420) DCS(0.00,0.0,0.0,0) BM(0101..00.10..11,11..011..111...0.0,0.0,0.0...0,0000...0..0000,0...11,100...010,101,101)
15675
15676      I - - - - -
15677
15678      I*** TEST 623 ***
15679      TEST-623 CHECKS THAT BUTA(CUA-TRACK) RESTARTS CUA TRACKING
15680      55731
15681      TEST623:
15682
15683      P0,      LOAD-ENUA(ZITARGET402),
15684      LOAD-ERROR(TEST623),
15685      DCS-CTR(C10.),
15686      P3,      BUTA(CUA-TRACK),
15687      NEXT,    J/EXEC623
15688      ISETUP FOR D = ZERO TEST
15689      ERROR DIRECTORY KEY
15690      ICOMPARE AT TARGET
15691      IRESET TO TRACKING CUA MODE
15692      I
(5573) DCS(1.00,1.0,0.0,0) BM(0101..00.11..11,00..0000..010...0.0,0.0,0.0...0,0000...0..0000,0...11,001...100,010,001)

```

```

15686      5421: 1(FREE)
15687      EXPEC623:
15688      PO,      BUMP-VERIFY,      !COUNT
15689      P3,      CSPD[02]_EMIT,      ! (PREV MARK)
15690      EMITC,    EMIT/073734,      ! CUA IS WORD WHICH LOADS CUA INTO CSP, ONLY
15691      NEXT,     J/GETCUA623      ! EXPL[001] CLEARED FROM BUTA(CLR=FLAG=...), ABOVE
15692      (5421) DCS[0.00,0.0,0.1] BM[0111..10.01..11.11..011..100...0.0.0.0...0.1101...1..0000,0...11,000...100,010,010]

15693      5422: 1(FREE)
15694      GETCUA623:
15695      SETUP,    RETURN/SCOPE623,      !GO TO SUBR WHICH:
15696      NEXT,     CALL[CUATOD]          ! CUA=WORD WHICH READS CUA), SINCE TRACKING RESET
15697      (5422) DCS[0.00,0.0,0.0] BM[0101..00.10..00.10..011..111...0.0.0.0...0.0000...0..0000,0...11,100...010,101,111]

15699      5423: 1(FREE)
15700      SCOPE623:
15701      PO,      BUSDIN_EMIT-[I],      !RESET PROC UCONS
15702      P3,      EM-CLK-IR[15-00],      !IR GETS JUNK, BUT DON'T CARE
15703      NEXT,     FLAG[8-0]_D[15-8]=[I], !RESET FLAG TO ALL ZERO
15704      BUTD[SCOPE], !NO ERROR: "TEST624A" (+1, WORDS)
15705      J/TEST624A ! ERROR: "SCERC620A" [-49, WORDS]
15706      (5423) DCS[0.00,0.1,0.0] BM[0000..00.00..00.01..0000...101...0.0.0.0...0.1011...0..0000,0...11,000...101,111,001]

15708      1 - - - - -
15709      *** TEST 624 ***
15710      !DO ANOTHER UBREAK JAM, CHECK JAM OCCURS, AND THAT P2 IS SEEN IN JAM WORD
15711      1 - - - - -
15712      !- PART A -
15713      !TEST-624-A CAUSES A UBREAK JAM, AND CHECKS TO SEE IT OCCURS
15714      5571:
15715      TEST624A:
15716      PO,      LOAD-ENUA(4777),      !SETUP JAMUPP ADDRESS
15717      P3,      LOAD-ERROR(TEST624A), !ERROR DIRECTORY KEY
15718      DCS-CTR(C9.), !COMPARE AT JAMUPP WORD
15719      NEXT,     J/LODIR624A
15720      (5571) DCS[1.00,1.0,0.0] BM[0110..00.10..01.11..111..111...0.0.0.0...0..0000...0..0000,0...11,000...101,110,010]

15722      5562:
15723      LODIR624A:
15724      P2=0,      IR_EMIT,      !IR PATTERN ASSERTS PREFETCH-H:
15725      P3,      EMIT/030603,      ! PREFETCH-H=(DOPH+NOVBL+SMON+SRYL+DMON+DRYL)
15726      FLAG[8-0]_D[15-8],      !FLAGS GET D, WHICH IS ZERO

```

```

15735      NEXT,     J/SETBRK624A      ! + (SOPH+NEGL+SBCL+RORL+ASRL+DMON+DRYL)
15736      (5562) DCS[0.00,0.0,0.0] BM[0011..00.00..01.10..000...011...0.0.0.0...0.1010...0..0000,0...11,000...100,010,100]

15738      5424: 1(FREE)
15739      SETBRK624A:
15740      SELECT,    UCON-PROC,      !ENABLE PROCESSOR UCON:
15741      ENABLE,    EN-CLK-UBREAK[11-00], ! EN UBREAK LOAD,
15742      PO,      BUSDIN_EMIT[15-00],      ! KEEP EMIT ON BUSDIN
15743      SET-UCON-CONTROL, !WRITE CONTROLS
15744      NEXT,     BUMP-VERIFY,      !COUNT
15745      J/LODBRK624A
15746      (5424) DCS[0.00,0.0,0.1] BM[0000..01.00..00.01..0000...000...0.0.0.0...0.1001...0..0000,0...11,000...100,010,101]

15748      5425: 1(FREE)
15749      LODBRK624A:
15750      PO,      BUMP-VERIFY,      !COUNT
15751      P2=T,      UBREAK_BUSDIN[11-00], !LOAD UBREAK REGISTER
15752      EMITML/6255, ! WITH SELECTED ADDRESS FROM EMIT
15753      NEXT,     J/SETRET624A
15754      (5425) DCS[0.00,0.0,0.1] BM[0000..00.11..00.10..101..101...0.0.0.0...0.1010...0..0000,0...11,000...100,010,110]

15756      5426: 1(FREE)
15757      SETRET624A:
15758      P3,      CSPD[00]_EMIT, RETURN/TEST624B, !RETURN ADDRESS FOR AFTER JAMUPP
15759      NEXT,     GOTO-PAGE(6),      ! RETURN TO START OF NEXT SUBTEST
15760      J/SETD624A
15761      (5426) DCS[0.00,0.0,0.0] BM[0101..10.10..11.10..101..110...0.0.0.0...0.0.1111...1..0000,0...11,100...011,000,110]

15763      6306: 1(FREE)
15764      SETD624A:
15765      P2=T,      D_CSPD(D05), D[IC]_ALU15, !SETUP D WITH VALUE TO GO INTO FLAG<8>
15766      NEXT,     J/SETFLG6241
15767      (6306) DCS[0.00,0.0,0.0] BM[1010..10.00..00.00..000...100...0.0.0.0...0.0.1010...0..0000,0...11,000...100,101,011]

15769      6453: 1(FREE)
15770      SETFLG6241:
15771      PO,      BUSDIN_EMIT-[I],      !KEEP IT ON
15772      P3,      FLAG[8-0]_D[15-8]=[I], !SET UBRAK FLAG<8>
15773      NEXT,     J/SETSR624A
15774      (6453) DCS[0.00,0.0,0.0] BM[0000..00.00..00.01..0000...001...0.0.0.0...0.1011...0..0000,0...11,000...100,101,100]

15776      6454: 1(FREE)
15777      SETSR624A:
15778      P2=T,      SR_CSPD(D05),      !SET SR<00>=1 FOR UBRAK JAMUPP EXPECTED
15779      NEXT,     J/UBRK624A
15780      (6454) DCS[0.00,0.0,0.0] BM[1010..10.00..00.00..000...000...0.0.0.0...0.0.1010...0..0000,0...11,000...010,101,101]

15782      6255:
15783      UBRR624A:
15784      P2=T,      D_ZERO, D[IC]_ALU15,
15785      1 - - - - -
15786      *** MICROBREAK HERE ***
15787      !P2 PULSE SHOULD OCCUR, SO D SHOULD BE ZEROED.
15788      !CUA GETS LOCKED WITH UADDR OF THIS UWORD (6255) ON JAMUPP
15789      !PREFETCH=JAM(1)H GETS PREFETCH-H ("1") AT JAMUPP

```

```

15780      P3,      FLAG(8-0)_D(15-8),      ZERO ALL THE FLAGS IN THIS WORD. THE UBREAK SHOULD
15781                                     STILL OCCUR, HOWEVER, AS IT LOOKS AT THE "NUA",
15782      NEXT,      J/ERROR624A      UPP SHOULD NOT BE USED.
(6255) DCS(0,00,0,0,0,0) BM(0011,00,00,00,00,000,100,0,0,0,0,0,0,0000,0,0,0,11,000,0,101,110,111)

15783
15784
15785      I(4777) JAMUPP001: ***COMPARE ENABLED ABOVE DONE HERE***
15786      THIS WORD TESTS SR<00>, WHICH SHOULD BE SET
15787      IF SR<00>=1, GOTO(JAMUPP002B), IF SR<00>=0, GOTO(JAMUPP003) (ERROR)
15788      P3-T,      SR-D      SAVE OLD D IN SR, FOR NOW
15789
15790      I(4757) JAMUPP002B: P2-T, D-CSPD(00)      GET RETURN ADDRESS INTO D
15791
15792      I(4XXX) JAMUPP002C: P0,      RETURN-D      LOAD RETURN ADDRESS
15793      P2-T,      D-SR      RESTORE OLD D FROM SR
15794
15795      I(7XXX) JAMUPP002D: P2-T, SR_ZERO, ZERO OUT SR, JAMUPPS NOW ILLEGAL
15796      NEXT,      BUTA(RETURN)      AND NOW RETURN
15797
15798
15799
15800
15801      IEXECUTE THE FOLLOWING WORD ONLY IF NO JAMUPP OCCURRED:
15802      6567:
15803      ERROR624A:
15804      P0,      LOAD-ENUA(0005),      IFORCE ERROR
15805      LOAD-ERROR(ERROR624A),      IFORCE DIRECTORY KEY
15806      DCS-CTR(C0,),      IFORCE COMPARE AT P3-T
15807      BUMP-VERIFY,      ICOUNT
15808      NEXT,      GOTO=PAGE(5),      IXFER TO 5
15809      J/TEST624B
(6567) DCS(1,00,1,0,0,1) BM(1111,00,00,00,000,101,0,0,0,0,0,0,0000,0,0,0,11,100,0,101,110,101)

15810
15811
15812      I - - - - -
15813
15814      I* PART B *
15815      ITEST-624-B CHECKS THAT D WAS ZEROED, IE, P2 DID OCCUR IN JAM WORD
15816      5565:
15817      TEST624B:
15818      P0,      LOAD-ENUA(ZTARGET434),      ISETUP FOR IR=ZERO TEST
15819      LOAD-ERROR(TEST624B),      IFORCE DIRECTORY KEY
15820      DCS-CTR(C6,),      ICOMPARE AT TARGET
15821      NEXT,      J/ZEROIR624B
(5565) DCS(1,00,1,0,0,0) BM(1001,00,11,11,00,011,100,0,0,0,0,0,0000,0,0,0,11,000,0,100,010,111)

15822
15823      5427: I(FREF)
15824      ZEROIR624B:
15825      SETUP,      RETURN/TEST624C,      IPUT THE ZEROS IN D INTO THE IR,
15826      NEXT,      CALL(DINTOIR-5)      I TO NEGATE PREFETCH-H (000000)=HALT
(5427) DCS(0,00,0,0,0,0) BM(0101,00,11,11,11,110,111,0,0,0,0,0,0000,0,0,0,11,100,0,010,111,011)

15827

```

```

15828
15829
15830
15831      I - - - - -
15832
15833      I* PART C *
15834      ITEST-624-C CHECKS THAT THE RIGHT CUA WAS LOCKED,
15835      ITHAT THE EXFLAG<2>:1>H READ AS "00", AND
15836      ITHAT PREFETCH=JAM(1)H = "1", FROM TEST-624-A, EVEN THOUGH TEST-624-B
15837      I RESET THE IR SO THAT PREFETCH-H="0"
15838      5776:
15839      TEST624C:
15840      P0,      LOAD-ENUA(ZTARGET402),      ISETUP FOR D = ZERO COMPARE
15841      LOAD-ERROR(TEST624C),      IFORCE DIRECTORY KEY
15842      DCS-CTR(C10,),      ICOMPARE AT TARGET
15843      NEXT,      J/EXPEC624C
(5776) DCS(1,00,1,0,0,0) BM(0101,00,11,11,11,00,000,010,0,0,0,0,0,0000,0,0,0,11,000,0,100,011,000)

15844
15845      5430: I(FREF)
15846      EXPEC624C:
15847      P3,      CSPD(02)_EMIT,      I(USE MASK OF ALL 1'S FROM BEFORE)
15848      EMIT/062551,      IWHAT THE CUA-EXFLAG-FOVP PORT OF HBMUX SHOULD BE
15849      NEXT,      J/GETCUA624C      ICUA=(6255), EXFLAG="00", PREFETCH=JAM(1)H="1"
(5430) DCS(0,00,0,0,0,0) BM(0110,10,01,01,01,101,001,0,0,0,0,0,0,1101,1,0,0000,0,0,0,11,000,0,100,011,001)

15850
15851      5431: I(FREF)
15852      GETCUA624C:
15853      SETUP,      RETURN/TEST624D,      IGO TO SUBR WHICH:
15854      P0,      BUMP-VERIFY,      ICOUNT
15855      NEXT,      CALL(CUAT00)      I (CUA)-XOR-CSP(02) -> D, BUT(D=ZERO)
(5431) DCS(0,00,0,0,0,0) BM(0101,00,11,11,01,110,111,0,0,0,0,0,0,0000,0,0,0,11,100,0,010,101,111)

15856
15857
15858
15859      I - - - - -
15860
15861      I* PART D *
15862      ITEST-624-D CHECKS THAT "PREFETCH=JAM(1)H" GOT "PREFETCH-H"="1" AFTER JAMUPP
15863      5761:
15864      TEST624D:
15865      P0,      LOAD-ENUA(ZTARGET403),      IBIT01 SET
15866      LOAD-ERROR(TEST624D),      IFORCE DIRECTORY KEY
15867      DCS-CTR(C3,),      ICOMPARE AT TARGET
15868      NEXT,      J/GOBUT624D
(5761) DCS(1,00,1,0,0,0) BM(1100,00,11,11,00,000,011,0,0,0,0,0,0,0000,0,0,0,11,000,0,100,011,010)

15869
15870
15871      5432: I(FREF)
15872      GOBUT624D:
15873      SETUP,      RETURN/CLEAR624,      IRETURN TO SCOPE LOOP TEST WORD
15874      NEXT,      GOTO=PAGE(7),      IBUT TABLE ON PAGE 7
15875      J/BUTPREFETCHJAM      IPREFETCH=JAM(1)H IN BIT01
(5432) DCS(0,00,0,0,0,0) BM(0111,00,00,11,10,101,111,0,0,0,0,0,0,0,0000,0,0,0,11,100,0,011,110,000)

```

```

15875
15876
15877 7165: 1(FREE)
15878 CLEAR624:
15879 SETUP, RETURN/SCOPF624, IRETURN FOR SCOPE LOOP TEST
15880 NEXT, CALL/CLEAR-I=0-B) ICLEAR OUT WHAT I'VE DONE
(7165) DCS[0.00.0.0.0.0] BM[0101..00.01..00.10..100..111..0.0.0..0..0.0000...0..0000.0...11.100...010.011.000]

15881
15882
15883 5224: 1(FREF)
15884 SCOPE624:
15885 P0, BUMP-VERIFY, ICOUNT
15886 BUSDIN_EMIT[I], IRESET PROC UCON
15887 EN-CLK-IR[15-00], I
15888 P3, BUTA(CUA-TRACK), IAND START CUA TRACKING AGAIN
15889 NEXT, BUTD(SCOPE), INO ERROR: "TEST701A" [+1, WORDS]
15890 J/TEST701A I ERROR: "SETIR624A" [-18, WORDS]
(5224) DCS[0.00.0.1.0.1] BM[0000..00.00..00.01..000..100...0.0.0..0..0..1.1001...0..0000.0...11.001...101.110.011]

15891
15892
15893
15894
15895
15896
15897
15898 1.PAGE=====
15899
15900
15901 .TDC * TEST701-702: LOAD/READ THE BA, FULL 18, BITS
15902
15903 I=====
15904 I*
15905 I* TESTS: 701A - 702B UWORDS: 033 + 042
15906 I*
15907 I* FUNCTIONS:
15908 I*
15909 I* THIS NEXT SET OF SIX TESTS CHECKS THE "BA" REGISTER RELATED FUNCTIONS:
15910 I* SPECIFICALLY,
15911 I* 1) LOAD/READ OF BA<15:00>, FROM BUS-A -> BA -> STATUS-MUX/PBA-PORT
15912 I* 2) BUT ON "BA<00>"
15913 I* 3) LOAD READ OF BA<17:16>, FROM BUS-B -> BA -> STATUS-MUX/SERVICE-PORT
15914 I* THE EFFECT OF PARTICULAR RT-UCON-FUNCTIONS/CONSOLE-18,-BIT-MODE
15915 I* IN THIS SITUATION IS ALSO EXAMINED FOR FUNCTIONALITY.
15916 I*
15917 I=====
15918
15919
15920
15921 I - - - - -
15922
15923 ITFST-701-A CHECKS THAT BA<15:00> CAN BE LOADED, AND READ BACK WITH (052525)
15924 5563:
15925 TEST701A:
15926 P0, LOAD-ENUA(ZTARGET402), ISETUP FOR D=ZERO COMPARE

```

```

15927 LOAD-ERROR(TEST701A), IERROR DIRECTORY KEY
15928 DCS-CTR(C12.), ICOMPARE AT TARGET
15929 BUMP-VERIFY, ICOUNT
15930 J/MASK701A I
(5563) DCS[1.00.1.0.0.1] BM[0011..00.11..11.00..000..010...0.0.0..0..0..0.0000...0..0000.0...11.000...100.111.110]

15931
15932 5476:
15933 MASK701A:
15934 P3, CSPD[04]_EMIT, EMIT/177777, IMASK TO READ ALL THE BITS
15935 NEXT, GOTO-PAGE(7), IXFER
15936 J/EXPECT701A I
(5476) DCS[0.00.0.0.0.0] BM[1111..10.11..11.11..111..111...0.0.0..0..0..0.1011...1..0000.0...11.100...001.110.110]

15937
15938 7166: 1(FREF)
15939 EXPECT701A:
15940 P3, CSPD[02]_EMIT, EMIT/052525, IEXPECTED DATA TO BE READ OUT OF PBA AFTER LOAD;
15941 NEXT, J/LOADBA701A I"0101 0101 0101 0101"
(7166) DCS[0.00.0.0.0.0] BM[0101..10.01..01.01..010..101...0.0.0..0..0..0.1101...1..0000.0...11.000...001.111.000]

15942
15943 7170: 1(FREF)
15944 LOADBA701A:
15945 P1, BA_ASPHI(C052525), ILOAD BA<15:00> WITH PATTERN
15946 NEXT, J/GOTEST701A I
(7170) DCS[0.00.0.0.0.0] BM[0000..00.00..11.01..111..000...0.0.0..1..0...0.0000...0..0000.0...11.000...001.111.001]

15947
15948 7171: 1(FREF)
15949 GOTEST701A:
15950 SETUP, RETURN/TEST701B, IGO TO SUBR THAT:
15951 NEXT, CALL(PBATOD) I (PBA)-XOR-CSP(02)->D, BUT(D-IS-ZERO)
(7171) DCS[0.00.0.0.0.0] BM[0101..00.10..10.10..011..111...0.0.0..0..0..0.0000...0..0000.0...11.100...010.101.001]

15952
15953
15954
15955
15956
15957
15958 I - - - - -
15959
15960 ITFST-701-B CHECKS THAT BA<00> WAS LOADED WITH A (1)
15961 5523:
15962 TEST701B:
15963 P0, LOAD-ENUA(ZTARGET403), IRT<00> SET
15964 LOAD-ERROR(TEST701B), IERROR DIRECTORY KEY
15965 DCS-CTR(C1.), ICOMPARE AT TARGET
15966 J/GOBUT701A I
(5523) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..000..011...0.0.0..0..0..0.0000...0..0000.0...11.000...100.011.100]

15967
15968 5434: 1(FREF)
15969 GOBUT701A:
15970 SETUP, RETURN/TEST701C, IRETURN TO START OF NEXT SUBTEST
15971 NEXT, GOTO-PAGE(7), IBUT TABLE
15972 J/BUTBA00 IBA<00>M IN BIT<00>

```

```
(5434) DCS[0.00,0.0,0.0] BM[0110..00.11..11.11..000..011..0.0,0.0,0.0,0.0000...0.0000,0...11,100...011,101,001]
15973
15974
15975
15976
15977
15978
15979 |TEST-701-C CHECKS THAT BA<15100> CAN BE LOADED, AND READ BACK WITH (125252)
15980 47701
15981 TEST701C:
15982 PO, LOAD-ENUA(ZTARGET402), |SETUP FOR D=ZERO COMPARE
15983 LOAD-ERROR(TEST701C), |ERROR DIRECTORY KEY
15984 DCS-CTR(C11.), |COMPARE AT TARGET
15985 NEXT, J/EXPECT701C
(6470) DCS[1.00,1.0,0.0] BM[0100..00.11..11.00..000..010..0.0,0.0,0.0,0.0000...0.0000,0...11,000...100,101,101]
15986
15987 6455: 1(FREE)
15988 EXPECT701C:
15989 P3, CSPD[02]_EMIT, EMIT/125252, |EXPECTED DATA TO BE READ OUT OF PBA AFTER LOAD;
15990 NEXT, J/LOADBA701C |"1010 1010 1010 1010"
(6455) DCS[0.00,0.0,0.0] BM[0101..10.10..10.10..101..010..0.0,0.0,0.0,0.1101...1.0000,0...11,000...100,101,110]
15991
15992 6456: 1(FREE)
15993 LOADBA701C:
15994 P1, BA_ASPI(C125252), |LOAD BA<15100> WITH PATTERN
15995 NEXT, J/GOTEST701C
(6456) DCS[0.00,0.0,0.0] BM[0000..00.00..11.01..110..000...0.0,0.0,1.0,0.0000...0.0000,0...11,000...100,101,111]
15996
15997 6457: 1(FREE)
15998 GOTEST701C:
15999 SETUP, RETURN/TEST701D, |GO TO SUBP THAT:
16000 NEXT, CALL(PBATOD) | (PBA)-XOR-CSP(02)->D, BUT(D-IS-ZERO)
(6457) DCS[0.00,0.0,0.0] BM[0101..00.10..10.01..010..111..0.0,0.0,0.0,0.0000...0.0000,0...11,100...010,101,001]
16001
16002
16003
16004
16005
16006
16007 | - - - - -
16008
16009 |TEST-701-D CHECKS THAT BA<00> WAS LOADED WITH A (0)
16010 5512:
16011 TEST701D:
16012 PO, LOAD-ENUA(ZTARGET402), |BIT<00> CLEAR
16013 LOAD-ERROR(TEST701D), |ERROR DIRECTORY KEY
16014 DCS-CTR(C3.), |COMPARE AT TARGET
16015 NEXT, J/GOBUT701D
(5512) DCS[1.00,1.0,0.0] BM[1100..00.11..11.00..000..010..0.0,0.0,0.0,0.0000...0.0000,0...11,000...100,011,101]
16016
16017 5435: 1(FREE)
16018 GOBUT701D:
16019 SETUP, RETURN/SCOPE701, |RETURN TO SCOPE LOOP TEST WORD
```

```
16020 NEXT, GOTO-PAGE(7), |BUT TABLE
16021 J/BUTBA00 |BA<00> IN BIT<00>
(5435) DCS[0.00,0.0,0.0] BM[0101..00.10..00.11..110..111..0.0,0.0,0.0,0.0000...0.0000,0...11,100...011,101,001]
16022
16023
16024
16025
16026 5436: 1(FREE)
16027 SCOPE701:
16028 PO, BUMP-VERIFY, |COUNT
16029 NEXT, BUTD(SCOPE), |NO ERROR: "TEST702A" (+1, WORDS)
16030 J/TEST702A |ERROR: "MASK701A" (-12, WORDS)
(5436) DCS[0.00,0.1,0.1] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0,0.0000...0.0000,0...11,000...100,111,111]
16031
16032
16033
16034
16035 | - - - - -
16036
16037 |TEST-702-A CHECKS THAT BA<17:16> CAN BE LOADED, AND READ BACK WITH "01"
16038 | WHEN IN 18, RIT CONSOLE MODE AND KT-NO-RELOCATE MODE
16039 5477:
16040 TEST702A:
16041 PO, LOAD-ENUA(ZTARGET402), |SETUP FOR D=ZERO COMPARE
16042 LOAD-ERROR(TEST702A), |ERROR DIRECTORY KEY
16043 DCS-CTR(C13.), |COMPARE AT TARGET
16044 NEXT, J/MASK702A
(5477) DCS[1.00,1.0,0.0] BM[0010..00.11..11.00..000..010..0.0,0.0,0.0,0.0000...0.0000,0...11,000...100,111,100]
16045
16046 5474:
16047 MASK702A:
16048 P3, CSPD[04]_EMIT, EMIT/001400, |MASK TO READ ONLY BITS<9:8>
16049 NEXT, J/EXPECT702A
(5474) DCS[0.00,0.0,0.0] BM[0000..10.00..11.00..000..000...0.0,0.0,0.0,0.1011...1.0000,0...11,000...100,011,111]
16050
16051 5437: 1(FREE)
16052 EXPECT702A:
16053 P3, CSPD[02]_EMIT, EMIT/000400, |EXPECTED DATA TO BE READ OUT OF SERVICE<9:8> AFTER LOAD;
16054 NEXT, J/SETLED702A |"0000 0001 0000 0000"
(5437) DCS[0.00,0.0,0.0] BM[0000..10.00..01.00..000..000...0.0,0.0,0.0,0.1101...1.0000,0...11,000...100,100,001]
16055
16056 5441: 1(FREE)
16057 SETLED702A:
16058 P3, SET-CONSOLE-LED, |ENTER INTO 18, BIT MODE FOR PBA<17:16> READ
16059 NEXT, J/SETKT702A
(5441) DCS[0.00,0.0,0.0] BM[0100..00.00..00.00..100..001...0.0,0.0,0.0,0.1101...1.0000,0...11,000...100,100,010]
16060
16061 5442: 1(FREE)
16062 SETKT702A:
16063 PO, BUMP-VERIFY, |COUNT
16064 BUDIN_SERVICE[-I], |READ SERVICE PORT BITS<9:8>
16065 KT-NO-RELOCATE[-I], |SETUP KT FOR BA<17:16> LOADABILITY
16066 NEXT, J/LOADBA702A
```

```
(5442) DCS(0,00,0,0,0,0,1) BM(0101,01,00,00,00,000,010,0,0,0,0,0,0,1,1001,0,0,0000,0,0,11,000,100,100,011)
16067
16068 5443: 1(FREE)
16069 LOADBA702A:
16070 P1, BA_BSPhi(C052525), |LOAD BA<17:16> WITH PATTERN "01"
16071 NEXT, J/GOTEST702A
16072 (5443) DCS(0,00,0,0,0,0,1) BM(0000,01,11,00,00,111,000,0,0,0,0,1,0,0,0,0000,0,0,0000,0,0,11,000,100,100,100)
16072
16073 5444: 1(FREE)
16074 GOTEST702A:
16075 SETUP, RETURN/TEST702B, |GO TO SUBR THAT:
16076 NEXT, CALL(GETPROCDA) | (SERVICE<9:8>)-XOR-CSP(02)->D, BUT(D-IS-ZERO)
16077 (5444) DCS(0,00,0,0,0,0,1) BM(0110,00,11,11,10,001,111,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,011,111,011)
16077
16078
16079
16080
16081
16082
16083
16084
16085
16086 1TEST-702-B CHECKS THAT BA<17:16> CAN BE LOADED, AND READ BACK WITH "10"
16087 1 WHEN IN 18, BIT CONSOLE MODE AND KT=NO-RELOCATE MODE
16088 6761:
16089 TFS702B:
16090 PO, LOAD-ENVA(2TARGET402), |SETUP FOR D=ZERO COMPARE
16091 P1, LOAD-ERROR(702B), |ERROR DIRECTORY KEY
16092 DCS-CTR(C11,), |COMPARE AT TARGET
16093 NEXT, J/EXPEC702B
16094 (6761) DCS(1,00,1,0,0,0,1) BM(0100,00,11,11,00,000,010,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,100,110,000)
16094
16095 6460: 1(FREE)
16096 EXPEC702B:
16097 P1, CSPD(02)_EMIT, EMIT/001000, |EXPECTED DATA TO BE READ OUT OF SERVICE<9:8> AFTER LOAD:
16098 NEXT, J/STKT702B |"0000 0010 0000 0000"
16099 (6460) DCS(0,00,0,0,0,0,1) BM(0000,10,00,10,00,000,000,0,0,0,0,0,0,0,1101,1,0,0000,0,0,11,000,100,110,001)
16099
16100 6461: 1(FREE)
16101 STKT702B:
16102 PO, BUSDIN_SERVICE(-1), |READ SERVICE PORT BITS<9:8>
16103 P1, KT=NO-RELOCATE(-7), |SETUP KT FOR BA<17:16> LOADABILITY
16104 NEXT, J/LOADBA702B
16105 (6461) DCS(0,00,0,0,0,0,1) BM(0101,01,00,00,00,000,010,0,0,0,0,0,0,1,1001,0,0,0000,0,0,11,000,100,110,010)
16105
16106 6462: 1(FREE)
16107 LOADBA702B:
16108 PO, BUMP-VERIFY, |COUNT
16109 P1, BA_BSPhi(C125252), |LOAD BA<17:16> WITH PATTERN "10"
16110 NEXT, J/GOTEST702B
16111 (6462) DCS(0,00,0,0,0,0,1) BM(0000,01,11,00,00,110,000,0,0,0,0,1,0,0,0,0000,0,0,0000,0,0,11,000,100,110,011)
16111
16112 6463: 1(FREE)
```

```
GOTEST702B:
16113 SETUP, RETURN/SCOPE702, |GO TO SUBR THAT:
16114 NEXT, CALL(GETPROCDA) | (SERVICE<9:8>)-XOR-CSP(02)->D, BUT(D-IS-ZERO)
16115 (6463) DCS(0,00,0,0,0,0,1) BM(0101,00,10,00,11,011,111,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,100,011,111,011)
16116
16117
16118
16119 5433: 1(FREE)
16120 SCOPE702:
16121 P1, CSPD(04)_EMIT, EMIT/177777, |RESET MASK FOR SUBSEQUENT TESTS
16122 NEXT, BUTD(SCOPE), |NO ERROR: "TEST710A" (+1, WORDS)
16123 J/TEST710A | ERROR: "MASK702A" (-1, WORDS)
16124 (5433) DCS(0,00,0,1,0,0,1) BM(1111,10,11,11,11,111,111,0,0,0,0,0,0,0,1011,1,0,0000,0,0,11,000,100,111,101)
16124
16125
16126
16127
16128
16129 1.PAGE=====
16130
16131
16132 .TnC * TEST710-722: BUS FUNCTION DECODE, BUS ERROR CONDITIONS
16133
16134 |=====
16135 |*
16136 |* TEST8: 710A - 722C UWORDS: 133 + 274
16137 |*
16138 |* FUNCTIONS:
16139 |*
16140 |* WE GET SNEAKY HERE AND TEST THE FULL BUS DECODE/STATUS MUX LOGIC, AND THE BUS ERROR
16141 |* CONDITIONS (ODD ADDRESS, INTERNAL ADDRESS, ETC) ALMOST WITHOUT EVER GOING OUT ON THE
16142 |* BUS (WE DO ONCE, FOR SSYN TIMEOUT).
16143 |*
16144 |=====
16145
16146
16147
16148 | - - - - -
16149
16150 1TESTING "DATO", "ODD-ADDRESS" JAMUUP, 18,16, BIT IO-PAGE DECODE
16151
16152 | - - - - -
16153
16154 1TEST-710-A FIRST ATTEMPTS TO DO A BUS "DATO" FUNCTION, TRYING TO FORCE AN "ODD ADDRESS"
16155 1 ABORT/JAMUUP
16156 5475:
16157 TFS710A:
16158 PO, LOAD-ENVA(4777), |JAMUUP ADDRESS
16159 LOAD-ERROR(702A), |ERROR DIRECTORY KEY
16160 DCS-CTR(C6,), |COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
16161 NEXT, J/LOADRET710A
16162 (5475) DCS(1,00,1,0,0,0,1) BM(1001,00,10,00,11,111,111,0,0,0,0,0,0,0,0,0000,0,0,0000,0,0,11,000,100,111,010)
```

```

16163 5472:
16164 LOADRET710A:
16165 P3, CSPD[00]_EMIT, RETURN/TEST710B, RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16166 NEXT, GOTO-PAGE(4); XFER
16167 J/LOADADR710A
(5472) DCS[0.00,0.0,0,0] BM[0110..10,11..01,00..100..100...0,0,0,0...0,1111...1..0000,0...11,100...011,000,1111

16168 4307: I(FREE)
16169 LOADADR710A:
16170 P3, CSPD[16]_EMIT, EMIT/160001, "DIAGNOSTIC" UNIBUS I/O PAGE ADDRESS; ODD BYTE
16171 NEXT, J/SETJAM710A
16172 (4307) DCS[0.00,0.0,0,0] BM[1110..10,00..00,00..000..001...0,0,0,0...0,0001...1..0000,0...11,000...011,001,010]

16173 4312: I(FREE)
16174 SETJAM710A:
16175 P2-T, SR_CSPD(D16), SET BUS<00>=(1) FOR JAMUPP EXPECTED
16176 NEXT, J/BUSFCN710A
16177 (4312) DCS[0.00,0.0,0,0] BM[1010..10,00..00,00..000..000...0,0,1,0...0,0001...0..0000,0...11,000...011,001,011]

16178 4313: I(FREE)
16179 BUSFCN710A:
16180 P1, BA_SR, B8PHI(C177777), SET BA<17:00>=(780001), ALTERED TO(160001)
16181 P2-T, D_ZERO, SINCE WE'VE LEFT 18, BIT MODE (IN LOADING BA, THAT IS)
16182 P3, DATO, DO A BUS "DATO", SHOULD GET ODD ADDRESS ABORTED
16183 NEXT, J/NEXT710A IGO DELAY
16184 (4313) DCS[0.00,0.0,0,0] BM[0011..01,11..00,00..101..000...0,1,0,1...1,0010...0..0000,0...11,000...100,100,101]

16185 1** AT THIS POINT JAMUPP SHOULD OCCUR **
16186 1** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
16187 1** RETURN TO ADDRESS LEFT IN CSP(00) **
16188 1*** END UP HERE IF NO JAMUPP ***
16189 4445:
16190 NEXT710A:
16191 SETUP, RETURN/TEST710A, FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16192 NEXT, GOTO-PAGE(7); MUST DELAY A FEW MICROWORDS FOR BUS
16193 J/BUTD-18-ZERO ERROR TO TAKE EFFECT (IGNORE "BUS" OUTCOME HERE)
16194 (4445) DCS[0.00,0.0,0,0] BM[0101..00,10..01,11..101..111...0,0,0,0...0,0000...0..0000,0...11,100...011,100,001]

16195 1*** END UP HERE IF JAMUPP ***
16196
16197
16198
16199
16200
16201
16202 I - - - - -
16203
16204 TEST-710-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG:
16205
16206 BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16207 FCN: ODD 0 SSYN YEL RED WCS PON MEM SSYN CACH ILL WGT RED ODD WCS UBRK
16208 TIME ZON ZON PAR DIS PAR TIME ERR ADR ADR ZON ADR PAR TRAP
16209 (101004) 1 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0

```

```

16210 6644:
16211 TEST710C:
16212 PO, LOAD-ENUA(2TARGET402), SETUP FOR D=ZERO COMPARE
16213 LOAD-ERROR(710B), ERROR DIRECTORY KEY
16214 DCS-CTR(C10,); ICOMPARE AT TARGET
16215 NEXT, J/GOTEST710B
16216 (6644) DCS[1.00,1.0,0,0] BM[0101..00,11..11,00..000..010...0,0,0,0...0,0000...0..0000,0...11,000...100,110,100]

16217 6464: I(FREE)
16218 GOTEST710B:
16219 SETUP, RETURN/TEST710C, IGO TO SUBR WHICH:
16220 NEXT, CALL(ODDJAMTOD) (JAMREG)-XOR=(101004) -> D, BUT(D=ZERO) (ODD-ADDRESS)
16221 (6464) DCS[0.00,0.0,0,0] BM[0110..00,11..01,00..101..111...0,0,0,0...0,0000...0..0000,0...11,100...010,101,100]

16222
16223
16224
16225 I - - - - -
16226
16227 TEST-710-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG;
16228 NOTE PBA<17:16> READ "TRUE" (OUT OF BA) WHEN IN 18, BIT CONSOLE MODE
16229
16230 BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16231 FCN: DATI BG 0 NPR DATOB DATO PBA PBA HIB LOB TAG CON FLT PON CACH YEL
16232 SERV TIME 17 16 ERR ERR ERR SERV SERV FAIL ERR ZON
16233 (002340) 0 0 0 0 0 1 0 0 1 1 1 0 0 0 0 0
16234
16235 6645:
16236 TEST710C:
16237 PO, LOAD-ENUA(2TARGET402), SETUP FOR D=ZERO COMPARE
16238 LOAD-ERROR(710C), ERROR DIRECTORY KEY
16239 DCS-CTR(C10,); ICOMPARE AT TARGET
16240 NEXT, J/EXPECT710C
16241 (6645) DCS[1.00,1.0,0,0] BM[0101..00,11..11,00..000..010...0,0,0,0...0,0000...0..0000,0...11,000...100,110,101]

16242 6465: I(FREE)
16243 EXPECT710C:
16244 P3, CSPD[02]_EMIT, EMIT/002340, IDATO(1)H SET, PBA<17:16>="00" IN 18, BIT CONSOLE MODE
16245 NEXT, J/GOTEST710C I IN SERVICE REG
16246 (6465) DCS[0.00,0.0,0,0] BM[0000..10,01..00,11..100..000...0,0,0,0...0,0101...1..0000,0...11,000...100,110,110]

16247 6466: I(FREE)
16248 GOTEST710C:
16249 SETUP, RETURN/TEST710D, IGO TO SUBR WHICH:
16250 NEXT, CALL(SERVICETOD) (SERVICE)-XOR=CSP(02) -> D, BUT(D=IS-ZERO)
16251 (6466) DCS[0.00,0.0,0,0] BM[0110..00,10..11,11..100..111...0,0,0,0...0,0000...0..0000,0...11,100...010,101,000]

16252
16253
16254
16255
16256 TEST-710-D CHECKS THAT PBA<17:16> ARE INDICATED IN SERVICE REG;

```

```
16257 | NOTE PRA<17:16> READ AS BA<15:13>="111" WHEN IN 16. BIT MODE
16258 |
16259 | BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16260 | FCN: DATI BG 0 NPR DATOB DATO PBA PBA NIB LOS TAG CON FLT POW CACH YEL
16261 | SERV TIME 17 16 ERR ERR ERR SERV SERV FAIL ERR ZON
16262 | (003740) 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
16263 |
16264 | 6574:
16265 | TEST710D:
16266 | PO, LOAD-ENUA(ZTARGET402), |SETUP FOR D=ZERO COMPARE
16267 | LOAD-ERROR(TEST710D), |ERROR DIRECTORY KEY
16268 | DCS-CTR(C11,), |COMPARE AT TARGET
16269 | NEXT, J/EXPECT710D
(6574) DCS[1.00,1.0,0.0] BM[0100..00,11..11,00..0000..010...0.0.0..0...0.0000...0..0000,0...11,000...100,119,111]
16270 |
16271 | 6467: 1(FREE)
16272 | EXPECT710D:
16273 | PO, BUMP-VERIFY, |COUNT
16274 | P3, CSPD[02]_EMIT, EMIT/003740, |DATO(1)H SET, PBA<17:16>="11" IN 16. BIT MODE
16275 | NEXT, J/CLEAR710D | IN SERVICE REG
(6467) DCS[0.00,0.0,0.0.1] BM[0000..10,01..11,11..100..000...0.0.0..0...0.101...1..0000,0...11,000...100,111,000]
16276 |
16277 | 6470: 1(FREE)
16278 | CLEAR710D:
16279 | P3, CLR-JAM-ERRORS-[I], |CLEAR OUT JAM-REG, FOR USE IN NEXT TEST
16280 | CLEAR-CONSOLE-LED, |BACK TO 16. BIT MODE
16281 | NEXT, J/GOTEST710D
(6470) DCS[0.00,0.0,0.0.0] BM[0100..00,00..10,00..010..001...0.0.0..0...0.1011...0..0000,0...11,000...100,111,001]
16282 |
16283 | 6471: 1(FREE)
16284 | GOTEST710D:
16285 | SETUP, RETURN/TEST710E, |GO TO SUBR WHICH:
16286 | NEXT, CALL(SERVICETOD) | (SERVICE)=XOR-CSP(02) -> D, BUT(D-IS-ZERO)
(6471) DCS[0.00,0.0,0.0.0] BM[0110..00,10..11,11..101..111...0.0.0..0...0.0000...0..0000,0...11,100...010,101,000]
16287 |
16288 |
16289 |
16290 |
16291 |
16292 | 1TEST-710-F CHECKS THAT THE CLEAR-JAM-ERRORS CLEARED THE JAM REG TO (001000)
16293 | 6575:
16294 | TEST710E:
16295 | PO, LOAD-ENUA(ZTARGET402), |SETUP FOR D=ZERO COMPARE
16296 | LOAD-ERROR(TEST710E), |ERROR DIRECTORY KEY
16297 | DCS-CTR(C10,), |COMPARE AT TARGET
16298 | NEXT, J/GOTEST710E
(6575) DCS[1.00,1.0,0.0.0] BM[0101..00,11..11,00..0000..010...0.0.0..0...0.0000...0..0000,0...11,000...100,111,010]
16299 |
16300 | 6472: 1(FREE)
16301 | GOTEST710F:
16302 | PO, BUMP-VERIFY, |COUNT
16303 | SETUP, RETURN/SCOPE710, |GO TO SUBR WHICH:
16304 | | CSP(02) <- (001000)
```

```
16305 | NEXT, CALL(CLPJAMTON) | (JAMREG)=XOR-CSP(02) -> D, BUT(D-IS-ZERO)
(6472) DCS[0.00,0.0,0.0.1] BM[0101..00,10..01,00..101..111...0.0.0..0...0.0000...0..0000,0...11,100...010,101,101]
16306 |
16307 |
16308 |
16309 | 5445: 1(FREE)
16310 | SCOPE710:
16311 | P2-U, IP_EMIT, EMIT/123456, |A "CMP-BYTE" INSTR, FOR NEXT TEST SERIES
16312 | NEXT, BUTD(SCOPE), |NO ERROR: "TEST711A" (+1, WORDS)
16313 | J/TEST711A | ERROR: "LOADRET710A" (-17, WORDS)
(5445) DCS[0.00,0.1,0.0.0] BM[1010..00,01..11,00..101..110...0.0.0..0...0.1010...0..0000,0...11,000...100,111,011]
16314 |
16315 |
16316 |
16317 |
16318 |
16319 |
16320 |
16321 |
16322 | 1TESTING "DATOB=BYTE", "SSYN TIMEOUT" JAMUPP
16323 |
16324 |
16325 |
16326 | 1TEST-711-A DOES A BUS "DATOB=BYTE" FUNCTION, TRYING TO FORCE AN "SSYN TIMEOUT" ABORT/JAMUPP
16327 | 5473:
16328 | TEST711A:
16329 | PO, LOAD-ENUA(4777), |JAMUPP ADDRESS
16330 | LOAD-ERROR(TEST711A), |ERROR DIRECTORY KEY
16331 | DCS-CTR(C4,), |COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
16332 | NEXT, J/LOADRET711A
(5473) DCS[1.00,1.0,0.0.0] BM[1011..00,10..01,11..111...0.0.0..0...0.0000...0..0000,0...11,000...100,111,000]
16333 |
16334 | 5470:
16335 | LOADRET711A:
16336 | PO, BUMP-VERIFY, |COUNT
16337 | P3, CSPD[00]_EMIT, RETURN/TEST711B, |RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16338 | NEXT, GOTO-PAGE(4), |XPER
16339 | NEXT, J/RUSFCN711A
(5470) DCS[0.00,0.0,0.0.1] BM[0110..10,10..10,11..100...100...0.0.0..0...0.111...1..0000,0...11,100...011,001,001]
16340 |
16341 | 4311: 1(FREE)
16342 | BUSFCN711A:
16343 | INOTE: RA=(160001) FROM PREVIOUS TEST
16344 | P2-T, SR_CSPB(B16), |SET BIT<00>=(1) FOR JAMUPP EXPECTED
16345 | P3, DATOB, |DO A BUS "DATOB=BYTE", SHOULD GET SSYN TIMEOUT ABORTED
16346 | | SINCE WERE USING THE "DIAGNOSTIC" UNITBUS TIMEOUT ADDR
16347 | NEXT, J/NEXT711A |GO DELAY
(4311) DCS[0.00,0.0,0.0.0] BM[1010..11,01..00,00..000...000...0.0.1..0..0...0.1010...0..0000,0...11,000...100,100,110]
16348 |
16349 | 1** AT THIS POINT JAMUPP SHOULD OCCUP **
16350 | 1** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
16351 | 1** RETURN TO ADDRESS LEFT IN CSP(00) **
```

```
16352 1*** END UP HERE IF NO JAMUPP ***
16353 4446:
16354 NEXT711A:
16355     SETUP, RETURN/TEST711A,          I FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16356     NEXT,    GOTO=PAGE(7),           I MUST DELAY A FEW MICROWORDS FOR BUS
16357     J/BUTD=IS-ZERO,                  I ERROR TO TAKE EFFECT {IGNORE "BUT" OUTCOME HERE}
16358 (4446) DCS[0.00.0.0.0.0] BM[0101..00.10..01.11..011..111..0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.100.001]

16359
16360 1*** END UP HERE IF JAMUPP ***
16361
16362 I - - - - -
16363
16364 I TEST-711-B CHECKS THAT THE RIGHT JAM (SSYN TIMEOUT) IS INDICATED IN THE JAMREG:
16365 I
16366 I     BIT:   15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
16367 I     FCN:   ODD   0   SSYN YEL RED WCS POW MEM SSYN CACH ILL NGT RED ODD WCS UBRK
16368 I     ADR     TIME ZON ZON PAR DIS PAR TIME ERR ADR APT ZON ADR PAR TRAP
16369 I     (021200) 0   0   1   0   0   0   1   0   1   0   0   0   0   0   0   0
16370
16371 6534:
16372 TEST711B:
16373     PO,      LOAD=ENUA(ZTARGET402),      I SETUP FOR D=ZERO COMPARE
16374     LOAD=ERROR(TEST711B),                I ERROR DIRECTORY KEY
16375     DCS=CTR(C10.),                       I COMPARE AT TARGET
16376     NEXT,    J/EXPECT711B
16377 (6534) DCS[1.00.1.0.0.0] BM[0101..00.11..11.00..000..010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...100.111.011]

16378
16379 6473: I(FREE)
16380 EXPECT711A:
16381     PO,      BUMP-VERIFY,                I COUNT
16382     P3,      CSPD[02]_EMIT, EMIT/021200, I "SSYN TIMEOUT(1)H" SET
16383     NEXT,    J/GOTEST711B                I IN JAMREG
16384 (6473) DCS[0.00.0.0.0.0] BM[0010..10.00..10.10..000..000...0.0.0.0.0.0.0.1101...1.0000.0...11.000...100.111.100]

16385
16386 6474: I(FREE)
16387 GOTEST711B:
16388     SETUP,   RETURN/TEST711C,            I GO TO SUBR WHICH:
16389     NEXT,    CALL(JAMTOD)                 I (JAMREG)-XOR-CSP(02) -> D, BUT(D=IS-ZERO)
16390 (6474) DCS[0.00.0.0.0.0] BM[0110..00.10..10.11..000..111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.101.110]

16391
16392 I - - - - -
16393
16394 I TEST-711-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17>16 ARE INDICATED IN SERVICE REG:
16395 I BACK IN 16. BIT MODE, SINCE I-O PAGE, PBA<17>16 READ AS "11"
16396 I
16397 I     BIT:   15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
16398 I     FCN:   DATA BG   0   NPR DATOB DATO PBA PBA HIB LOB TAG CON FLT POW CACH YEL
16399 I
```

```
16400 I     SERV    TIME    17   16   ERR   ERR   ERR   SERV   SERV   FAIL   ERR   ZON
16401 I     (005740) 0   0   0   0   1   0   1   1   1   1   1   0   0   0   0   0
16402
16403 6530:
16404 TEST711C:
16405     PO,      LOAD=ENUA(ZTARGET402),      I SETUP FOR D=ZERO COMPARE
16406     LOAD=ERROR(TEST711C),                I ERROR DIRECTORY KEY
16407     DCS=CTR(C11.),                       I COMPARE AT TARGET
16408     NEXT,    J/EXPECT711C
16409 (6530) DCS[1.00.1.0.0.0] BM[0100..00.11..11.00..000..010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...100.111.101]

16410
16411 6475: I(FREE)
16412 EXPECT711C:
16413     P3,      CSPD[02]_EMIT, EMIT/005740, I DATOB(1)H SET, PBA<17>16="11"
16414     NEXT,    J/GOTEST711C                I IN SERVICE REG
16415 (6475) DCS[0.00.0.0.0.0] BM[0000..10.10..11.11..100..000...0.0.0.0.0.0.0.1101...1.0000.0...11.000...100.111.110]

16416
16417 6476: I(FREE)
16418 GOTEST711C:
16419     SETUP,   RETURN/TEST711D,            I GO TO SUBR WHICH:
16420     NEXT,    CALL(CJESRVICETOD)          I CLR-JAM-ERRORS-[1], FOR NEXT TEST
16421     (SERVICE)-XOR-CSP(02) -> D, BUT(D=IS-ZERO)
16422 (6476) DCS[0.00.0.0.0.0] BM[0110..00.10..10.10..110..111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.100.111]

16423
16424 I - - - - -
16425
16426 I TEST-711-D CHECKS THAT THE CLEAR-JAM-ERRORS CLEARED THE JAM REG TO (001000)
16427 6526:
16428 TEST711D:
16429     PO,      LOAD=ENUA(ZTARGET402),      I SETUP FOR D=ZERO COMPARE
16430     LOAD=ERROR(TEST711D),                I ERROR DIRECTORY KEY
16431     DCS=CTR(C10.),                       I COMPARE AT TARGET
16432     BUMP-VERIFY,                          I COUNT
16433     NEXT,    J/GOTEST711D
16434 (6526) DCS[1.00.1.0.0.0] BM[0101..00.11..11.00..000..010...0.0.0.0.0.0.0.0000...0.0000.0...11.000...100.111.111]

16435
16436 6477: I(FREE)
16437 GOTEST711D:
16438     SETUP,   RETURN/SCOPE711,            I GO TO SUBR WHICH:
16439     NEXT,    CALL(CLRJAMTOD)              I CSP(02) <= (001000)
16440     (JAMREG)-XOR-CSP(02) -> D, BUT(D=IS-ZERO)
16441 (6477) DCS[0.00.0.0.0.0] BM[0101..00.10..01.00..110..111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.101.101]

16442
16443 5446: I(FREE)
16444 SCOPE711:
16445     NEXT,    BUTD[SCOPE],                 I NO ERROR: "TEST712A" (+1. WORDS)
16446     J/TEST712A                           I ERROR: "LOADRET711A" (-1. WORDS)
16447 (5446) DCS[0.00.0.1.0.0] BM[0000..00.00..00.00..000..000...0.0.0.0.0.0.0.0000...0.0000.0...11.000...100.111.001]
```

```

16446
16447
16448
16449
16450
16451
16452 | .....
16453
16454 ITESTING "DATI", "INTERNAL ADDRESS" JAMUPP
16455 | .....
16456
16457 1TST-712-A DOES A BUS "DATI" FUNCTION TO AN INTERNAL ADDRESS, TRYING TO FORCE AN "INTERNAL ADDRESS" ABORT/JAMUPP
16458 5471:
16459 TST712A:
16460      PO,          LOAD=ENUA(4777),           !JAMUPP ADDRESS
16461                      LOAD=ERROR(TEST712A),    !ERROR DIRECTORY KEY
16462                      DCB=CTR(CS.),            !COMPARE JUST AFTER BUS CYCLE UNORD, AT JAM
16463                      NEXT,                     !
16464                      J/LOADRET712A
(5471) DCS[0.00.1.0.0.0] BM[0101..00.10.01.11..111...0.0.0.0.0...0.0000...0.0000.0...11.000...100.110.100]
16465
16466      5466:
16467      LOADRET712A:
16468          PO,          RUMP=VERIFY,             !COUNT
16469          P3,          CSBP[00]_ENIT, RETURN/TEST712B, !RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16470          NEXT,        GOTO=PAGE(4),             !XFER
16471                      J/GENADR712A              !
(5466) DCS[0.00.0.0.0.0] BM[0101..10.10.10.00..101..100...0.0.0.0.0...0.111...1..0000.0...11.100...011.001.100]
16472
16473      4314: I(FREE)
16474      GENADR712A:
16475          P2=T,        D_NOT=ASPHI(C000001),     !MAKE D=(177776), INTERNAL ADDRESS OF PSW
16476          P3,          ASPL0[111]_D,             ! AND SAVE IT
16477          N*XT,        J/BUSPCN712A              !
(4314) DCS[0.00.0.0.0.0] BM[0000..00.00..11.01..000..000...0.1.0.0.0...0.0000...0.0001.0...11.000...011.001.110]
16478
16479      4316: I(PREF)
16480      BUSPCN712A:
16481          P1,          RA=ASPL0(R11),             !BA=(177776)
16482          P2=T,        SR=SPHI(C000001),         !SET BIT#000001 FOR JAMUPP EXPECTED
16483          P3,          DATI,                       !DO A BUS "DATI", SHOULD GET INTERNAL ADDR ABORTED
16484          NEXT,        J/NEXT712A                 !GO DELAY
(4316) DCS[0.00.0.0.0.0] BM[0101..01.11..10.01..000..000...0.0.1..1..0...1.0110...0.0000.0...11.000...100.100.111]
16485
16486      !! AT THIS POINT JAMUPP SHOULD OCCUR **
16487      !! CLASSIC FLOW (4777) => (4757) => (7XXX) => (4XXX), AND THEN WE'RE BACK HERE **
16488      !! RETURN TO ADDRESS LEFT IN CSP(00) **
16489
16490      !!! END UP HERE IF NO JAMUPP ***
16491      4447:
16492      NEXT712A:
16493          SETUP,       RETURN/TEST712A,           !FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16494          NEXT,        GOTO=PAGE(77),             ! MUST DELAY A FEW MICROWORDS FOR BUS

```

```

16495 J/BUTD-IS-ZERO I ERROR TO TAKE EFFECT (IGNORE "BUT" OUTCOME HERE)
(4447) DCS10.00.0.0.0.0] BM10101.00.10.01.11.001.111...0.0.0.0.0...0.0000...0.0000.0...11.100...011.100.001]

16496
16497
16498 I== END UP HEPE IF JAMUPP ==
16499
16500
16501 I - - - - -
16502
16503 ITEST712-B CHECKS THAT THE RIGHT JAM (INTERNAL ADDRESS) IS INDICATED IN THE JAMREG;
16504 I FOR THIS JAM, THE JAMREG SHOULD REMAIN CLEAR
16505 I
16506 I BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16507 I FCM: ODD 0 88YN YEL RED WCS POW MEN 88YN CACH ILL MGT RED ODD WCS UBRK
16508 I ADR TIME ZON ZON PAR DIS PAR TIME ERR ADR AYT ZON ADR PAR TRAP
16509 I (0010000) 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
16510
16511 5505:
16512 TEST712B:
16513 PN, LOAD=ENUA(ZTARGET402), ISETUP FOR D=ZERO COMPARE
16514 LOAD=ERROR(TEST712B), IERROR DIRECTORY KEY
16515 DCB=CTR(C10.), ICOMPARE AT TARGET
16516 NEXT, J/GOTEST712B I
(5505) DCS11.00.1.0.0.0] BM10101.00.11.11.00.000.010...0.0.0.0.0...0.0000...0.0000.0...11.000...100.101.000]

16517
16518 5450: I(FREE)
16519 GOTEST712B:
16520 SETUP, RETURN/TEST712C, IGO TO SUBR WHICH:
16521 I (001000) -> CSP(02)
16522 NEXT, CALL[CLRJAMTOD] I (JAMREG)-XOR-CSP(02) -> D, BUT(D-IS-ZERO)
(5450) DCS10.00.0.0.0.0] BM10110.00.10.10.10.100.111...0.0.0.0.0...0.0000...0.0000.0...11.100...010.101.101]

16523
16524
16525
16526 I - - - - -
16527
16528 ITEST712-C CHECKS THAT THE "OTHER-JAM-H" SIGNAL IS LOW, INDICATING AN INTERNAL ADDRESS JAM
16529 6524:
16530 TEST712C:
16531 PN, LOAD=ENUA(ZTARGET401), IBIT<01> CLEAR
16532 LOAD=ERROR(TEST712C), IERROR DIRECTORY KEY
16533 DCB=CTR(C3.), ICOMPARE AT TARGET
16534 NEXT, J/GOTEST712C I
(6524) DCS11.00.1.0.0.0] BM1100.00.11.11.00.000.001...0.0.0.0.0...0.0000...0.0000.0...11.000...101.000.000]

16535
16536 6500: I(FREE)
16537
16538 CORUT712C:
16539 SETUP, RETURN/TEST712D, IRETURN TO START OF NEXT SUBTEST
16540 NEXT, GOTO-PAGE(7), I BUT TABLE
16541 J/BUTOTHERJAM I"OTHER JAM H" TN BIT<01>
(6500) DCS10.00.0.0.0.0] BM10110.00.10.10.10.101.111...0.0.0.0.0...0.0000...0.0000.0...11.100...011.101.010]

```

```

16541
16542
16543
16544
16545
16546 |TEST-712-D CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG;
16547 | BACK IN 16. BYT MODE, SINCE I-O PAGE, PBA<17:16> READ AS "11"
16548 |
16549 |
16550 | BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16551 | FCN: DATI 5G 0 NPR DATOB DATO PBA PBA HIB LOB TAG CON FLT POW CACH YEL
16552 | SERV TIME 17 16 ERR ERR ERR SERV SERV FAIL ERR ZON
16553 | (101740) 1 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
16554
16555 6523:
16556 TEST712D:
16557 PO, LOAD=ENUA(ZTARGET402), |SETUP FOR BZERO COMPARE
16558 LOAD=ERROR(TEST712D), |ERROR DIRECTORY KEY
16559 DCS=CTR(C1.), |COMPARE AT TARGET
16560 J/EXPEC712D
16561 (6523) DCS(1.00,1.0,0,0) BM(10100..00,11..11,00..0000..010...0.0,0,0,0...0.0000...0..0000,0...11,000...101,000,001)
16562
16563 6501: 1(FREE)
16564 EXPEC712D:
16565 P3, CSPD[02]_EMIT, EMIT/101740, |DATI(1)H SET, PBA<17:16>="11"
16566 NEXT, J/GOTEST712D | IN SERVICE REG
16567 (6501) DCS(1.00,0.0,0,0) BM(1000..10,00..11,11..100..0000...0.0,0,0,0...0.1101...1..0000,0...11,000...101,000,010)
16568
16569 6502: 1(FREE)
16570 GOTEST712D:
16571 SETUP, RETURN/SCOPE712, |GO TO SUBR WHICH:
16572 | CLR-JAM=ERRORS-(1), FOR NEXT TEST
16573 NEXT, CALL[CJESERVICETOD] | (SERVICE)-ROR-CSP(02) -> D, BUT(D-IS-ZERO)
16574 (6502) DCS(1.00,0.0,0,0) BM(1010..00,10..01,00..111..111...0.0,0,0,0...0.0000...0..0000,0...11,100...010,100,111)
16575
16576 5447: 1(FREE)
16577 SCOPE712:
16578 NEXT, BUTD[SCOPE], |NO ERROR: "TEST713A" (+1, WORDS)
16579 J/TEST713A | ERROR: "LOADRET712A" (-12, WORDS)
16580 (5447) DCS(1.00,0.1,0,0) BM(0000..00,00..00,00..0000..0000...0.0,0,0,0...0.0000...0..0000,0...11,000...100,110,111)
16581
16582
16583
16584
16585
16586
16587
16588
16589

```

```

16590
16591 |TEST-713-A DOES A BUS "DATI-NOINT" FUNCTION TO AN INTERNAL ADDRESS, TRYING TO FORCE
16592 | AN "ILLEGAL INTERNAL ADDRESS" ABORT/JAMUPP
16593 5467:
16594 TEST713A:
16595 PO, LOAD=ENUA(4777), |JAMUPP ADDRESS
16596 LOAD=ERROR(TEST713A), |ERROR DIRECTORY KEY
16597 DCS=CTR(C4.), |COMPARE JUST AFTER BUS CYCLE UWORD, AT JAM
16598 RUMP=VERIFY, |COUNT
16599 NEXT, J/LOADRET713A
16600 (5467) DCS(1.00,1.0,0,1) BM(1011..00,10..01,11..111..111...0.0,0,0,0...0.0000...0..0000,0...11,000...100,110,100)
16601
16602 5464:
16603 LOADRET713A:
16604 P3, CSPD[00]_EMIT, RETURN/TEST713B, |RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16605 NEXT, GOTO-PAGE(7), |XFER
16606 J/BUSFCN713A
16607 (5464) DCS(1.00,0.0,0,0) BM(1010..10,11..00,01..110..111...0.0,0,0,0...0.1111...1..0000,0...11,100...001,110,111)
16608
16609 7167: 1(FREE)
16610 BUSFCN713A:
16611 P1, BA_ASPL0(R11), |BA=(177776), GENERATED IN PREVIOUS SET OF TESTS
16612 P2-T, SR_BSPHI(C000001), |SET BIT<40>=(1) FOR JAMUPP EXPECTED
16613 P3, DATI-NOINT, |DO A BUS "DATI-NOINT",
16614 | SHOULD GET "ILLEGAL INTERNAL ADDR" ABORTED
16615 NEXT, J/NEXT713A |GO DELAY
16616 (7167) DCS(1.00,0.0,0,0) BM(1010..01,11..10,01..000..000...0.0,1,1,1...0.0001...0..0000,0...11,000...011,110,101)
16617
16618 |** AT THIS POINT JAMUPP SHOULD OCCUR **
16619 |** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
16620 |** RETURN TO ADDRESS LEFT IN CSP(00) **
16621
16622 |*** END UP HERE IF NO JAMUPP ***
16623 7365:
16624 NEXT713A:
16625 SETUP, RETURN/TEST713A, |FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16626 NEXT, GOTO-PAGE(7), | MUST DELAY A FEW MICROWAYS FOR BUS
16627 J/BUTD-IS-ZERO | ERROR TO TAKE EFFECT (IGNORE "BUT" OUTCOME HERE)
16628 (7365) DCS(1.00,0.0,0,0) BM(1010..00,10..01,10..111..111...0.0,0,0,0...0.0000...0..0000,0...11,100...011,100,001)
16629
16630 |*** END UP HERE IF JAMUPP ***
16631
16632
16633
16634
16635
16636
16637
16638
16639

```

```

16632 |TEST-713-B CHECKS THAT THE RIGHT JAM (ILLEGAL INTERNAL ADDRESS) IS INDICATED IN THE JAMREG;
16633 |
16634 | BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16635 | FCN: ODD 0 SSYN YEL RPD WCS POW MEM SSYN CACH ILL MGT RED ODD WCS UBRK
16636 | ADR TIME ZON ZON PAR DIS PAR TIME ERR ADR ABT ZON ADP PAR TRAP
16637 | (001940) 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0
16638
16639

```

```

16639 5451:
16640 TEST713B:
16641 PO, LOAD=ENUA(ZTARGET402), ;SETUP FOR D=ZERO COMPARE
16642 LOAD=ERROR(TEST713B), ;ERROR DIRECTORY KEY
16643 DCS=CTR(C10,); ;COMPARE AT TARGET
16644 BUMP=VERIFY, ;COUNT
16645 NEXT, J/EXPECT713B
(5616) DCS[1.00.1.0.0.0.1] BM[0101.00.11.11.00.000.010...0.0.0.0.0...0.0000...0...0000.0...11.000...100.101.010]
16646
16647 5452: I(FREE)
16648 EXPECT713B:
16649 P3, CSPD[02]_EMIT, EMIT/001040, ;ILLEGAL INTERNAL ADDRESS(1)H" SET
16650 NEXT, J/GOTEST713B
(5457) DCS[0.00.0.0.0.0.0] BM[0000.10.00.10.00.100.000...0.0.0.0.0...0.1101...1.0000.0...11.000...100.101.011]
16651
16652 5453: I(FREE)
16653 GOTEST713B:
16654 SETUP, RETURN/TEST713C, ;GO TO SUBR WHICH:
16655 NEXT, CALL(JAMTOD) ; (JAMTOD)=XOR-CSP(02) -> D, BUT(D-IS-ZERO)
(5453) DCS[0.00.0.0.0.0.0] BM[0110.00.10.10.10.010.111...0.0.0.0.0...0.0000...0...0000.0...11.100...010.101.110]
16656
16657
16658
16659
16660
16661 ;TEST-713-C CHECKS THAT THE "OTHER-JAM-H" SIGNAL IS HIGH, INDICATING A JAM PRESENT OTHER
16662 ; THAN ONLY A VALID "INTERNAL ADDRESS" JAM
16663 5452:
16664 TEST713C:
16665 PO, LOAD=ENUA(ZTARGET403), ;BIT<01> SET
16666 LOAD=ERROR(TEST713C), ;ERROR DIRECTORY KEY
16667 DCS=CTR(C3,); ;COMPARE AT TARGET
16668 NEXT, J/GOTEST713C
(6572) DCS[1.00.1.0.0.0.0] BM[1100.00.11.11.00.000.011...0.0.0.0.0...0.0000...0...0000.0...11.000...101.000.011]
16669
16670 5453: I(FREE)
16671 GOTEST713C:
16672 SETUP, RETURN/TEST713D, ;RETURN TO START OF NEXT SUBTEST
16673 NEXT, GOTO=PAGE(7); ;BUT TABLE
16674 J/BUTOTHERJAM ;"OTHER JAM H" IN BIT<01>
(6503) DCS[0.00.0.0.0.0.0] BM[0110.00.10.10.10.001.111...0.0.0.0.0...0.0000...0...0000.0...11.100...011.101.010]
16675
16676
16677
16678
16679
16680 ;TEST-713-D CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG;
16681 ; BACK IN 16, BIT MODE, SINCE I-O PAGE, PBA<17:16> READ AS "11"
16682
16683
16684
16685
16686
16687
16688
16689
16690
16691
16692
16693
16694
16695
16696
16697
16698
16699
16700
16701
16702
16703
16704
16705
16706
16707
16708
16709
16710
16711
16712
16713
16714
16715
16716
16717
16718
16719
16720
16721
16722
16723
16724
16725
16726
16727
16728
16729
16730
16731
16732
16733
16734

```

```

16686
16687
16688
16689
16690
16691
16692
16693
16694
16695
16696
16697
16698
16699
16700
16701
16702
16703
16704
16705
16706
16707
16708
16709
16710
16711
16712
16713
16714
16715
16716
16717
16718
16719
16720
16721
16722
16723
16724
16725
16726
16727
16728
16729
16730
16731
16732
16733
16734

```

KD11-K MICRO V00A-1 00:00:03 12-MAR-77 PAGE 351 SEQ 0433

```

16735 (5465) DCS[1.00.1.0.0.0] BM[1000..00.10..01.11..111..111...0.0.0..0..0.0000...0..0000.0...11.000...100.110.010]
16736 5462:
16737 LOADRET720A:
16738 P3, CSPD[00].EMIT, RETURN/TEST720B, RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16739 NEXT, GOTO-PAGE(7), J/LOADIR720A JXFER
16740 (5462) DCS[0.00.0.0.0.0] BM[0110..10.10..10.10..000..111...0.0.0..0..0.1111...1..0000.0...11.100...001.111.010]
16741 7172: 1(FREE)
16742 LOADIR720A:
16743 P2=H, IR_EMIT, EMIT/000000, !HALT INSTRUCTION IS -BYTE
16744 NEXT, J/SETADR720A
16745 (7172) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..000...0.0.0..0..0.1.1010...0..0000.0...11.000...001.111.100]
16746 7174: 1(FREE)
16747 SETADR720A:
16748 P3, CSPD[16].EMIT, EMIT/060001, !BIT<15:13>="011", -IOPAGE; ODD-ADDRESS
16749 NEXT, J/SETJAM720A !DO A BUS "DATIB=-BYTE", SHOULD GET ODD ADDRESS ABORTED
16750 (7174) DCS[0.00.0.0.0.0] BM[0110..10.00..00.00..000..001...0.0.0..0..0.0.0001...1..0000.0...11.000...001.111.101]
16751 7175: 1(FREE)
16752 SETJAM720A:
16753 P2=T, SR_CSPD(D16), !SET BIT<00>=(1) FOR JAMUPP EXPECTED
16754 NEXT, J/BUSFCN720A
16755 (7175) DCS[0.00.0.0.0.0] BM[1010..10.00..00.00..000..000...0.0.1..0..0.0.0001...0..0000.0...11.000...001.111.110]
16756 7176: 1(FREE)
16757 BUSFCN720A:
16758 P1, BA_SR, !SET BA<17:10>=(060001)
16759 P3, DATIB, !DO A BUS "DATIB=-BYTE", SHOULD GET ODD ADDRESS ABORTED
16760 NEXT, J/NEXT720A !GO DELAY
16761 (7176) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..000...0.0.0..1..0.0.1.0011...0..0000.0...11.000...011.110.100]
16762 1** AT THIS POINT JAMUPP SHOULD OCCUR **
16763 1** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
16764 1** RETURN TO ADDRESS LEFT IN CSP(00) **
16765 1*** END UP HERE IF NO JAMUPP ***
16766 7364:
16767 NEXT720A:
16768 SETUP, RETURN/TEST720A, !FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16769 NEXT, GOTO-PAGE(7), ! MUST DELAY A FEW MICROWORDS FOR BUS
16770 J/BUSD-18-ZERO ! ERROR TO TAKE EFFECT (IGNORE "BUT" OUTCOME HERE)
16771 (7364) DCS[0.00.0.0.0.0] BM[0101..00.10..01.10..101..111...0.0.0..0..0.0.0000...0..0000.0...11.100...011.100.001]
16772 1*** END UP HERE IF JAMUPP ***
16773 1 - - - - -
16774 1 - - - - -
16775 1 - - - - -
16776 1 - - - - -
16777 1 - - - - -
16778 1 - - - - -
16779 1 - - - - -
16780 !TEST-720-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG;

```

KD11-K MICRO V00A-1 00:00:03 12-MAR-77 PAGE 352 SEQ 0434

```

16781 1
16782 1 BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16783 1 FCN: ODD 0 SSYN YEL RED WCS PDM MEM SSYN CACH ILL MGT RED ODD WCS UBRK
16784 1 ADR TIME ZON ZON PAR DIS PAR TIME ERR ADR ADR ADP PAR TRAP
16785 1 (101004) 1 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0
16786 4520:
16787 TEST720B:
16788 P0, LOAD-ENUA(ZTARGET402), !SETUP FOR D=ZERO COMPARE
16789 LOAD-ERROR(TEST720B), !ERROR DIRECTORY KEY
16790 DCS-CTR(C10.), !COMPARE AT TARGET
16791 BUMP-VERIFY, !COUNT
16792 NEXT, J/GOTEST720B
16793 (6520) DCS[1.00.1.0.0.1] BM[0101..00.11..11.00..000..010...0.0.0..0..0.0.0000...0..0000.0...11.000...101.000.110]
16794 6506: 1(FREE)
16795 GOTEST720B:
16796 SETUP, RETURN/TEST720C, !GO TO SUBR WHICH:
16797 NEXT, CALL[ODDJAMTOD] ! (JAMREG)-XOR-(101004) -> D, BUT(D=ZERO) [ODD-ADDRESS]
16798 (6506) DCS[0.00.0.0.0.0] BM[0100..00.11..11.10..000..111...0.0.0..0..0.0.0000...0..0000.0...11.100...010.101.100]
16799 1 - - - - -
16800 1 - - - - -
16801 1 - - - - -
16802 1 - - - - -
16803 1 - - - - -
16804 !TEST-720-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG;
16805 ! NOTE: PBA<17:16> SHOULD NOT BE FORCED TO "11", IN 16. BIT MODE, WHEN BA<15:13>="011"
16806 1
16807 1 BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16808 1 FCN: DATI 8G 0 NPR DATOR DATOR PBA PBA HIB LOB TAG CON FLT POW CACH YEL
16809 1 SERV TIME 17 16 ERR ERR ERR SERV SERV FAIL ERR ZON
16810 1 (100340) 1 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0
16811 4760:
16812 TEST720C:
16813 P0, LOAD-ENUA(ZTARGET402), !SETUP FOR D=ZERO COMPARE
16814 LOAD-ERROR(TEST720C), !ERROR DIRECTORY KEY
16815 DCS-CTR(C11.), !COMPARE AT TARGET
16816 NEXT, J/GOTEST720C
16817 (4760) DCS[1.00.1.0.0.0] BM[0100..00.11..11.00..000..010...0.0.0..0..0.0.0000...0..0000.0...11.000...011.001.111]
16818 4317: 1(FREE)
16819 GOTEST720C:
16820 SETUP, RETURN/TEST721A, !GO TO SUBR WHICH:
16821 NEXT, CALL[DATISERVICETOD] ! CLR-JAM-ERRORS, TO RESET FOR NEXT TEST
16822 ! (SERVICE)-XOR-(100340) -> D, BUT(D=ZERO) (DATI[1]H)
16823 (4317) DCS[0.00.0.0.0.0] BM[0100..00.10..11.11..001..111...0.0.0..0..0.0.0000...0..0000.0...11.100...010.100.101]
16824 16825
16826 16827
16828 16829

```

KD11-K MICRO V00A-1 00100103 12-MAR-77 PAGE 353 SEQ 0435

```

16829
16830
16831 |-----|
16832 |TESTING 'DATOB=BYTE', 'ODD=ADDRESS' JAMUPP, I/O=PAGE-PBA<17:16> DECODE
16833
16834 |-----|
16835
16836
16837 |TEST-721-A ATTEMPTS TO DO A BUS "DATOB=BYTE" FUNCTION, TRYING TO FORCE AN "ODD ADDRESS"
16838 | AROPT/JAMUPP, ALSO CHECKING THAT ALL SERVICE / JAM STATUS BITS SET CORRECTLY
16839 4571:
16840 TEST721A:
16841     PO,          LOAD=ENUA(4777),          |JAMUPP ADDRESS
16842     LOAD=ERROR(TEST721A),          |ERROR DIRECTORY KEY
16843     DCS=CTR(C6.),          |COMPARE JUST AFTER BUS CYCLE UNWORD, AT JAM
16844     NEXT,        J/LOADRET721A
16845 (4571) DCS(1.00,1.0,0.0) BM(1001.00,10.01,11.111.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.100.0.011,010,000)
16846
16847 4320: 1(FREE)
16848 LOADRET721A:
16849     P3,          CSPD[00]_EMIT, RETURN/TEST721R,          |RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16850     NEXT,        GOTO=PAGE(7),          |XFER
16851     NEXT,        J/SETADR721A
16852 (4320) DCS(0.00,0.0,0.0) BM(0100.10.10.11,11.011.111.0.0.0.0.0.0.0.1111.1.0000.0.0.11,100.0.001,111,011)
16853
16854 7173: 1(FREE)
16855 SETADR721A:
16856     P3,          CSPD[16]_EMIT, EMIT/120001,          |BIT<15:13>="101", -IOPAGE, ODD=ADDRESS
16857     NEXT,        J/SETJAM721A
16858 (7173) DCS(0.00,0.0,0.0) BM(1010.10.00.00,000.001.0.0.0.0.0.0.0.0001.1.0000.0.0.11,100.0.010,000,000)
16859
16860 7200: 1(FREE)
16861 SETJAM721A:
16862     P2=T,        SR_CSPD(D16),          |SET BIT<00>=(1) FOR JAMUPP EXPECTED
16863     NEXT,        J/BUSFCN721A
16864 (7200) DCS(0.00,0.0,0.0) BM(1010.10.00.00,000.000.0.0.0.0.0.0.0.0001.1.0000.0.0.11,100.0.010,000,001)
16865
16866 7201: 1(FREE)
16867 BUSFCN721A:
16868     P1,          BA_SR,          |NOTE: "IR"=(000000)=BYTE FROM PREV TEST
16869     P1,          DATOB,          |SET BA<17:00>=(120001)
16870     NEXT,        J/NEXT721A
16871     |DO A BUS "DATOB=BYTE", SHOULD GET ODD ADDRESS ABORTED
16872     |DO DELAY
16873 (7201) DCS(0.00,0.0,0.0) BM(0000.00.00.00,000.000.0.0.0.0.1.0.0.1.0101.0.0.0000.0.0.11,100.0.011,110,010)
16874
16875 |** AT THIS POINT JAMUPP SHOULD OCCUR **
16876 |** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (4XXX), AND THEN WE'RE BACK HERE **
16877 |** RETURN TO ADDRESS LEFT IN CSP(00) **
16878
16879 |*** FND UP HERE IF NO JAMUPP ***
16880 7362:
16881 NEXT721A:
16882     SETUP, RETURN/TEST721A,          |FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16883

```

KD11-K MICRO V00A-1 00100103 12-MAR-77 PAGE 354 SEQ 0436

```

16877     NEXT,        GOTO=PAGE(7),          | MUST DELAY A FEW MICROWORDS FOR BUS
16878     J/BUTD-IS=ZERO          | ERROR TO TAKE EFFECT (IGNORE "BUT" OUTCOME HERE)
16879 (7362) DCS(0.00,0.0,0.0) BM(0100.00,10.10.11,11.001.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11,100.0.011,100,001)
16880
16881 |*** END UP HERE IF JAMUPP ***
16882
16883 |-----|
16884
16885
16886 |TEST-721-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG:
16887 |
16888 |   BIT:   15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
16889 |   FCN:   ODD   0   SSYN YEL RED   WCS   POW   MEM   SSYN CACH ILL MGT RED   ODD   WCS   UBRK
16890 |   |   ADR   TIME ZON   PAR   DIS   PAR   TIME ERR ADR   ADR   ZON   ADR   PAR   TRAP
16891 |   (101004) 1   0   0   0   0   0   1   0   0   0   0   0   0   1   0   0
16892
16893 4573:
16894 TEST721B:
16895     PO,          LOAD=ENUA(ZTARGET402),          |SETUP FOR D=ZERO COMPARE
16896     LOAD=ERROR(TEST721B),          |ERROR DIRECTORY KEY
16897     DCS=CTR(C10.),          |COMPARE AT TARGET
16898     NEXT,        J/GOTEST721B
16899 (4573) DCS(1.00,1.0,0.0) BM(0101.00,11.11,00.000.0010.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11,100.0.011,010,001)
16900
16901 4321: 1(FREE)
16902 GOTEST721B:
16903     SETUP, RETURN/TEST721C,          |GO TO SUBR WHICH:
16904     NEXT,        CALL(ODDJAMTOD)          | (JAMREG)-XOR-(101004) -> D, BUT(D=ZERO) [ODD=ADDRESS]
16905 (4321) DCS(0.00,0.0,0.0) BM(0100.00,10.10.11,11.111.111.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11,100.0.010,101,100)
16906
16907 |-----|
16908
16909 |TEST-721-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REG:
16910 | NOTE: PBA<17:16> SHOULD NOT BE FORCED TO "11", IN 16, BIT MODE, WHEN BA<15:13>="101"
16911 |
16912 |   BIT:   15   14   13   12   11   10   09   08   07   06   05   04   03   02   01   00
16913 |   FCN:   DAT1   BG   0   NPR DATOB DATO   PBA   PBA   HIB   LOB   TAG   CON   FLT   POW   CACH YEL
16914 |   |   SERV   TIME   |   17   16   ERR   ERR   ERR   SERV SERV FAIL   ERR ZON
16915 |   (00234n) 0   0   0   0   0   1   0   0   1   1   1   0   0   0   0   0
16916
16917 4577:
16918 TEST721C:
16919     PO,          LOAD=ENUA(ZTARGET402),          |SETUP FOR D=ZERO COMPARE
16920     LOAD=ERROR(TEST721C),          |ERROR DIRECTORY KEY
16921     DCS=CTR(C11.),          |COMPARE AT TARGET
16922     NEXT,        J/GOTEST721C
16923 (4577) DCS(1.00,1.0,0.0) BM(0100.00,11.11,00.000.0010.0.0.0.0.0.0.0.0000.0.0.0.0.0.0.11,100.0.011,010,010)
16924
16925 4322: 1(FREE)
16926 GOTEST721C:

```

```

16926      SETUP, RETURN/SCOPE721,          !GO TO SUBR WHICH:
16927      NEXT, CALL(DATOSERVICETOD)         ! CLR=JAM-ERRORS, TO RESET FOR NEXT TEST
16928      (4322) DCS[0.00.0.0.0.0] BM[0101.00.10.01.01.100.111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.100.110]
16929
16930
16931
16932
16933      5454: 1(FREE)
16934      SCOPE721:
16935      NEXT, BUTD(SCOPE),                   !NO ERROR: "TEST722A" (+1, WORDS)
16936      J/TEST722A                          ! ERROR: "LOADRET720A" (-24, WORDS)
16937      (5454) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.0.0.0.0.0.0.0.0000...0.0000.0...11.100...100.110.011]
16938
16939
16940
16941
16942      | - - - - -
16943      |TESTING 'INVALIDATE', 'ODD-ADDRESS' JAMUPP, I/O-PAGE-PBA<17:16> DECODE
16944      | - - - - -
16945
16946      |TEST-722-A ATTEMPTS TO DO A BUS "INVALIDATE" FUNCTION, TRYING TO FORCE AN "ODD ADDRESS"
16947      | ABORT/JAMUPP, ALSO CHECKING THAT ALL SERVICE / JAM STATUS BITS SET CORRECTLY
16948      5463:
16949      TEST722A:
16950      PO, LOAD-ENUA(4777),                 !JAMUPP ADDRESS
16951      LOAD-ERROR(TEST722A),               !ERROR DIRECTORY KEY
16952      DCS-CTR(C6.),                      !COMPARE JUST AFTER BUS CYCLE UNWORD, AT JAM
16953      NEXT, J/LOADRET722A
16954      (5463) DCS[1.00.1.0.0.0] BM[1001.00.10.01.11.111...111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...100.101.110]
16955
16956      5456: 1(FREE)
16957      LOADRET722A:
16958      PO, CSPD[00]_EMIT, RETURN/TEST722B, !RETURN AFTER SUCCESSFUL JAM TO NEXT TEST
16959      NEXT, GOTO-PAGE(4),                 !XFER
16960      NEXT, J/SETADR722A
16961      (5456) DCS[0.00.0.0.0.0] BM[0100.10.10.01.10.100...100...0.0.0.0.0.0.0.111...1.0000.0...11.100...100.111.000]
16962
16963      4470:
16964      SETADR722A:
16965      PO, CSPD[16]_EMIT, EMIT/140001,     !BIT<15:13>="110", -IOPAGE; ODD-ADDRESS
16966      NEXT, J/SETJAM722A
16967      (4470) DCS[0.00.0.0.0.0] BM[1100.10.00.00.00.000.0.01...0.0.0.0.0.0.0.0001...1.0000.0...11.100...011.010.011]
16968
16969      4323: 1(FREE)
16970      SETJAM722A:
16971      PO, SR_CSPD(D16),                   !SET BIT<00>=(1) FOR JAMUPP EXPECTED
16972      NEXT, J/BUSFCN722A
16973      (4323) DCS[0.00.0.0.0.0] BM[1010.10.00.00.00.000.0.000...0.0.1.0.0.0.0.0001...0.0000.0...11.100...011.010.100]

```

```

16972
16973      4324: 1(FREE)
16974      BUSFCN722A:
16975      PO, BA_SR,                         !SET BA<17:00>=(140001)
16976      PO, INVALIDATE,                   !DO A BUS "INVALIDATE", SHOULD GET ODD ADDRESS ABORTED
16977      NEXT, J/NEXT722A                  !GO DELAY
16978      (4324) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.0.0.0.0.1.0.0.1.011...0.0000.0...11.100...101.010.100]
16979
16980      !** AT THIS POINT JAMUPP SHOULD OCCUR **
16981      !** CLASSIC FLOW (4777) -> (4757) -> (7XXX) -> (6XXX), AND THEN WE'RE BACK HERE **
16982      !** RETURN TO ADDRESS LEFT IN CSP(00) **
16983
16984      !*** END UP HERE IF NO JAMUPP ***
16985      4524:
16986      NEXT722A:
16987      SETUP, RETURN/TEST722A,             !FORCE A SCOPE LOOP ON THIS TEST, BUT FIRST
16988      NEXT, GOTO-PAGE(7),                 ! MUST DELAY A FEW MICROWORDS FOR BUS
16989      J/BUTD-15-ZERO                      ! ERROR TO TAKE EFFECT (IGNORE "BUT" OUTCOME HERE)
16990      (4524) DCS[0.00.0.0.0.0] BM[0101.00.10.01.10.011...111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.100.001]
16991
16992      !*** END UP HERE IF JAMUPP ***
16993
16994      | - - - - -
16995
16996      |TEST-722-B CHECKS THAT THE RIGHT JAM (ODD ADDRESS) IS INDICATED IN THE JAMREG:
16997      |
16998      | BIT: 15 14 13 12 11 10 09 08 07 06 05 04 03 02 01 00
16999      | FCN: ODD 0 0 SSYN YEL RED WCS POW MEM SSYN CACH ILL MGT RED ODD WCS UBRK
17000      | ADR TIME ZON ZON PAR DIS PAR TIME ERR ADR APT ZON ADR PAR TRAP
17001      | (101004) 1 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0
17002
17003      4464:
17004      TEST722B:
17005      PO, LOAD-ENUA(2TARGET402),          !SETUP FOR D=ZERO COMPARE
17006      LOAD-ERROR(TEST722B),               !ERROR DIRECTORY KEY
17007      DCS-CTR(C11.),                      !COMPARE AT TARGET
17008      BUMP-VERIFY,                         !COUNT
17009      J/GOTEST722B
17010      (4464) DCS[1.00.1.0.0.1] BM[0100.00.11.11.00.000.0.010...0.0.0.0.0.0.0.0000...0.0000.0...11.100...011.010.101]
17011
17012      4325: 1(FREE)
17013      GOTEST722B:
17014      SETUP, RETURN/TEST722C,             !GO TO SUBR WHICH:
17015      NEXT, CALL(ODDJAMTOD)              ! (JAMREG)-XOR-(101004) -> D, BUT(D=ZERO) [ODD-ADDRESS]
17016      (4325) DCS[0.00.0.0.0.0] BM[0100.00.10.01.10.101...111...0.0.0.0.0.0.0.0000...0.0000.0...11.100...010.101.100]
17017
17018
17019

```

```
17020 |TEST-722-C CHECKS THAT THE RIGHT BUS FUNCTION DECODE / PBA<17:16> ARE INDICATED IN SERVICE REGI
17021 | NOTE: PBA<17:16> SHOULD NOT BE FORCED TO "11", IN 16 BIT MODE, WHEN BA<15:13>="110"
17022 |
17023 |      BIT:      15  14  13  12  11  10  09  08  07  06  05  04  03  02  01  00
17024 |      FCN:      DATI  BG   0  NPR  DATOB  DATO  PBA  PBA  HIB  LOB  TAG  COM  FLT  POW  CACH  YEL
17025 |      |          SERV  TIME  |          17  16  ERR  ERR  ERR  SERV  SERV  FAIL  ERR  ZON
17026 |      (002340)  0   0   0   0   0   1   0   0   1   1   1   0   0   0   0   0
17027 |
17028 | 4465:
17029 | TEST722C:
17030 |      PO,          LOAD-ENUA(ZTARGET402),          |SETUP FOR D=ZERO COMPARE
17031 |                  LOAD-ERRORR(TEST722C),          |ERROR DIRECTORY KEY
17032 |                  DCS=CTR(C11.),                  |COMPARE AT TARGET
17033 |      NEXT,        J/GOTEST722C
17034 | (4465) DCS(1.00,1.0,0.0) BM(0100.00,11.11,1.00.000.010...0.0,0.0,0.0...0.0000...0.0000,0...11.000...011.010,110)
17035 |
17036 | 4326: 1(FREE)
17037 | GOTEST722C:
17038 |      SETUP,      RETURN/SCOPE722,          |GO TO ADDR WHICH:
17039 |                  NEXT, CALL(DATOSERVICETOD)    | CLR-JAM-ERRORS, TO RESET FOR NEXT TEST
17040 |                  | (SETVICE)=XDR-(002340) -> D, BUT(D=ZERO) (DATO[1]H)
17041 | (4326) DCS(0.00,0.0,0.0) BM(0100.00,01.10,10.111.111...0.0,0.0,0.0...0.0000...0.0000,0...11.100...010.100,110)
17042 |
17043 |
17044 | 4327: 1(FREE)
17045 | SCOPE722:
17046 |      NFXT,      BUTD[SCOPE],          |NO ERROR: "TEST730A" (+1. WORDS)
17047 |                  J/TEST730A          | ERROR: "SETADR720A" (-11. WORDS)
17048 | (4327) DCS(0.00,0.1,0.0) BM(0000.00,00.00,00.00.000.000...0.0,0.0,0.0...0.0000...0.0000,0...11.000...10.111,001)
17049 |
17050 |
17051 |
17052 |
17053 | 1.PAGE=====
17054 |
17055 |
17056 | .TOC * TEST730-731: BUS CYCLES TO/FROM MEMORY
17057 |
17058 | |=====
17059 | |*
17060 | |* TESTS: 730A - 731E          UWORDS: 063 + 057
17061 | |*
17062 | |* FUNCTIONS:
17063 | |*
17064 | |* THESE TESTS CHECK THAT ACTUAL BUS CYCLES CAN BE CORRECTLY EXECUTED.
17065 | |*
17066 | |=====
17067 |
17068 |
17069 |
```

```
17070 |
17071 |
17072 |
17073 |
17074 |
17075 | |THIS FIRST SERIES OF TESTS DOES A DATO/DATIP/DATD/DATIB SEQUENCE, CHECKING THAT
17076 | |FACH FUNCTION OPERATES AS EXPECTED.
17077 |
17078 | |-----
17079 |
17080 | |TEST-730-A DOES A DATO, AND THEN CHECKS THAT THE DBUF LATCH (DS) ALSO GETS LOADED WITH THE
17081 | | DATA, AND THAT IT IS ENABLED ON BUSDIN IN THE MICROWORD AFTER THE BUS CYCLE (IE, EMIT
17082 | | IS TEMPORARILY DISABLED)
17083 | 4471:
17084 | TEST730A:
17085 |      PO,          LOAD-ENUA(ZTARGET402),          |SETUP FOR D=ZERO COMPARE
17086 |                  LOAD-ERRORR(TEST730A),          |ERROR DIRECTORY KEY
17087 |                  DCS=CTR(C9.),                  |COMPARE AT TARGET
17088 |      NEXT,        J/LOADIR730A
17089 | (4471) DCS(1.00,1.0,0.0) BM(0110.00,11.11,1.00.000.010...0.0,0.0,0.0...0.0000...0.0000,0...11.000...111.101,100)
17090 |
17091 | 4754:
17092 | LOADIR730A:
17093 |      P2=U,      IR_EMIT, EMIT/125200,          |"CMP-BYTE" INSTR; INSTRS-E8(412) DECODE
17094 |      NEXT,      J/LOADDATA730A
17095 | (4754) DCS(0.00,0.0,0.0) BM(1010.00,10.10,10.10.000.000...0.0,0.0,0.0...1.1010...0.0000,0...11.000...011.011,000)
17096 |
17097 | 4330: 1(FREE)
17098 | LOADDATA730A:
17099 |      P3,      CSPD[16]_EMIT, EMIT/125252,          |PATTERN (125252) IN BASCON AREA
17100 |      NEXT,      J/BUSFCN730A
17101 | (4330) DCS(0.00,0.0,0.0) BM(1010.00,10.10,10.10.101.010...0.0,0.0,0.0...0.0001...1.0000,0...11.000...011.011,001)
17102 |
17103 | 4331: 1(FREE)
17104 | BUSFCN730A:
17105 |      PO,          BUMP-VERIFY,          |COUNT
17106 |      P1,          BA_ASPHI(C000000),          |USE MEMORY ADDR(000000)
17107 |      P2=T,        D_CSPB(B16), D(C)=0,          |USE DATA (125252)
17108 |      P3,          DATO,                  |FOR A BUS "DATO" CYCLE
17109 |      NEXT,      J/GETDBUF730A
17110 | (4331) DCS(0.00,0.0,0.1) BM(1010.11,01.11,01.100.000...0.1,0.1,0.0...1.0010...0.0000,0...11.000...011.011,010)
17111 |
17112 | 4332: 1(FREE)
17113 | GETDBUF730A:
17114 |      PO,          BUMP-VERIFY,          |COUNT
17115 |      P3,          CSPD[17]_BUSDIN, EMIT/037777,    |DBUF SHOULD BE ENABLED; EMIT IS NOISE
17116 |
17117 |      P3=T,        D_JUNK, SAVE=D(C),          |MANGLE DATA IN D, DONT CARE WHAT RESULTS
17118 |      NEXT,      J/COMP730A
17119 | (4332) DCS(0.00,0.0,0.11) BM(0011.10,11.11,11.111.111...1.1,0.0,0.0...0.0000...1.0000,0...11.000...011.011,011)
17120 |
17121 | 4333: 1(FREE)
17122 | COMP730A:
17123 |      PO,          BUMP-VERIFY,          |COUNT
17124 |      P2=T,        D_CSPD[17]-XDR-ASPHI(C125252),    |COMPARE RECEIVED:EXPCTED
17125 |
```

```

17119 NEXT, J/ZAPDBUF730A
(4333) DCS[0.00,0.0,0.1] BM[0110..10.00..11.01..110..000...0.1.0..0..0.0000...0..0000,0...11.000...011,011,100]
17120
17121 4334: I(FREE)
17122 ZAPDBUF730A:
17123 P3, DBUF_D-[I], ICOPY ZEROED(?) D-REG INTO DBUF
17124 NEXT, J/GOBUT730A
(4334) DCS[0.00,0.0,0.0] BM[0100..00.00..00.00..000...100...0.0.0..0..0..1.1011...0.0000,0...11.000...011,011,101]
17125
17126 4335: I(FREE)
17127 GOBUT730A:
17128 SETUP, RETURN/TEST730B, IRETURN TO START OF NEXT SUBTEST
17129 NEXT, GOTO-PAGE(7), ISET TABLE
17130 J/BUTD-18-ZERO ICHECK FOR EQUALITY
(4335) DCS[0.00,0.0,0.0] BM[0100..00.10..01.10..110..111...0.0.0..0..0..0.0000...0..0000,0...11.100...011,100,001]
17131
17132
17133
17134
17135
17136
17137
17138
17139
17140 ITEST-730-B DOES A DATIP, AND THEN CHECKS THAT:
17141 I 1) NO ODD ADDRESS ERROR RESULTS
17142 I 2) THE RIGHT DATA (OUTPUT ABOVE) IS RETRIEVED (NOTE DBUF LATCH WAS ZEROED
17143 I TO ALTER ITS COPY OF THE DATA)
17144 I 3) THE BUS HOLDING FUNCTION OF THE "DATIP" SHOULD BE EMPLOYED; "BBY" SHOULD REMAIN ASSERTED
17145 I WELL PAST THE "NORMAL" 1 MICROWORD AFTER THE BUS CYCLE, IN FACT, IT SHOULD REMAIN ASSERTED
17146 I (HOLDING BUSDIN=UNIBUS-DATA-BUFFER, NOT EMIT, ETC) UNTIL CLEARED BY ANOTHER BUS CYCLE
17147 I (NOT A DATIP), OR A BUTA(LAST) (DONE HERE).
17148 4466:
17149 TEST730B:
17150 P0, LOAD=ENUA(ZTARGET402), ISETUP FOR D=ZERO COMPARE
17151 LOAD=ERROR(TEST730B), IERROR DIRECTORY KEY
17152 DCS=CTR(C7), ICOMPARE AT TARGET
17153 NFXT, J/EXPECT730B
(4466) DCS[1.00,1.0,0.0] BM[1000..00.11..11.00..000...010...0.0.0..0..0..0.0000...0..0000,0...11.000...011,011,110]
17154
17155 4336: I(FREE)
17156 EXPECT730B:
17157 P3, CSPD[02]_EMIT, EMIT/000340, ISERVICE PORT BITS FOR A "DATIP"
17158 NEXT, J/BUSFCN730B
(4336) DCS[0.00,0.0,0.0] BM[0000..10.00..00.11..100..000...0.0.0..0..0..0.1101...1..0000,0...11.000...011,011,111]
17159
17160 4337: I(FREE)
17161 BUSFCN730B:
17162 P1, BA_ASPHI(C000001), IUSE MEMORY ADDR(000001), ODD BYTE
17163 P2-T, D_ZERO, D[CU]_ALU15, IZAP D
17164 P3, DATIP, IFOR A BUS "DATIP" CYCLE
17165 BUTA(CLR=FLAG-RES-UCON), IRESET BUSDIN TO EMIT/ZAP DBUF_D UCON
17166 NFXT, J/GETIT730B

```

```

(4337) DCS[0.00,0.0,0.0] BM[0011..00.00..11.01..000...100...0.1.0..1.0..1.0100...0..0000,0...11.010...011,100,000]
17167
17168 4340: I(FREE)
17169 GETIT730B:
17170 P0, BUMP-VERIFY, ICOUNT
17171 P3, CSPD[17]_BUSDIN, EMIT/092525, IUNIBUS DATA SHOULD BE ENABLED; EMIT IS NOISE
17172 NFXT, J/GOBUT730B
(4340) DCS[0.00,0.0,0.1] BM[0101..10.01..01.01..010..101...0.0.0..0..0..0.0000...1..0000,0...11.000...011,100,001]
17173
17174 4341: I(FREE)
17175 GORUT730B:
17176 SETUP, RETURN/TEST730C, IEXEC SUBR WHICH:
17177 NFXT, CALL[CSP17XOR125252] I CSP(17)-XOR=(125252) -> D, BUT(D=ZERO)
(4341) DCS[0.00,0.0,0.0] BM[0100..00.10..01.10..000...111...0.0.0..0..0..0.0000...0..0000,0...11.100...010,000,011]
17178
17179
17180
17181
17182
17183
17184
17185 ITEST-730-C NOW CHECKS THAT THE "DATIP" FUNCTION IS STILL HOLDING THE BUS BY
17186 I VERIFYING THAT THE UNIBUS DATA BUFFER IS STILL ENABLED ON BUSDIN, DATA=(125252)
17187 4460:
17188 TEST730C:
17189 P0, LOAD=ENUA(ZTARGET402), ISETUP FOR D=ZERO COMPARE
17190 LOAD=ERROR(TEST730C), IERROR DIRECTORY KEY
17191 DCS=CTR(C5), ICOMPARE AT TARGET
17192 BUMP-VERIFY, ICOUNT
17193 NFXT, J/GORUT730C
(4460) DCS[1.00,1.0,0.1] BM[1010..00.11..11.00..000...010...0.0.0..0..0..0.0000...0..0000,0...11.000...011,100,010]
17194
17195 4342: I(FREE)
17196 GORUT730C:
17197 SETUP, RETURN/TEST730C1, IEXEC SUBR WHICH:
17198 NEXT, CALL[BUSDINXOR125252] I 1) BUSDIN -> CSP(17), EMIT=(052525)
17199 (4342) DCS[0.00,0.0,0.0] BM[0100..00.10..01.01..000...111...0.0.0..0..0..0.0000...0..0000,0...11.100...001,111,111]
17200
17201
17202
17203
17204
17205
17206
17207
17208 ITEST-730-C1 NOW CHECKS THAT THE "DATIP" FUNCTION WILL RELEASE THE BUS BY
17209 I VERIFYING THAT THE UNIBUS DATA BUFFER IS NOT ENABLED ON BUSDIN, DATA=(125252),
17210 I AFTER ISSUING BUTA(LAST), WHICH SHOULD CLEAR THE DATIP/BBY FLOP
17211 4460:
17212 TEST730C1:
17213 P0, LOAD=ENUA(ZTARGET402), ISETUP FOR D=ZERO COMPARE
17214 LOAD=ERROR(TEST730C1), IERROR DIRECTORY KEY

```

```
17214          DCS-CTR(C5.),          ICOMPARE AT TARGET
17215          P3,          BUTA(LAST),          IACTIVE BUT EFFECT: CLEAR OUT DATIP
17216          NEXT,          J/GOBUT730C1
(4450) DCS[1.00.1.0.0.0] BM[1010.00.11.11.00.000.010.0.0.0.0.0.0.0000.0.0.0.0000.0.0.10.000.0.100.100.011]
17217
17218          4443:
17219          GORUT730C1:
17220          SETUP,          RETURN/TEST730D,          IEXEC SUBR WHICH:
17221          NEXT,          CALL(BUSBDINXOR052525)          I 1) BUSBDIN -> CSP(17), EMIT=(052525)
17222          (4443) DCS[0.00.0.0.0.0] BM[0100.00.10.01.10.001.111.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.100.0.010.000.100]
17223
17224
17225
17226
17227          I - - - - -
17228
17229          ITEST-730-D DOES A DATO FOLLOWED BY A DATIB=BYTE+ODD, AND THEN CHECKS THAT THE RIGHT DATA
17230          I IS WRITTEN/RETRIEVED FROM MEMORY LOCATIONS (000000)/(000001)
17231          I AND THAT THE EMIT-DISABLE/UNIBUS DATA BUFFER ENABLE IS HANDLED CORRECTLY
17232          4461:
17233          TEST730D:
17234          P0,          LOAD-ENUA(ZTARGET402),          ISETUP FOR D=ZERO COMPARE
17235          LOAD-ERROR(TEST730D),          IERROR DIRECTORY KEY
17236          DCS-CTR(C15.),          I HOLD UP FOR NOW
17237          NEXT,          J/LOADDATA730D
(4461) DCS[1.00.1.0.0.0] BM[0000.00.11.11.00.000.010.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.000.0.011.100.011]
17238
17239          4343: I(FREE)
17240          LOADDATA730D:
17241          P3,          CSPD[16]_EMIT, EMIT/052525,          IPATTERN (052525) IN BASCON AREA
17242          NEXT,          J/SETZER730D
(4343) DCS[0.00.0.0.0.0] BM[0101.10.01.01.01.010.101.0.0.0.0.0.0.0.0001.0.0.0.0000.0.0.11.000.0.011.100.100]
17243
17244          4344: I(FREE)
17245          SFTZER730D:
17246          P1,          BA_ASPHI(C000000),          IUSE MEMORY ADDR(000000)
17247          P2-T,          D_CSPB(B16), D[C]_0,          IUSE DATA (052525)
17248          P3,          DATO,          I FOR A BUS "DATO" CYCLE
17249          NEXT,          J/MANGLED730D
(4344) DCS[0.00.0.0.0.0] BM[1010.11.01.11.01.100.000.0.0.1.0.0.1.0010.0.0.0.0000.0.0.11.000.0.011.100.101]
17250
17251          4345: I(FREE)
17252          MANGLED730D:
17253          P3-T,          D_ASPHI(C125252), SAVE-DIC],          IMANGLE D CONTENTS, AFTER/AT P3-T
17254          NEXT,          J/BUSFCN730D
(4345) DCS[0.00.0.0.0.0] BM[1111.00.00.11.01.110.111.0.0.1.0.0.0.0.0000.0.0.0.0000.0.0.11.000.0.011.100.110]
17255
17256          4346: I(FREE)
17257          BUSFCN730D:
17258          P0,          RUMP-VERIFY,          I COUNT
17259          DCS-CTR(C12.),          ICOMPARE AT TARGET
17260          P1,          BA_ASPHI(C000001),          IADDRESS ODD BYTE
```

```
17261          P3,          DATIB,          I BYTE READ -> PLAIN DATI; NO ODD ADDRESS ERROR
17262          NEXT,          J/GETIT730D
(4346) DCS[0.00.1.0.0.1] BM[0011.00.00.11.01.000.000.0.0.0.0.0.1.0011.0.0.0.0000.0.0.11.000.0.011.100.111]
17263
17264          4347: I(FREE)
17265          GETIT730D:
17266          P3,          CSPD[17]_BUSBDIN, EMIT/000000,          IUNIBUS DATA SHOULD BE ENABLED; EMIT IS NOISE
17267          NEXT,          J/GOBUT730D
(4347) DCS[0.00.0.0.0.0] BM[0000.10.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.000.0.011.101.000]
17268
17269          4350: I(FREE)
17270          GORUT730D:
17271          SETUP,          RETURN/TEST730E,          I GO TO SUBR WHICH:
17272          NEXT,          CALL(CLRSEVICETOD)          I (SERVICE)-XOR-(000140) -> D, BUT(D=ZERO)
(4350) DCS[0.00.0.0.0.0] BM[0100.00.10.01.10.010.111.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.100.0.010.100.100]
17273
17274
17275
17276
17277
17278
17279
17280          I - - - - -
17281
17282
17283          ITEST-730-E NOW CHECKS THAT THE BUS FUNCTION ABOVE ACTUALLY RETRIEVED THE RIGHT DATA:
17284          I THE (052525) WRITTEN TO MEMORY LOCATION (000000) IN TEST730D
17285          4462:
17286          TEST730E:
17287          P0,          LOAD-ENUA(ZTARGET402),          ISETUP FOR D=ZERO COMPARE
17288          LOAD-ERROR(TEST730E),          IERROR DIRECTORY KEY
17289          DCS-CTR(C4.),          ICOMPARE AT TARGET
17290          NEXT,          J/GOBUT730E
(4462) DCS[1.00.1.0.0.0] BM[1011.00.11.11.00.000.010.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.000.0.011.101.001]
17291
17292          4351: I(FREE)
17293          GORUT730E:
17294          SETUP,          RETURN/SCOPE730,          IEXEC SUBR WHICH:
17295          NEXT,          CALL(CSP17XOR052525)          I CSP(17)-XOR-(052525) -> D, BUT(D=ZERO)
(4351) DCS[0.00.0.0.0.0] BM[0100.00.01.11.01.010.111.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.100.0.010.000.101]
17296
17297
17298          4352: I(FREE)
17299          SCOPE730:
17300          NEXT,          BUTD(SCOPE),          I NO ERROR: "TEST731A" (+1, WORDS)
17301          J/TEST731A          I ERROR: "LOADIR730A" (-24, WORDS)
(4352) DCS[0.00.0.1.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0.0000.0.0.0.0000.0.0.11.000.0.111.101.101]
17302
17303
17304
17305
17306
```

```
17307
17308
17309
17310
17311
17312
17313
17314
17315
17316
17317
17318
17319
17320
17321
17322
17323
17324
17325
17326
17327
17328
17329
17330
17331
17332
17333
17334
17335
17336
17337
17338
17339
17340
17341
17342
17343
17344
17345
17346
17347
17348
17349
17350
17351
17352
17353
17354

! - - - - -
THIS SECOND SERIES OF TESTS DOES A DATOB/DATI/DATI-CLKIR SEQUENCE, CHECKING THAT
EACH FUNCTION OPERATES AS EXPECTED.
! - - - - -
TEST-731-A DOES A DATOB*BYTE*ODD, AND THEN CHECKS THAT THE DBUF LATCH (DS) ALSO GETS LOADED WITH THE
DATA, AND THAT IT IS ENABLED ON BUSDIN IN THE MICROWORD AFTER THE BUS CYCLE (IE, EMIT
IS TEMPORARILY DISABLED)
47551
TEST731A:
PO, LOAD=ENUA(ZTARGET402), !SETUP FOR D=ZERO COMPARE
LOAD=ERROR(TEST731A), !ERROR DIRECTORY KEY
DCS=CTR(C6.), !COMPARE AT TARGET
NEXT, J/BUSFCN731A
(4755) DCS(1.00,1.0,0.0) BM(1001.00,11.11.00.000.010...0.0.0.0.0...0.0000...0.0000,0...11.000...111.110,100)
47641
RUSFCN731A:
P1, BA_ASPI(C000001), !USE MEMORY ADDR(000001), ODD BYTE
P2=T, D_ZERO, D(C)_ALU15, !USE DATA (000), ONE BYTE (ODD) ONLY
P3, DATOB, !FOR A BUS "DATOB*BYTE*ODD" CYCLE
NEXT, J/GETDBUF731A
(4764) DCS(0.00,0.0,0.0) BM(0011.00,00.11.01.000.100...0.1.0.1.0...1.0101...0.0000,0...11.000...011.101,011)
47531
GFTDBUF731A:
PO, BUMP-VERIFY, !COUNT
P3, CSPD(17)_BUSDIN, !DBUF SHOULD BE ENABLED; EMIT IS NOISE
P1=T, D_ASPI(C17777), SAVE=D(C), !HANDLE DATA IN D, DON'T CARE WHAT RESULTS
NEXT, J/COMP731A
(4753) DCS(0.00,0.0,0.0) BM(1111.10,00.11.01.101.111...1.1.0.0.0...0.0000...1.0000,0...11.000...011.101,100)
47541
COMP731A:
P2=T, D_CSPD(D17), SAVE=D(C), !COMPARE RECEIVED:(000000)
NEXT, J/GOBUT731A
(4754) DCS(0.00,0.0,0.0) BM(1010.10,00.00.00.000.111...0.1.0.0.0...0.0000...0.0000,0...11.000...011.101,101)
47551
GOBUT731A:
SETUP, RETURN/TEST731B, !RETURN TO START OF NEXT SUBTEST
NEXT, GOTO-PAGE(7), !BUT TABLE
J/BUTD-18-ZERO !CHECK FOR EQUALITY
(4755) DCS(0.00,0.0,0.0) BM(0100.00,10.01.01.111.111...0.0.0.0.0...0.0000...0.0000,0...11.100...011.100,001)
17351
17352
17353
17354
```

```
17355
17356
17357
17358
17359
17360
17361
17362
17363
17364
17365
17366
17367
17368
17369
17370
17371
17372
17373
17374
17375
17376
17377
17378
17379
17380
17381
17382
17383
17384
17385
17386
17387
17388
17389
17390
17391
17392
17393
17394
17395
17396
17397
17398
17399
17400
17401
17402
17403

! - - - - -
TEST-731-B NOW CHECKS THAT NONE OF THE ABOVE BUS FUNCTIONS HAS ALTERED THE IR FROM WHEN
IT WAS LOADED IN TEST730A, WITH THE VALUE (125200), INSTR5-E08(412) DECODE
44571
TEST731B:
PO, LOAD=ENUA(ZTARGET412), !SETUP FOR INSTR5/E08 DECODE
LOAD=ERROR(TEST731B), !ERROR DIRECTORY KEY
DCS=CTR(C5.), !COMPARE AT TARGET
NEXT, J/BUSFCN731B
(4457) DCS(1.00,1.0,0.0) BM(1010.00,11.11.00.001.010...0.0.0.0.0...0.0000...0.0000,0...11.000...011.101,110)
43561
RUSFCN731B:
P1, BA_ASPI(C000000), !USE MEMORY ADDR(000000)
P3, DATI, !FOR A BUS "DATI" CYCLE
NEXT, J/GETIT731B
(4356) DCS(0.00,0.0,0.0) BM(0000.00,00.11.01.100.000...0.0.0.1.0...1.0110...0.0000,0...11.000...011.101,111)
43571
GFTIT731B:
PO, RUMP-VERIFY, !COUNT
P3, CSPD(17)_BUSDIN, EMIT/152500, !UNIBUS DATA SHOULD BE ENABLED; EMIT IS NOISE
NEXT, J/GOBUT731B ! THIS DATA IS INSTR5-E08(405) DECODE
(4357) DCS(0.00,0.0,0.0) BM(1101.10,01.01.01.000.000...0.0.0.0.0...0.0000...1.0000,0...11.000...011.110,000)
43601
GOBUT731B:
SETUP, RETURN/TEST731C, !RETURN TO START OF NEXT SUBTEST
NEXT, GOTO-PAGE(7), !BUT TABLE
J/RUTINSTR5 !CHECK THAT IF STILL HAS DATA (125200)
(4360) DCS(0.00,0.0,0.0) BM(0100.00,10.01.01.110.111...0.0.0.0.0...0.0000...0.0000,0...11.100...011.000,001)
17386
17387
17388
17389
17390
17391
17392
17393
17394
17395
17396
17397
17398
17399
17400
17401
17402
17403

! - - - - -
TEST-731-C NOW CHECKS THAT THE BUS FUNCTION ABOVE ACTUALLY RETRIEVED THE RIGHT DATA:
THE (052524) WRITTEN TO MEMORY LOCATION (001)/(000) IN TEST730D
AND THE (000) WRITTEN TO MEMORY LOCATION (001) IN TEST731A
TOGETHER THESE FORM A (000125) IN MEMORY LOCATION (001)(000): INSTR5-E78(432) DECODE
44561
TEST731C:
PO, LOAD=ENUA(ZTARGET432), !SETUP FOR INSTR5-E78(412) DECODE
LOAD=ERROR(TEST731C), !ERROR DIRECTORY KEY
```

```
17404          DCS-CTR(C7),          ICOMPARE AT TARGET
17405          BUMP-VERIFY,          ICOUNT
17406          NEXT, J/COMP731C
(4456) DCS(1.00,1.0,0.0) BM(1000.00,11.11.00,011.010...0.0,0.0,0.0...0.0000...0.0000,0...11.000...011.110,001)
17407
17408          4361: 1(FREE)
17409          COMP731C:
17410          P2=U, D_CSPD(017), DIC=0,          IGET DATA READ FROM DATI, ABOVE
17411          NEXT, J/GOBUT731C
(4361) DCS(0.00,0.0,0.0) BM(1010.10,00.00,000.000...0.1,0.0,0.0...0.0000...0.0000,0...11.000...011.110,010)
17412
17413          4362: 1(FREE)
17414          GOBUT731C:
17415          SETUP, RETURN/TEST731D,          IRETURN TO START OF NEXT SUBTEST
17416          NEXT, CALL(DINTOIR=5)          ISUBR FOR: P -> IR, BUT(INSTRS)
(4362) DCS(0.00,0.0,0.0) BM(1010.00,10.01.01.101.111...0.0,0.0,0.0...0.0000...0.0000,0...11.100...010.111,011)
17417
17418
17419
17420
17421
17422 | - - - - -
17423
17424 |TEST-731-D DOES A DATI-CLKIR, AND THEN CHECKS THAT THE RIGHT DATA
17425 | IS WRITTEN/RETRIEVED FROM MEMORY LOCATIONS (000000)/(000001)
17426 | AND THAT THE EMIT-DISABLE/UNIBUS DATA BUFFER ENABLE IS HANDLED CORRECTLY
17427 4455:
17428 TEST731D:
17429          PO,          LOAD-ENUA(ZTARGET402),          ISETUP FOR D=ZERO COMPARE
17430          LOAD-ERROR(TEST731D),          IERROR DIRECTORY KEY
17431          DCS-CTR(C14),          ICOMPARE AT TARGET
17432          NEXT, J/LOADIR731D
(4455) DCS(1.00,1.0,0.0) BM(0001.00,11.11.00,000.010...0.0,0.0,0.0...0.0000...0.0000,0...11.000...011.110,011)
17433
17434          4363: 1(FREE)
17435          LOADIR731D:
17436          P2=U,          IR_EMIT, EMIT/125200,          IPREV DATA IN IR FOR INSTRS-E88(412)
17437          NEXT, J/BUSFCN731D
(4363) DCS(0.00,0.0,0.0) BM(1010.00,10.10.10,000.000...0.0,0.0,0.0...1.1010...0.0000,0...11.000...011.110,100)
17438
17439          4364: 1(FREE)
17440          BUSFCN731D:
17441          P1,          BA_ASPhi(C000000),          IFROM LOCATION (000000)
17442          P3,          DATI-CLKIR,          IDO A DATI, AND CLKIR
17443          NEXT, J/MANGLED731D
(4364) DCS(0.00,0.0,0.0) BM(0000.00,00.11.01.100.000...0.0,0.1,0.0...1.0000...0.0000,0...11.000...011.110,101)
17444
17445          4365: 1(FREE)
17446          MANGLED731D:
17447          P2=U,          IIR_DATA,          IIR SHOULD GET DATA HERE
17448          EMIT/152500,          INOISE ON EMIT: INSTRS-E88(405)
17449          NEXT, J/GOBUT731D
(4365) DCS(0.00,0.0,0.0) BM(1101.00,01.01.01.000.000...0.0,0.0,0.0...0.0000...0.0000,0...11.000...011.110,110)
```

```
17450
17451          4366: 1(FREE)
17452          GOBUT731D:
17453          SETUP, RETURN/TEST731E,          IGO TO SUBR WHICH:
17454          NEXT, CALL(CLSERVICE)          I (SERVICE) XOR=(000340) -> D, BUT(D=ZERO)
(4366) DCS(0.00,0.0,0.0) BM(1010.00,10.01.01.100.111...0.0,0.0,0.0...0.0000...0.0000,0...11.100...010.100,100)
17455
17456
17457
17458
17459
17460
17461
17462 | - - - - -
17463
17464
17465 |TEST-731-E NOW CHECKS THAT THE BUS FUNCTION ABOVE ACTUALLY RETRIEVED THE RIGHT DATA:
17466 4454:
17467 TEST731E:
17468          PO,          LOAD-ENUA(ZTARGET432),          ISETUP FOR INSTRS-E78(432) DECODE
17469          LOAD-ERROR(TEST731E),          IERROR DIRECTORY KEY
17470          DCS-CTR(C3),          ICOMPARE AT TARGET
17471          NEXT, J/GOBUT731E
(4454) DCS(1.00,1.0,0.0) BM(1100.00,11.11.00,011.010...0.0,0.0,0.0...0.0000...0.0000,0...11.000...011.110,111)
17472
17473          4367: 1(FREE)
17474          GOBUT731E:
17475          SETUP, RETURN/SCOPE731,          IRETURN TO SCOPE LOOP TEST WORD
17476          NEXT, GOTO-PAGE(7),          IBUT TABLE
17477          J/BUTINSTRS          ICHECK FOR RIGHT DATA
(4367) DCS(0.00,0.0,0.0) BM(1010.00,01.11.11.000.011...0.0,0.0,0.0...0.0000...0.0000,0...11.100...011.000,001)
17478
17479          4370: 1(FREE)
17480          SCOPE731:
17481          P2=U,          IR_EMIT, EMIT/125200,          IPRESET IR FOR SCOPE LOOP
17482          NEXT, RUTD(SCOPE),          INO ERROR: "LOADIR740A" (+5, WORDS)
17483          J/LOADIR740A          I ERROR: "BUSFCN731A" (-18, WORDS)
(4370) DCS(0.00,0.1,0.0) BM(1010.00,10.10.10,000.000...0.0,0.0,0.0...1.1010...0.0000,0...11.000...111.110,101)
17485
17486
17487
17488
17489 | THESE SUBROUTINES ARE USED IN THE PREVIOUS TESTS:
17490
17491          7177: 1(FREE)
17492
17493          RDX12:
17494          P3,          CSPD(17)_BUSDIN, EMIT/052525,          IENTRY FOR "BUSDINXOR125252"
17495          NEXT, J/C17X12
(7177) DCS(0.00,0.0,0.0) BM(1010.10,01.01.01.010.101...0.0,0.0,0.0...0.0000...1.0000,0...11.000...010.000,011)
17496          7203: 1(PREF)
```

```

17497 C17X12:
17498 P2-T, D_CSPD[17]-XOR-ASPHI(C12S252), ENTRY FOR "CSP17XOR12S252"
17499 SAVE-D[C],
17500 NEXT, J/BUTD-IS-ZERO IFINISH OUT BY TESTING RESULT OF COMPARISON
(7203) DCS[0.00.0.0.0.0] BM[0110..10.00..11.01..110..111...0.0.0.0.0...0.0000...0.0000.0...11.000...011.100.001]

17501
17502 7204: 1(FREE)
17503 BDX05:
17504 P3, CSPD[17]-BUSPIN, EMIT/052525, ENTRY FOR "BUSPINXOR052525"
17505 NEXT, J/C17X05
17506 (7204) DCS[0.00.0.0.0.0] BM[0101..10.01..01.01..010..101...0.0.0.0.0...0.0000...1.0000.0...11.000...010.000.101]

17507
17508 7205: 1(FREE)
17509 C17X05:
17510 P2-T, D_CSPD[17]-XOR-ASPHI(C052525), ENTRY FOR "CSP17XOR052525"
17511 SAVE-D[C],
17512 NEXT, J/BUTD-IS-ZERO IFINISH OUT BY TESTING RESULT OF COMPARISON
(7205) DCS[0.00.0.0.0.0] BM[0110..10.00..11.01..111..111...0.0.0.0.0...0.0000...0.0000.0...11.000...011.100.001]

17513
17514
17515
17516
17517
17518
17519
17520 1.PAGE=====
17521
17522
17523 .T0C * TEST740: BUS CYCLE MODIFICATION - PREFETCH ALTERATION, OVERLAP YANK
17524
17525 |*****
17526 |*
17527 |* TESTS: 740A - 740D UNWORDS: 020 + 043
17528 |*
17529 |* FUNCTIONS:
17530 |*
17531 |* THERE FOUR TESTS CHECK THAT LOGIC CONCERNED WITH:
17532 |*
17533 |* 1) BCO/PREFETCH = BCO[1] * NOT(PREFETCH*BUS[INSTR1])
17534 |* IE, ALTER BUS CODE (**1) -> (**0), IN A PREFETCH SITUATION
17535 |*
17536 |* AND 2) BEGIN-DATA-XFER = BEGIN[BUS/UCOM] * NOT(PULSE-SUPPRESS[0]) * NOT(INH-SPLO)
17537 |* IE, "YANK" [STOP] A BUS CYCLE IN A NON-OVERLAP SITUATION
17538 |*
17539 |*****
17540
17541
17542
17543
17544
17545
17546
17547

```

```

17548 1TEST-740-A CHECKS THAT BC[0]-H DOES NOT GET ALTERED FROM (1) -> (0) WHEN:
17549 1 BUTA[INSTR1]-H IS ASSERTED, BUT PREFETCH-H IS NEGATED
17550 4765:
17551 LOADIR740A:
17552 P2-U, IR_EMIT, EMIT/056000, 1(OVERLAP,-PREFETCH), (-BYTE,DOP,-SMO)
17553 NEXT, J/TEST740A
(4765) DCS[0.00.0.0.0.0] BM[0101..00.11..00.00..000..000...0.0.0.0.0...1.1010...0.0000.0...11.000...111.100.100]

17554
17555 4744:
17556 TEST740A:
17557 P0, LOAD-ENUA(ZTARGET402), 1SETUP FOR D=ZERO COMPARE
17558 LOAD-ERROR(TEST740A), 1ERROR DIRECTORY KEY
17559 DCS-CTR(C12.), 1COMPARE AT TARGET
17560 NEXT, J/BUSFCN740A
(4744) DCS[1.00.1.0.0.0] BM[0011..00.11..11.00..000..010...0.0.0.0.0...0.0000...0.0000.0...11.000...011.111.001]

17561
17562 4371: 1(FREE)
17563 BUSFCN740A:
17564 P0, BUMP-VERIFY, 1COUNT
17565 P1, BA-ASPHI(C000000), 1USE MEMORY ADDR (000000)
17566 P2-T, SR_ZERO, 1SR<00>=(0), FOR UNEXPECTED JAMUPPS
17567 P3, DATS, 1CODE=(3)/DATS, POSSIBLY ALTERED TO (2)/DATS
17568 NEXT, J/GOBUT740A 1ACTIVE EFFECT ONLY - BRANCH MASKED
17569 (4371) DCS[0.00.0.0.0.1] BM[0011..00.00..11.01..100..000...0.0.0.1.1...0.0011...0.0000.0...00.110...111.111.110]

17570
17571 4776:
17572 GORUT740A:
17573 SFTUP, RETURN/LOADIR740B, 1GO TO SUBR WHICH:
17574 NEXT, CALL(DATISERVICETOD) 1 CLR-JAM-ERRORS, FOR INSURANCE
17575 (4776) DCS[0.00.0.0.0.0] BM[0100..00.01..11.11..010..111...0.0.0.0.0...0.0000...0.0000.0...11.100...010.100.101]

17576
17577
17578
17579
17580
17581
17582
17583 1TEST-740-B CHECKS THAT BC[0]-H DOES GET ALTERED FROM (1) -> (0) WHEN:
17584 1 BUTA[INSTR1]-H IS ASSERTED, AND PREFETCH-H IS ASSERTED
17585 4372: 1(FREE)
17586 LOADIR740B:
17587 P2-U, IR_EMIT, EMIT/020606, 1(OVERLAP,PREFETCH), (-BYTE,DOP,SMO)
17588 NEXT, J/TEST740B
(4372) DCS[0.00.0.0.0.0] BM[0010..00.00..01.10..000..110...0.0.0.0.0...1.1010...0.0000.0...11.000...100.101.011]

17589
17590 4453:
17591 TEST740B:
17592 P0, LOAD-ENUA(ZTARGET402), 1SETUP FOR D=ZERO COMPARE
17593 LOAD-ERROR(TEST740B), 1ERROR DIRECTORY KEY
17594 DCS-CTR(C12.), 1COMPARE AT TARGET
17595 NEXT, J/BUSFCN740B

```

```
(4453) DCS[1.00,1.0,0,0] BM[0011..00.11..11.00..000..010...0,0,0,0...0.0000...0.0000,0...11.000...011.111,011]
17596
17597 4373: 1(FREE)
17598 BUSFCN740B:
17599 P1, BA_ASPHI(C000000), IUSE MEMORY ADDR (000000)
17600 P3, DATB, ICODE=(3)/DATB, POSSIBLY ALTERED TO (2)/DATO
17601 NEXT, J/GOBUT740B IACTIVE EFFECT ONLY - BRANCH MASKED
17602 (4373) DCS[0.00,0.0,0,0] BM[0000..00.00..11.01..100..000...0,0,0,0...1.0011...0.0000,0...00.110...111.110,110]
17603
17604 4766:
17605 GORUT740B:
17606 SETUP, RETURN/TEST740C, IGO TO SUBR WHICH:
17607 NEXT, CALL(DATOSERVICETOD) I CLR-JAN-ERRORS, FOR INSURANCE
17608 (4766) DCS[0.00,0.0,0,0] BM[0100..00.10..01.01..010..111...0,0,0,0...0.0000...0.0000,0...11.100...010.100,110]
17609
17610
17611
17612
17613
17614 ! - - - - -
17615
17616 ITEST-740-C CHECKS THAT BC(0)-H DOES NOT GET ALTERED FROM (1) -> (0) WHEN:
17617 I BUTA[INSTR1]-H IS NEGATED, BUT PREFETCH-H IS ASSERTED
17618 4452:
17619 TEST740C:
17620 PO, LOAD-ENUA(ZTARGET402), ISETUP FOR D=ZERO COMPARE
17621 LOAD-ERROR(TEST740C), IERROR DIRECTORY KEY
17622 DCS-CTR(C12,), ICOMPARE AT TARGET
17623 BUMP-VERIFY, ICOUNT
17624 NEXT, J/BUSFCN740C IIR AS ABOVE, (OVERLAP,PREFETCH), (-BYTE,DOP,SMO)
(4452) DCS[1.00,1.0,0,0] BM[0011..00.11..11.00..000..010...0,0,0,0...0.0000...0.0000,0...11.000...011.111,100]
17625
17626 4374: 1(FREE)
17627 BUSFCN740C:
17628 P1, BA_ASPHI(C000000), IUSE MEMORY ADDR (000000)
17629 P3, DATB, ICODE=(3)/DATB, POSSIBLY ALTERED TO (2)/DATO
17630 NEXT, J/GOBUT740C I(020606) TARGETS TO (042), MASKED UNDER (766)
(4374) DCS[0.00,0.0,0,0] BM[0000..00.00..11.01..100..000...0,0,0,0...1.0011...0.0000,0...11.000...011.111,101]
17631
17632 4375: 1(FREE)
17633 GORUT740C:
17634 SETUP, RETURN/LOADIR740D, IGO TO SUBR WHICH:
17635 NEXT, CALL(DATISERVICETOD) I CLR-JAN-ERRORS, FOR INSURANCE
17636 (4375) DCS[0.00,0.0,0,0] BM[0100..00.10..00.00..000..111...0,0,0,0...0.0000...0.0000,0...11.100...010.100,101]
17637
17638
17639
17640
17641
```

```
17642 ! - - - - -
17643
17644 ITEST-740-D CHECKS THAT WHEN BUTA[INSTR1]-H IS ASSERTED, AND OVERLAP=L IS
17645 I NEGATED, THEN THE BUS CYCLE IS NOT EVEN ALLOWED TO BEGIN
17646 4400: 1(FREE)
17647 LOADIR740D:
17648 PO, BUMP-VERIFY, ICOUNT
17649 P2-U, IR_FMIT, EMIT/076000, I(-OVERLAP,-PREFETCH), (-BYTE,-DOP,-SOP)
17650 NEXT, J/TEST740D
(4400) DCS[0.00,0.0,0,0] BM[0111..00.11..00.00..000..000...0,0,0,0...1.1010...0.0000,0...11.000...100.101,001]
17651
17652 4451:
17653 TEST740D:
17654 PO, LOAD-ENUA(ZTARGET402), ISETUP FOR D=ZERO COMPARE
17655 LOAD-ERROR(TEST740D), IERROR DIRECTORY KEY
17656 DCS-CTR(C11,), ICOMPARE AT TARGET
17657 NEXT, J/BUSFCN740D
(4451) DCS[1.00,1.0,0,0] BM[0100..00.11..11.00..000..010...0,0,0,0...0.0000...0.0000,0...11.000...100.000,001]
17658
17659 4401: 1(FREE)
17660 BUSFCN740D:
17661 P1, BA_ASPHI(C000001), IUSE MEMORY ADDR (000001), TRY TO FORCE ODD ADDR
17662 P3, DATO, ITRY TO ALTER SERVICE FROM (100340)/DATI TO (002340)/DATO
17663 NEXT, J/GOBUT740D IACTIVE EFFECT ONLY - BRANCH MASKED
17664 (4401) DCS[0.00,0.0,0,0] BM[0000..00.00..11.01..000..000...0,0,0,0...1.0010...0.0000,0...00.110...111.110,111]
17665
17666 4767:
17667 GORUT740D:
17668 SETUP, RETURN/SCOPE740, IGO TO SUBR WHICH:
17669 NEXT, CALL(CJESERVICETOD) I [CSP(02) LOADED IN LAST TEST, SAME VALUE= (100340)]
17670 (4767) DCS[0.00,0.0,0,0] BM[0100..00.10..00.00..010..111...0,0,0,0...0.0000...0.0000,0...11.100...010.100,111]
17671
17672
17673
17674 4402: 1(FREE)
17675 SCOPE740:
17676 P2-U, IR_FMIT, EMIT/056000, IRELOAD IR FOR TEST740A
17677 NEXT, RUTD(SCOPE), I[NO ERROR: "TEST761A" (+1, WORDS)
17678 J/TEST761A I ERROR: "LOADIR740A" (-16, WORDS)
(4402) DCS[0.00,0.1,0,0] BM[0101..00.11..00.00..000..000...0,0,0,0...1.1010...0.0000,0...11.000...111.100,101]
17679
17680
17681
17682
17683
17684 I.PAGE=====
17685
17686
17687 .TDC * TEST761-763: TESTING UNIRUS INTERRUPT SERVICE WITH DL11-W LINE CLOCK
17688
17689 I=====
```

```
17690 |*
17691 |* TESTS: 761A - 763D
17692 |*
17693 |* FUNCTIONS: TESTS 761A - 763D CAUSE AN INTERRUPT ON THE UNIBUS AT
17694 |* LEVEL BR6, USING THE DL11-W LINE CLOCKS AS A GENERATOR,
17695 |* THEN TEST TO SEE THAT ALL THE APPROPRIATE BELLS AND
17696 |* WHISTLES AND PARAPHERNEALXJAYSIGURE ALSO RESPOND.
17697 |*
17698 |*****
17699 |
17700 |
17701 |
17702 |
17703 |
17704 |
17705 | - - - - -
17706 |
17707 | THIS FIRST SET OF THREE TESTS CLEAR OUT ALL THE I-O UCON REGISTERS;
17708 | CLR-JAM-ERRORS, CLR-YELLOW-ZONE, CLEAR-CONSOLE-SERVICE, CLR-MFR-TIMEOUT, CLR-PWR-FAIL, AND
17709 | ALSO THIS TIME DO AN "INIT" ON THE UNIBUS, VIA THE BUS-INIT-UCON FUNCTION.
17710 | ALSO SET ALL THE FLAGS<8:0>H=ZERO, AND THE FULL PS<15:00>H=ZERO,
17711 | (IE, PROCESSOR PRIO=0, T-BIT=0).
17712 |
17713 | THEN CHECK TO SEE THAT ALL THE RELEVANT BUS CONTROL INTERRUPT LOGIC IS RESET.
17714 |
17715 |
17716 |
17717 |
17718 |
17719 | - - - - -
17720 |
17721 | TEST 761A CHECKS THAT SERVICE-H=NOT(INTR-HIGH-H=FLAG7(0)H=BG-SERVICE(0)H) IS LOW,
17722 | WHEN INTR-HIGH-H=HIGH, FLAG7(0)H=HIGH, AND BG-SERVICE(0)H=HIGH
17723 | IE, AFTER UNIBUS-INIT, AND CLEAR SERVICE CONDITIONS, W/ PSW PRIO=000,
17724 | THERE SHOULD BE NOTHING PENDING
17725 |
17726 | 4745:
17727 | TEST761A:
17728 | P0, LOAD-ENUA(ZTARGET402), |BIT<00> CLEAR
17729 | LOAD-ERROR(TEST761A), |ERROR DIRECTORY KEY
17730 | DCS-CTR(C12), |COMPARE AT TARGET
17731 | P3, BUTA(CLR-FLAG-RES-UCON), |SET SR/LOAD, BUSDIN/EMIT
17732 | NEXT, J/ZERODSR761A |
17733 | (4745) DCS[1.00.1.0.0.0] BM[0011..00.11..11.00..0000..010..0.0.0..0.0..0.0000..0.0000.0...11.010...111.111.010]
17734 |
17735 | 4772:
17736 | ZERODSR761A:
17737 | P2-T, D_ZERO, SR_ZERO, D(C)_ALUIS, |ZERO FOR FLAGS/PS/JAMUPP FLAG
17738 | NEXT, J/ZEROIT761A |
17739 | (4772) DCS[0.00.0.0.0.0] BM[0011..00.00..00.00..0000..100..0.1.1..0.0..0.0000..0.0000.0...11.000...100.000.011]
17740 |
17741 | 4403: I(FREE)
17742 | ZEROIT761A:
17743 | P0, BUMP-VERIFY, |COUNT
17744 | BUSDIN_EMIT=I; |KEEP IT ON
```

```
17742 P3, FLAG[8-0]_D[15-8]=[I], |ZERO THE FLAGS
17743 PS_0=[I], |ZERO ALL OF THE PS
17744 NEXT, J/CLEAR761A |
17745 | (4403) DCS[0.00.0.0.0.1] BM[1000..00.00..00.01..010..011...0.0.0..0.0...1.1011...0..0000.0...11.000...100.000.100]
17746 |
17747 | 4404: I(FREE)
17748 | CLEAR761A:
17749 | SETUP, RETURN/GOBUT761A, |GO TO SUBR WHICH DOES THE CLEARS AND BUS-INIT
17750 | NEXT, CALL(CLEAR-I-O-A) |
17751 | (4404) DCS[0.00.0.0.0.0] BM[0100..00.10..00.00..101..111...0.0.0..0.0...0.0000...0..0000.0...11.100...010.010.111]
17752 |
17753 | 4405: I(FREE)
17754 | GORUT761A:
17755 | SETUP, RETURN/TEST761B, |RETURN TO START OF NEXT SUBTEST
17756 | NEXT, GOTO-PAGE(7), |BUT TABLE
17757 | J/BUTSERVICE |SERVICE-H IN BIT<00>
17758 | (4405) DCS[0.00.0.0.0.0] BM[0100..00.11..11.11..001..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.100.101]
17759 |
17760 |
17761 | - - - - -
17762 |
17763 | TEST 761B CHECKS THAT VECTOR-LOAD[1]H=-(UNIBUS-INTR-L) IS LOW
17764 |
17765 | 4771:
17766 | TEST761B:
17767 | P0, LOAD-ENUA(ZTARGET401), |BIT<01> CLEAR
17768 | LOAD-ERROR(TEST761B), |ERROR DIRECTORY KEY
17769 | DCS-CTR(C3), |COMPARE AT TARGET
17770 | NEXT, BUMP-VERIFY, |COUNT
17771 | J/GOBUT761B |
17772 | (4771) DCS[1.00.1.0.0.1] BM[1100..00.11..11.00..0000..001...0.0.0..0.0...0.0000...0..0000.0...11.000...100.000.110]
17773 |
17774 | 4406: I(FREE)
17775 | GOBUT761B:
17776 | SETUP, RETURN/TEST761C, |RETURN TO START OF NEXT SUBTEST
17777 | NEXT, GOTO-PAGE(7), |BUT TABLE
17778 | J/BUTVECTLOAD |VECTOR-LOAD[1]H IN BIT<01>
17779 | (4406) DCS[0.00.0.0.0.0] BM[0100..00.11..11.11..100..111...0.0.0..0.0...0.0000...0..0000.0...11.100...011.100.110]
17780 |
17781 |
17782 | - - - - -
17783 |
17784 | TEST 761C CHECKS THAT BG-SERVICE(0)H=BR>PS-I IS HIGH WHEN NO DEVICES REQUEST INTR ON UNIBUS
17785 |
17786 | 4774:
17787 | TEST761C:
17788 | P0, LOAD-ENUA(ZTARGET407), |BIT<02> SET (ACTIVE LOW)
```

```

17788      LOAD-ERROR(TEST761C),      !ERROR DIRECTORY KEY
17789      DCS-CTR(C3.),                !COMPARE AT TARGET
17790      NEXT,                          !
(4774) DCS[1.00.1.0.0.01] BM[1100..00.11..11.00..000..111...0.0.0.0.0.0.0.0000...0..0000.0...11.000...100.000.111]
17791      4407: 1(FREE)
17792      GOBUT761C:
17793      SETUP, RETURN/SCOPE761,      !RETURN TO SCOPE LOOP TEST WORD
17794      NEXT, GOTO-PAGE(7),          !BUT TABLE
17795      J/BUTBGSERVL                 !B6-SERVIS[0]H IN BIT<0>
17796      (4407) DCS[0.00.0.0.0.0] BM[0100..00.10..00.01..000..111...0.0.0.0.0.0.0.0000...0..0000.0...11.100...011.001.110]
17797
17798
17799      4410: 1(FREE)
17800      SCOPE761:
17801      P0, BUMP-VERIFY,              !COUNT
17802      P3, CSPDI03]_EMIT, EMIT/177546, !!(177546) IS UNIBUS ADDR OF CSR FOR DL11-W
17803      NEXT, BUTDISCOPE],            !NO ERROR: "TEST762A" (+1. WORDS)
17804      J/TEST762A                   ! ERROR: "SCOPE761A" (-8. WORDS)
17805      (4410) DCS[0.00.0.1.0.1] BM[1111..10.11..11.01..100..110...0.0.0.0.0.0.0.1010...1..0000.0...11.000...111.111.011]
17806
17807
17808
17809
17810
17811
17812
17813      !THE FOLLOWING FIVE TESTS NOW CAUSE AN INTERRUPT ON THE UNIBUS, AND THEN
17814      !CHECK THAT THE RUN CONTROL LOGIC RESPONDS CORRECTLY.
17815      !FIRST THE LINE CLOCK INTR ENABLE BIT<0> IS SET, AND THE PROCESSOR PRIORITY IS SET TO (6).
17816      !SINCE THE DL11-W IS A BR6 DEVICE, THE INTERRUPT SHOULD NOT COME THRU UNTIL THE PRIORITY IS
17817      !LOWERED TO LEVEL (5) OR LOWER. NOTE THAT THE MICROCODE MUST GO IN TO A WAIT LOOP FOR A
17818      !MINIMUM OF 16.7 MILLISEC @ 60 HZ, OR 20 MILLISEC @ 50 HZ TO GUARANTEE THAT AN INTERRUPT WILL
17819      !BE PENDING. THE DELAY, BASED UPON UP3-UP3 = 170 NS, WILL BE 32.3 MILLISEC, AN ADEQUATE MARGIN.
17820
17821
17822
17823
17824
17825
17826
17827      !TEST 762A ENABLES THE INTERRUPT, THEN WAITS FOR THE DELAY PERIOD.
17828      4773:
17829      TEST762A:
17830      P0, LOAD-ENUA(ZAPD762A),        !COMPARE JUST AFTER BUS CYCLE INVOKED
17831      LOAD-ERROR(TEST762A),          !ERROR DIRECTORY KEY
17832      DCS-CTR(C5.),                  !COMPARE JUST AFTER "DATO" INITIATED, BELOW
17833      NEXT, J/DW11L762A              !
(4773) DCS[1.00.1.0.0.0] BM[1010..00.10..00.11..001..101...0.0.0.0.0.0.0.0000...0..0000.0...11.000...100.001.001]
17834
17835      4411: 1(FREE)
17836      DW11L762A:

```

```

17837      P2-T, SR_CSPD(D05),          !GET SR(177546), ADDR OF DL11-W CSR
17838      NEXT, GOTO-PAGE(7),            !XFER
17839      J/MASK762A                      !
(4411) DCS[0.00.0.0.0.0] BM[1010..10.00..00.00..000..111...0.0.1.0.0.0.0.01010...0..0000.0...11.100...010.000.010]
17840
17841      7202: 1(FREE)
17842      MASK762A:
17843      P3, CSPD[04]_EMIT, EMIT/177777, !READ ALL BITS IN REGISTERS READ
17844      NEXT, J/PRI06762A              !
(7202) DCS[0.00.0.0.0.0] BM[1111..10.11..11.11..111...0.0.0.0.0.0.0.01011...1..0000.0...11.000...010.000.111]
17845
17846      7207: 1(FREE)
17847      PRI06762A:
17848      P3, CSPD[15]_EMIT,              !ENITCON FOR: PS<7:5>="110"=(6), T-BIT=0
17849      EMIT/000300,                    ! AND BIT<0> SET FOR DL11-W INTR ENABLE
17850      NEXT, J/GFTIT762A              !
(7207) DCS[0.00.0.0.0.0] BM[0000..10.00..00.11..000..000...0.0.0.0.0.0.0.0010...1..0000.0...11.000...010.001.000]
17851
17852      7210: 1(FREE)
17853      GFTIT762A:
17854      P1, BA_SR,                      !BA <- (177546), BITS<17:16> FORCED TO "11" ON IO-PAGE ADDR55
17855      P2-T, D_CSPB(B15),              !!(000300) INTO D ** NOTE: BIT<1,0> = 00 FOR BA LOAD **
17856      P3, DATO,                      !WRITE IT OUT
17857      NEXT, GOTO-PAGE(4),              !XFER FOR DCS XTN BITS
17858      J/ZAPD762A                      !
(7210) DCS[0.00.0.0.0.0] BM[1010..11.10..00.00..000..100...0.1.0.1.0.0.0.10010...0..0000.0...11.100...011.001.101]
17859
17860      4315: 1(FREE)
17861      ZAPD762A:
17862      P3-T, D_ZERO, DIC]_ALU15,        !ZERO D, DIC] FOR LOOP; MUST DO AFTER P3-T
17863      NEXT, J/NEXTD762A              !ENTER AT 2ND WORD OF LOOP; LET D SETTLE
(4315) DCS[0.00.0.0.0.0] BM[0011..00.00..00.00..000..100...1.1.0.0.0.0.0.0000...0..0000.0...11.000...100.110.111]
17864
17865      !*** THE FOLLOWING TWO WORDS NOW GO INTO A COUNT LOOP TO WAIT FOR THE LINE CLOCK TO INTERRUPT
17866      !*** THE WAIT WILL BE A MAXIMUM DELAY, DEPENDING UPON THE PROCESSOR UP3-UP3 CYCLE TIME:
17867      !***
17868      !***
17869      !***
17870      !***
17871      !***
17872      !***
17873      !***
17874      !***
17875      !***
17876      !***
17877      !***
17878      !*** NOTE THAT THE ABOVE LOOP TIME IS A MAXIMUM VALUE; WE WILL EXIT EARLY IF THE INTERRUPT COMES
17879      !*** THROUGH BEFORE WE OVERFLOW THE COUNTER. IF NO INTERRUPT COMES THROUGH BY THE TIME THE COUNT
17880      !*** HAS OVERFLOWED, IT WILL BE CONSIDERED AN ERROR.
17881      !***
17882
17883      !* ENTER HERE FOR ANOTHER LOOP *

```

```
17884 4551:
17885 BUMPD762A:
17886 P3-T, D_D-PLUS-1, DIC)_COUNT, |BUMP D, SAVE CARRYOUT
17887 NEXT, BUTR(BG-SERVICE-L), |NEGATED: "NEXTD762A"
17888 J/SETPR6-762A |ASSERTED: "SETPR6-762A"
(4551) DCS(0.00,0.0,0.0) BM(1001..01,11..01,01..0000..110...1.0..0..0..0000...0..0000,0...01,100...100,110,011)
17889
17890 4467:
17891 NEXTD762A:
17892 P0, DCS-CTR(C15.), |STALL; NOTE: NO BUMP-VERIFIES IN THIS LOOP
17893 NEXT, BUTR(D(C)-B), |SET: "TEST762A1" D OVERFLOWN, ERROR
17894 J/BUMPD762A |CLEAR: "BUMPD762A" GO FOR NEXT LOOP
(4467) DCS(0.00,1.0,0.0) BM(0000..00,00..00,00..0000..000...0.0,0..0..0..0000...0..0000,0...10,011...101,101,001)
17895
17896
17897 1- COME HERE IF D OVERFLOWN *
17898 4553:
17899 TEST762A1:
17900 P0, LOAD-ERROR(TEST762A1), |ERROR DIRECTORY KEY
17901 DCS-CTR(C0.), |SIGNAL ERROR NOW
17902 NEXT, J/TEST762A |FORCE A SCOPE LOOP
(4553) DCS(1.00,1.0,0.0) BM(1111..00,00..00,00..0000..000...0.0,0..0..0..0000...0..0000,0...11,000...111,111,011)
17903
17904
17905 1- COME HERE IF EXIT LOOP OK: D NOT OVERFLOWN, INTERRUPT PENDING ***
17906 4463:
17907 SETPR6-762A:
17908 P2-T, D_CSPD(D15), D(C)_0, |GET D<7:5>=(6), D<4>=(0)
17909 NEXT, J/SETPRI762A |
(4463) DCS(0.00,0.0,0.0) BM(1010..10,00..00,00..0000..000...0.1,0..0..0..0010...0..0000,0...11,000...100,001,011)
17910
17911 4413: 1(FREE)
17912 SETPRI762A:
17913 P0, BUSDIN_EMIT-[1], |KEEP IT ON
17914 P2, PS[3-0]_D[3-0]=[1], |FOR USE IN TEST763A, PS[CC]="0000" HERE
17915 P3, PS[7-4]_D[7-4]=[1], |PRIO=6; T-BIT=0
17916 NEXT, J/TEST762B |NOTE: THE BR6 PENDING INTR SHOULD NOW HIDE UNDER PROCESSOR PRIO(6)
(4413) DCS(0.00,0.0,0.0) BM(1000..00,00..01..010..000...0.0,0..0..0..1.011...0..0000,0...11,000...111,101,001)
17917
17918
17919
17920
17921 1-TEST 762B CHECKS THAT BG-SERVICE(0)H=BR>PS-L IS HIGH WHEN THE PROCESSOR PRIORITY(=6) IS AS HIGH
17922 1-AS THE ONLY DEVICE WISHING TO REQUEST AN INTERRUPT (AT BR6).
17923 4751:
17924 TEST762B:
17925 P0, LOAD-ENVA(ZTARGET407), |BIT<02> SET (ACTIVE LOW)
17926 LOAD-ERROR(TEST762B), |ERROR DIRECTORY KEY
17927 DCS-CTR(C3.), |COMPARE AT TARGET
17928 BUMP-VERIFY, |
17929 NEXT, J/GOBUT762B |
(4751) DCS(1.00,1.0,0.0) BM(1100..00,11..11,00..0000..111...0.0,0..0..0..0000...0..0000,0...11,000...111,110,010)
17930
17931 4762:
```

```
17932 GORUNT762B:
17933 SETUP, RETURN/TEST762C, |RETURN TO START OF NEXT SUBTEST
17934 NEXT, GOTO-PAGE(7), |BUT TABLE
17935 J/BUTBGSEVRL |BG-SERVICE(0)H IN BIT<02>
(4762) DCS(0.00,0.0,0.0) BM(0100..00,11..11,00..0000..111...0.0,0..0..0..0000...0..0000,0...11,100...011,001,110)
17936
17937
17938
17939
17940
17941
17942
17943 1-TEST 762C CHECKS THAT BG-SERVICE(0)H=BR>PS-L IS LOW WHEN A BR6 DEVICE (DL11-W) IS REQUESTING AN INTR,
17944 1-AND THE PPROCESSOR PRIORITY(=5) IS < THE BR LEVEL(=6), NO OTHER DEVICES PRESENT.
17945 4740:
17946 TEST762C:
17947 P0, LOAD-ENVA(ZTARGET403), |BIT<02> CLEAR (ACTIVE LOW)
17948 LOAD-ERROR(TEST762C), |ERROR DIRECTORY KEY
17949 DCS-CTR(C7.), |COMPARE AT TARGET
17950 NEXT, J/FILL762C |
(4740) DCS(1.00,1.0,0.0) BM(1000..00,11..11,00..0000..011...0.0,0..0..0..0000...0..0000,0...11,000...100,001,100)
17951
17952 4414: 1(FREE)
17953 FILL762C:
17954 P2-T, D_ASPhi(C125252), D(C)_ALU00, |BIT<7:5>="101"=(5), BIT<4>=0
17955 NEXT, J/DELAY762C |
(4414) DCS(0.00,0.0,0.0) BM(1111..00,00..11,01..110..010...0.1,0..0..0..0000...0..0000,0...11,000...100,001,101)
17956
17957 4415: 1(FREE)
17958 DELAY762C:
17959 SETUP, RETURN/SETPRI762C, |EXEC 3, UWORDS AFTER SETTING PRIO, FOR DELAY
17960 P2, PS[3-0]_D[3-0]=0, |FOR USE IN TEST763A, PS[CC]="1010"
17961 P3, PS[7-4]_D[7-4]=0, |SET PRIO=(5), AT P3-T OF THIS UWORD
17962 NEXT, GOTO-PAGE(7), |GO DO A JUMP, AND A BUTA(RETURN)
17963 J/BUTD-18-ZERO |DON'T REALLY CARE ABOUT THE RESULT OF THIS
(4415) DCS(0.00,0.0,0.0) BM(0111..00,01..00,00..110..111...0.0,0..0..0..1.010...0..0000,0...11,100...011,100,001)
17964
17965 7206: 1(FREE)
17966 SETPRI762C:
17967
17968 SETUP, RETURN/TEST762D, |BG-SERVICE(0)H SHOULD BE ASSERTED BY NOW
17969 NEXT, GOTO-PAGE(7), |RETURN TO START OF NEXT SUBTEST
17970 J/BUTBGSEVRL |BUT TABLE
17971 |BG-SERVICE(0)H IN BIT<02>
(7206) DCS(0.00,0.0,0.0) BM(0100..00,11..11,00..001..111...0.0,0..0..0..0000...0..0000,0...11,100...011,001,110)
17972
17973
17974
17975
17976
17977
17978 1-TEST 762D CHECKS THAT BG-SERVICE(1)H+FLTPPT-SERVICE-H IS HIGH, WHEN BG-SERVICE(1)H IS SET
```

```
17979 4741:
17980 TEST762D:
17981 PO, LOAD-ENUA(ZTARGET417), ;BIT<03> SET
17982 LOAD-ERROR(TEST762D), ;ERROR DIRECTORY KEY
17983 DCS-CTR(C3.), ;COMPARE AT TARGET
17984 NEXT, J/GOBUT762D
17985 (4741) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..001..111..0.0.0..0.0..0.0000...0.0000.0...11.000...100.001.110]
17986 4416: 1(FREE)
17987 GOBUT762D:
17988 SETUP, RETURN/TEST762E, ;RETURN TO START OF NEXT SUBTEST
17989 NEXT, GOTO-PAGE(7), ;BUT TABLE
17990 J/BUTSGPPSERV ;BG-SERVICE(1)H=PLTPT-SERVICE-H IN BIT<03>
17991 (4416) DCS[0.00.0.0.0.0] BM[0100..00.11..10.11..000..111..0.0.0..0.0..0.0000...0.0000.0...11.100...011.000.111]
17992
17993
17994
17995
17996 | - - - - -
17997
17998 |TEST 762E NOW READS THE "SERVICE" PORT OF THE STATUS MUX TO SEE:
17999 |SERVICE<15:00>H = "0 100 011 111 100 000"
18000
18001 | IMPORTANT BITS ARE:
18002 | B15 = DATI(1)H = 0 B14 = BG-SERVICE(1)H = 1 B11 = DATOB(1)H = 0
18003 | B10 = DATO(1)H = 1 B09 = PBA<17>H = 1 B08 = PBA<16>H = 1
18004
18005 4730:
18006 TEST762E:
18007 PO, LOAD-ENUA(ZTARGET402), ;SETUP FOR D=ZERO TEST
18008 LOAD-ERROR(TEST762E), ;ERROR DIRECTORY KEY
18009 DCS-CTR(C10.), ;COMPARE AT TARGET
18010 NEXT, J/EXPECT762E
18011 (4730) DCS[1.00.1.0.0.0] BM[0101..00.11..11.00..000..010..0.0.0..0.0..0.0000...0.0000.0...11.000...100.001.111]
18012 4417: 1(FREF)
18013 EXPECT762E:
18014 P3, CSPD[02]_EMIT, ;EXPECTED VALUE OUT OF SERVICE PORT:
18015 NEXT, EMIT/043740, ;"0100 0111 1110 0000"
18016 J/GOGET762E
18017 (4417) DCS[0.00.0.0.0.0] BM[0100..10.01..11.11..100..000...0.0.0..0.0..0.1101...1.0000.0...11.000...100.010.000]
18018 4420: 1(FREE)
18019 GOGET762E:
18020 PO, BUMP-VERIFY, ;
18021 SETUP, RETURN/TEST762F, ;GO TO SUBN WHICH:
18022 NEXT, CALL[SERVICEOD] ;CSP(02),XOR,SERVICE -> D, PUT(D=ZERO)
18023 (4420) DCS[0.00.0.0.0.1] BM[0100..00.11..10.01..001..111..0.0.0..0.0..0.0000...0.0000.0...11.100...010.101.000]
18024
18025
```

```
18026 | - - - - -
18027
18028 |TEST 762F CHECKS THAT SERVICE-H=NOT(INTR-HIGH-H=FLAG7(0)H=BG-SERVICE(0)H) IS HIGH,
18029 | WHEN INTR-HIGH-H=HIGH, FLAG7(0)H=HIGH, AND BG-SERVICE(0)H=LOW
18030 4711:
18031 TEST762F:
18032 PO, LOAD-ENUA(ZTARGET403), ;BIT<00> SET
18033 LOAD-ERROR(TEST762F), ;ERROR DIRECTORY KEY
18034 DCS-CTR(C3.), ;COMPARE AT TARGET
18035 NEXT, J/GOBUT762F
18036 (4711) DCS[1.00.1.0.0.0] BM[1100..00.11..11.00..000..011..0.0.0..0.0..0.0000...0.0000.0...11.000...100.010.001]
18037 4421: 1(FREF)
18038 GOBUT762F:
18039 SETUP, RETURN/TEST763A, ;RETURN TO START OF NEXT SUBTEST
18040 NEXT, GOTO-PAGE(7), ;BUT TABLE
18041 J/BUTSERVICE ;SERVICE-H IN BIT<00>
18042 (4421) DCS[0.00.0.0.0.0] BM[0100..00.11..11.10..001..111..0.0.0..0.0..0.0000...0.0000.0...11.100...011.100.101]
18043
18044
18045
18046
18047 | - - - - -
18048
18049 |THIS NEXT SEQUENCE OF TWO TESTS RESPONDS TO THE BUS INTERRUPT REQUEST BY:
18050 | (1) ASSERTING "ALLOW-BG(1)H", THUS ALLOWING THE BUS GRANT TO THE DL11-W, SO THAT
18051 | (2) "VECTOR-LOAD(1)H" WILL BE ASSERTED, INDICATING THAT THE DEVICE HAS PUT ITS
18052 | VECTOR ON UNIBUS DATA<8:0>L, AND THEN
18053 | (3) ACTUALLY READING THE VECTOR FOR THE DL11-W (100)(R), AND VALIDATING ITS CORRECTNESS.
18054
18055
18056
18057 | - - - - -
18058
18059 |TEST 763A CHECKS THAT AFTER "ALLOW-BG(1)H" IS GIVEN TO THE INTERRUPTING DEVICE,
18060 | THEN "VECTOR-LOAD(1)H" IS ASSERTED
18061 4761:
18062 TEST763A:
18063 PO, LOAD-ENUA(VECTLOAD763A), ;COMPARE AT "VECTOR-LOAD(1)H" IN BIT<01> SET
18064 LOAD-ERROR(TEST763A), ;ERROR DIRECTORY KEY
18065 DCS-CTR(C5.), ;COMPARE AT TARGET
18066 NEXT, J/EXPECT763A
18067 (4761) DCS[1.00.1.0.0.0] BM[1010..00.10..01.11..011..011..0.0.0..0.0..0.0000...0.0000.0...11.000...100.010.010]
18068 4422: 1(FREF)
18069 EXPECT763A:
18070 PO, BUMP-VERIFY, ;COUNT
18071 P3, CSPD[01]_EMIT, EMIT/100, ;DL11-W VECTOR IS (100)R
18072 NEXT, J/ALLOW763A
18073 (4422) DCS[0.00.0.0.0.1] BM[0000..10.00..00.01..000..000...0.0.0..0.0..0.1110...1.0000.0...11.000...111.011.001]
18074 4731:
18075 ;*** *** *01
```

```
18075 ALLOW763A:
18076 PO, BUSDIN_PS-[I],
18077
18078 P2, ALLOW-BG[I]H-[I],
18079 NEXT, J/READVECT763A
(4731) DCS[0.00.0.0.0.0] BM[0100.00.01.10.01.000.000.0.0.0.0.0.0.1.011.0.0.000.0.11.000.100.010.011]
18080
18081 4423: 1(FREE)
18082 READVECT763A:
18083 P3, CSPD[07]_BUSDIN,
18084
18085 EMIT/052525,
18086 NEXT, J/TESTVECT763A
(4423) DCS[0.00.0.0.0.0] BM[0101.10.01.01.01.010.101.0.0.0.0.0.0.0.1000.1.0000.0.11.000.100.010.100]
18087
18088 4424: 1(FREE)
18089 TESTVECT763A:
18090 P2-T, SR_CSPD[D01],
18091 NEXT, BUTR(VECTOR-LOAD),
18092 J/ALLOW763A
(4424) DCS[0.00.0.0.0.0] BM[1010.10.00.00.00.000.000.0.0.1.0.0.0.0.1110.0.0000.0.10.001.111.011.001]
18093
18094
18095
18096
18097
18098
18099
18100
18101
18102
18103
18104
18105
18106
18107
18108
18109
18110
18111
18112
18113
18114
18115
18116
18117
18118
18119
18120
18121
```

```
18122
18123
18124
18125
18126
18127
18128
18129
18130
18131
18132
18133
18134
18135
18136
18137
18138
18139
18140
18141
18142
18143
18144
18145
18146
18147
18148
18149
18150
18151
18152
18153
18154
18155
18156
18157
18158
18159
18160
18161
18162
18163
18164
18165
18166
18167
18168
```

```

18169
18170 |-----|
18171
18172
18173 |*** ALL DONE WITH LINE CLOCK, DISABLE IT FROM FURTHER INTERRUPTS BEFORE LEAVING FOR EOP ***|
18174
18175 4763:
18176 WILL764A:
18177 P2=T, D_ZERO, DIC)ALU00, INON MUST GO KILL THE LINE CLOCK
18178 P3, DATA, ISAP(177946), DL11=N CSR
18179 NEXT, BUTA(CLR-FLAG-RES-UCON), IZAP UCON, RES-CONTROL FOR EOP ROUTINE (MUST BE DONE)
18180 J/EOP001
(4763) DCS(0.00,0.0,0.0) BM(0011..00.00..00.00..000..010..0.1.0..0..0..1.0010...0..0000,0...11.010...100,011,001]
18181
18182
18183
18184
18185
18186
18187
18188 1.PAGE=====
18189
18190
18191 .TOC * END OF PASS CODE
18192
18193 |=====|
18194 |
18195 | FINAL ENTRY HERE TO TEST FOR MULTIPLE PASSES, VERIFY MODE, ET AL
18196 |
18197 |
18198 | *** INDICATIONS ON CONSOLE DISPLAY ***
18199 |
18200 | 0.0,0.0,0.0, DCS IS RUNNING; REQUIRES APPROX 8. SECONDS/64. PASSES
18201 | "RUN" "PROC" "USER" "CONSOLE" "BATTERY"
18202 | (ON) (BLINK) (BLINK) (OFF) (OFF)
18203 |
18204 | 0 0 0 0 0 0 ERROR DETECTED / PASS=1 / SCOPE LOOPING
18205 | 2 1 2 1 2 1
18206 |
18207 | 0.0,0.0,0.0, ERROR DETECTED / 1<PASS<64. / SCOPE LOOPING
18208 | 2.1,2.1,2.1.
18209 |
18210 | 1.2,3.3,2.1. SUCCESSFUL 64. PASSES COMPLETED
18211 |
18212 |
18213 | AT END OF PASS, THE GENERAL REGISTERS (BASE MACHINE R0-R7)
18214 | CONTAIN THE FOLLOWING INFORMATION:
18215 |
18216 | R0 = (NU)
18217 | R1 = (NU)
18218 | R2 = (NU)
18219 | R3 = (NU)
18220 | R4 = (NU)
18221 | R5 = REVISION NUMBER OF DCS MICROCODE, WITH

```

```

18222 | BIT15 SET TO INDICATE END OF PASS
18223 | R6 = (NU)
18224 | R7 = (123321), END OF PASS INDICATION CONSTANT
18225 |
18226 |=====|
18227
18228
18229 4431: 1(FREE)
18230 FOP001:
18231 P0, DCS-CTR(C15.), I HOLD UP ERROR COMPARE
18232 NEXT, GOTO-PAGE(6), I XFER
18233 J/EOP002
(4431) DCS(0.00,1.0,0.0) BM(0000..00.00..00.00..000..110..0.0.0..0..0..0.0000...0..0000,0...11.100...100,101,010]
18234
18235
18236 6442: 1(FREE)
18237 FOP002:
18238 P3, CSDP(01)EMIT, EMIT/123321, I SETUP SUCCESSFUL END OF PASS CONSTANT
18239 NEXT, J/EOP003
(6442) DCS(0.00,0.0,0.0) BM(1010..10.01..10.11..010..001...0.0.0..0..0..0.1110...1..0000,0...11.000...101,001,000]
18240
18241
18242 6510: 1(FREE)
18243 FOP003:
18244 P2=T, D_CSPD(D01), DIC)0, I GET EOP CONSTANT INTO
18245 P3, PC_D, I GPR PC, ON BOTH SP SIDES
18246 NEXT, J/EOP004 I FOR DISPLAY FROM "CONSOLE HALT" ROUTINE
(6510) DCS(0.00,0.0,0.0) BM(1010..10.00..10.01..111..000...0.1.0..0..0..0.1110...0..0011,0...11.000...101,001,001]
18247
18248
18249 6511: 1(FREE)
18250 FOP004:
18251 SETUP, RETURN/EOP005, I GO TO DISPLAY-D-IN-LIGHTS ROUTINE
18252 NEXT, CALL[DISPLAY] I WHICH SHOULD PUT (212121) INTO DISPLAY
(6511) DCS(0.00,0.0,0.0) BM(0111..00.01..00.01..001..111...0.0.0..0..0..0.0000...0..0000,0...11.100...010,010,001]
18253
18254
18255 7211: 1(FREE)
18256 FOP005:
18257 P0, SIGNAL-EOP, I TRY TO SIGNAL EOP
18258 P2=T, SR_ALL-ONES, I SET SR TO ALL ONES, FOR FULL BASE MACHINE INIT
18259 NEXT, GOTO-PAGE(6), I XFER TO 6
18260 RUD(VERIFY-MODE), I IF TRUE, "VFY001" (NEXT PAGE FOLLOWING)
18261 J/EOP006 I IF FALSE, "EOP006" (NEXT MICROWORD)
(7211) DCS(0.00,0.0,1.0) BM(1111..00.00..11.01..101..110...0.0.1..0..0..0.0000...0..0000,0...11.100...111,111,110]
18262
18263
18264 6776:
18265 FOP006:
18266 P0, DCS-CTR(C11.), I HOLD UP ERROR COMPARE
18267 P3=0, SET-CONSOLE-DP-LEDS, I UCON SET DP LEDES, INDICATING >=1 SUCCESSFUL PASSES

```

```

18268 NEXT, BUTD(EOP-OVERFLOW), IIF TRUE, "EOP007" FOLLOWING, DONE 64, PASSES
18269 J/EOP007 IIF FALSE, "TEST001", GO FOR NEXT PASS AT (4000)
(6776) DCS[0.00.1.0.0.0] BM[0100.00.00.00.00.110.001...0.0.0.0.0...1.1011...0.0000.0...11.000...101.001.010]
18270 I BEGIN COUNTDOWN
18271 I 3 ... 2 ... 1 .
18272
18273 6512: 1(FREF)
18274 EOP007:
18275 SETUP, RETURN/EOP010, I GO TO SUBR THAT PUTS REV-NUMBER, WITH
18276 NEXT, GOTO-PAGE(7), I B<15>=(1), INTO B.M. GPR "R5"
18277 J/INSERTOPREVNO I
18278 (6512) DCS[0.00.0.0.0.0] BM[0111.00.01.00.01.010.111...0.0.0.0.0...0.0000...0.0000.0...11.100...010.001.100]
18279
18280 7212: 1(FREF)
18281 EOP010:
18282 SETUP, RETURN/CON99, I RETURN TO "FORCE CONSOLE-MODE WAIT" ROUTINE IN BASE MACHINE
18283 NEXT, GOTO-PAGE(3), I GOTO "INIT---" ROUTINE FOR FULL BASE MACHINE
18284 J/INIT01 I MICROCODE INITIALIZATION, BUTA(RETURN) AT END TO "CON99"
18285 (7212) DCS[0.00.0.0.0.0] BM[0001.00.00.01.00.000.011...0.0.0.0.0...0.0000...0.0000.0...11.100...100.001.010]
18286
18287
18288
18289
18290
18291 1.PAGE=====
18292
18293 .TOC * VERIFY MODE CODE
18294
18295 I
18296 I VERIFY MODE ENTERS HERE;
18297 I
18298 I
18299
18300 6774:
18301 VFY001:
18302 PO, DCS-CTR(C4.), I LOAD COUNTER FOR COMPARE IN 4. MICROWORDS
18303 NEXT, GOTO-PAGE(4), I ZFER
18304 J/VFY002 I
(6774) DCS[0.00.1.0.0.0] BM[1011.00.00.00.00.000.100...0.0.0.0.0...0.0000...0.0000.0...11.100...100.001.010]
18305
18306 4412: 1(FREF)
18307 VFY002:
18308 P3, CLEAR-CONSOLE-LED, I MAKE "CONSOLE" LED BLINK, JUST FOR FUN
18309 NEXT, J/VFY003 I
18310 (4412) DCS[0.00.0.0.0.0] BM[0100.00.00.00.00.010.001...0.0.0.0.0...1.1011...0.0000.0...11.000...010.101.101]
18311
18312 4255:

```

```

18313 VFY003:
18314 PO, LOAD-ENUA(ZTARGET523), I ERROR CODE = [4255], ENUA = [7523]
18315 LOAD-ERROR(VFY003), I ERROR DIRECTORY KEY
18316 BUMP-VERIFY, I GIVE A VERIFY PULSE
18317 NEXT, J/VFY004 I
(4255) DCS[1.00.0.0.0.1] BM[0000.00.11.11.01.010.011...0.0.0.0.0...0.0000...0.0000.0...11.000...100.011.011]
18318
18319 4433: 1(FREF)
18320 VFY004:
18321 SETUP, RETURN/VFY005, I RETURN TO INLINE
18322 NEXT, GOTO-PAGE(7), I BUT'S ARE ON PAGE 7
18323 J/ZTARGET522 I TWUA = [7522], NOT EQUAL TO ENUA
18324 (4433) DCS[0.00.0.0.0.0] BM[0111.00.01.11.11.101.111...0.0.0.0.0...0.0000...0.0000.0...11.100...101.010.010]
18325
18326 I
18327 I NEXT MICROWORD COMES FROM ZTARGET522, AT WHICH THE ENABLED COMPARE
18328 I TAKES PLACE, ENUA WAS SETUP NOT EQUAL TO TWUA, SO ERROR SHOULD BE SIGNALLED
18329
18330
18331 7375:
18332 VFY005:
18333 P3, SET-CONSOLE-LED, I THE OTHER HALF OF MAKING IT BLINK
18334 NEXT, BUTD(SCOPE), I NO ERROR; "VFY005" [SELF LOOP, SHOULDN'T HAPPEN]
18335 J/VFY005 I ERROR: "VFY006" [SHOULD HAPPEN]
18336 (7375) DCS[0.00.0.1.0.0] BM[0100.00.00.00.00.100.001...0.0.0.0.0...1.1011...0.0000.0...11.000...011.111.101]
18337
18338 7374:
18339 VFY006:
18340 PO, SIGNAL-EOP, I GIVE AN EOP PULSE, AFTER ERROR SIGNALLED
18341 NEXT, GOTO-PAGE(6), I LOOP BACK
18342 J/VFY001 I ON CONTINUOUS VERIFY
18343 (7374) DCS[0.00.0.0.1.0] BM[0000.00.00.00.00.000.110...0.0.0.0.0...0.0000...0.0000.0...11.100...111.111.100]
18344
18345
18346
18347
18348
18349 1.PAGE=====
18350
18351 .TOC * DCS MICROCODE REVISION NUMBER
18352
18353 I
18354 I THE FOLLOWING ROUTINE WILL PUT THE CURRENT DCS MICROCODE REVISION NUMBER
18355 I INTO B.M. GPR "R5", FROM THE ENIT FIELD OF THE MICROWORD.
18356 I
18357 I THE ENTRY "INSERTREVNO" IS USED, EXCEPT AT END OF PASS; IE B<15>=(0)
18358 I THE ENTRY "INSERTOPREVNO" IS USED ONLY AT END OF PASS; IE, B<15>=(1)
18359
18360

```

```
18361 7214: 1(FREE)
18362 INSERT02:
18363 P0, BUSDIN_EMIT-[I], ISELECT EMIT
18364 NEXT, J/INSERT02
(7214) DCS[0.00.0.0.0.0] BM[0000..00.00..00.01..000..000...0.0.0.0.0...1.1001...0..0000.0...11.000...010.001.101]
18365 7215: 1(FREE)
18366 INSERT03:
18367 P3, CSPD[17]_EMIT, EMIT/REV-NUMBER-AND-B15, DCS REVISION NUMBER, B15 SET
18368 NEXT, J/INSERT04
18369 (7215) DCS[0.00.0.0.0.0] BM[1000..10.00..00.01..000..001...0.0.0.0.0...0.0000...1..0000.0...11.000...010.010.000]
18370
18371 7216: 1(FREE)
18372 INSERT03:
18373 P0, BUSDIN_EMIT-[I], ISELECT EMIT
18374 NEXT, J/INSERT03
18375 (7216) DCS[0.00.0.0.0.0] BM[0000..00.00..00.01..000..000...0.0.0.0.0...1.1001...0..0000.0...11.000...010.001.111]
18376
18377 7217: 1(FREE)
18378 INSERT03:
18379 P3, CSPD[17]_EMIT, EMIT/REV-NUMBER, DCS REVISION NUMBER, B15 CLEAR
18380 NEXT, J/INSERT04
(7217) DCS[0.00.0.0.0.0] BM[0000..10.00..00.01..000..001...0.0.0.0.0...0.0000...1..0000.0...11.000...010.010.000]
18381
18382 7220: 1(FREE)
18383 INSERT04:
18384 P2-T, D_CSPD[D17], D[C]_0, IGET IT
18385 P3, RS_0, AND STUFF IT
18386 NEXT, J/RESETUOMP AND RETURN
18387 (7220) DCS[0.00.0.0.0.0] BM[1010..10.00..10.01..110..000...0.1.0.0.0...0.0000...0..0011.0...11.000...010.111.001]
18388
18389
18390
18391 1.PAGE=====
18392
18393 .TOC * COMMON SUBROUTINES
18394
18395 .TOC * CONSOLE DISPLAY SUBROUTINE
18396
18397
18398
18399
18400
18401
18402
18403 7221: 1(FREE)
18404 DISPLAY:
18405 P3-U, CLEAR-CONSOLE-COUNTER, IPOINT TO DIGITS ...XX
18406 NEXT, J/DISP002
(7221) DCS[0.00.0.0.0.0] BM[0100..00.00..00.00..010..000...0.0.0.0.0...1.1011...0..0000.0...11.000...010.010.010]
18407
18408
```

```
18409 7222: 1(FREE)
18410 DISP002:
18411 P3-U, STROBE-CONSOLE-DISPLAY, IWRITE OUT DIGITS ...XX
18412 NEXT, J/DISP003
(7222) DCS[0.00.0.0.0.0] BM[0100..00.00..00.00..000..001...0.0.0.0.0...1.1011...0..0000.0...11.000...010.010.011]
18413
18414 7223: 1(FREE)
18415 DISP003:
18416 P3-U, INCREMENT-CONSOLE-COUNTER, IPOINT TO DIGITS ...XX..
18417 NEXT, J/DISP004
(7223) DCS[0.00.0.0.0.0] BM[0100..00.00..00.00..100..000...0.0.0.0.0...1.1011...0..0000.0...11.000...010.010.100]
18418
18419
18420 7224: 1(FREE)
18421 DISP004:
18422 P3-U, STROBE-CONSOLE-DISPLAY, IWRITE OUT DIGITS ...XX..
18423 NEXT, J/DISP005
(7224) DCS[0.00.0.0.0.0] BM[0100..00.00..00.00..000..001...0.0.0.0.0...1.1011...0..0000.0...11.000...010.010.101]
18424
18425 7225: 1(FREE)
18426 DISP005:
18427 P3-U, INCREMENT-CONSOLE-COUNTER, IPOINT TO DIGITS XX...
18428 NEXT, J/DISP006
(7225) DCS[0.00.0.0.0.0] BM[0100..00.00..00.00..100..000...0.0.0.0.0...1.1011...0..0000.0...11.000...010.010.110]
18429
18430
18431 7226: 1(FREE)
18432 DISP006:
18433 P3-U, STROBE-CONSOLE-DISPLAY, IWRITE OUT DIGITS XX...
18434 NEXT, J/RESETUOMP IGO RESET PROC UCON/EMIT, DO A (RETURN)
18435 (7226) DCS[0.00.0.0.0.0] BM[0100..00.00..00.00..000..001...0.0.0.0.0...1.1011...0..0000.0...11.000...010.111.001]
18436
18437
18438
18439
18440
18441
18442 1.PAGE=====
18443
18444 .TOC * CLEAR I-O / BUS CONTROL / SERVICE AREA STATUS LATCHES SUBR
18445
18446
18447
18448
18449
18450
18451
18452
18453 7227: 1(FREE)
18454 CLEAR-I-O-A1
18455 P3, BUS-INIT-UCON-[I], IDO A 10 MILLISEC UNIBUS INIT
18456 NEXT, J/CLEAR-T-O-B
(7227) DCS[0.00.0.0.0.0] BM[0100..00.01..11.00..000..000...0.0.0.0.0...1.1011...0..0000.0...11.000...010.011.000]
18457
```

```

18457 7230: 1(FREE)
18458 CLEAR-I-O-E:
18459 P3, CLR-JAM-ERRORS-[I], |RESET CACHE ERROR STATUS
18460 NEXT, J/CLEARI002
18461 (7230) DCS[0.00,0.0,0.0] BM[0100..00.00..10.00..000..000...0.0,0.0,0.0...1.1011...0..0000,0...11.000...010.011.001]
18462 7231: 1(FREE)
18463 CLEARI002:
18464 P3, CLEAR-CONSOLE-SERVICE, |CLEAR OUT CONSOLE SRVC(1)H
18465 P3, CLR-NPR-TIMEOUT-[I], |RESET NPR/BACK TIME OUT STATUS
18466 NEXT, J/CLEARI004
18467 (7231) DCS[0.00,0.0,0.0] BM[0100..00.00..11.00..110..000...0.0,0.0,0.0...1.1011...0..0000,0...11.000...010.011.010]
18468 7232: 1(FREE)
18469 CLEARI004:
18470 P3, CLEAR-CONSOLE-LED, |SETUP FOR 14. BIT ADDRESS MODE ON UNIBUS
18471 P3, CLR-PWR-FAIL-[I], |RESET POWER FAIL STATUS
18472 NEXT, J/CLEARI005
18473 (7232) DCS[0.00,0.0,0.0] BM[0100..00.01..00.00..010..001...0.0,0.0,0.0...1.1011...0..0000,0...11.000...010.011.011]
18474 7233: 1(FREE)
18475 CLEARI005:
18476 P3, CLR-YELLOW-ZONE-[I], |RESET YELLOW ZONE STATUS
18477 NEXT, J/RESETUONP
18478 (7233) DCS[0.00,0.0,0.0] BM[0100..00.01..01.00..000..000...0.0,0.0,0.0...1.1011...0..0000,0...11.000...010.111.001]
18479 |"RESETUONP" IS AT END OF "DINTOIR" SUBROUTINE
18480 |THIS WORD (1) ENABLES BUSDIN_EMIT, (2) ENABLES PROC EN-CLK-IR,
18481 |AND EXITS WITH A BUTA(RETURN)
18482
18483
18484
18485
18486
18487
18488
18489
18490
18491 .TNC * SUBR FOR PUTTING SELECTED PORTIONS OF D[15-00] INTO IN
18492
18493
18494
18495
18496
18497
18498
18499
18500
18501
18502
18503
18504
18505
18506
18507
18508
18509
18510
18511
18512
18513
18514
18515
18516
18517
18518
18519
18520
18521
18522
18523
18524
18525
18526
18527
18528
18529
18530
18531
18532
18533
18534
18535
18536
18537
18538
18539
18540
18541
18542
18543
18544
18545
18546
18547
18548
18549
18550
18551
18552
18553
18554

```

```

18507
18508
18509
18510 7234: 1(FREE)
18511 D[15-12]:
18512 P2-U, IR_DBUF, |JUST HAPPENS, DONT CARE NOW
18513 P3, DBUF_D, |PATTERN IN D -> DBUF
18514 A#BSPHI[17]_D, |SAVE IN ASPHI, TOO
18515 NEXT, J/D1512A
18516 (7234) DCS[0.00,0.0,0.0] BM[0000..00.11..00.01..011..000...0.0,0.0,0.0...1.1010...0..1011,0...11.000...010.011.101]
18517 7235: 1(FREE)
18518 D1512A:
18519 P2-U, IR_DBUF, |SEND DBUF -> IR
18520 P3, DBUF_D, |JUST HAPPENS, DONT CARE NOW
18521 NEXT, J/BUTIR15-12
18522 (7235) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...1.1010...0..0000,0...11.000...011.000.000]
18523 7236: 1(FREE)
18524 D[11-06]:
18525 P2-T, SR_ASPHI[17]-AND-007700, |MASK ALL BITS EXCEPT D<11:06> TO ZEROES
18526 NEXT, J/D1106A
18527 (7236) DCS[0.00,0.0,0.0] BM[1011..11.00..11.01..011..000...0.0,1.0,0.0...0.0000...0..0000,0...11.000...010.011.111]
18528 7237: 1(FREE)
18529 D1106A:
18530 P2-T, D_SR-IOR-170000, SAVE-D[C], |FORCE BITS D<15:12> TO ONES
18531 |THIS MANUEVER SHOULD FORCE INSTR5 DIAGNOSTIC (E78)
18532 |FROM ADDRESSES (732), (725) WHEN PATTERN GOES INTO IR
18533 P2-U, IR_DBUF, |JUST HAPPENS, DONT CARE NOW
18534 P3, DBUF_D, |PUT PATTERN JUST SET IN D INTO DBUF
18535 NEXT, J/DTOIRB
18536 (7237) DCS[0.00,0.0,0.0] BM[1110..11.01..00.00..000..111...0.1,0.0,0.0...1.1010...0..0000,0...11.000...010.100.011]
18537 7240: 1(FREE)
18538 D[05-00]:
18539 P2-T, SR_ASPHI[17]-AND-000077, |MASK ALL BITS EXCEPT D<05:00> TO ZEROES
18540 NEXT, J/D0500A
18541 (7240) DCS[0.00,0.0,0.0] BM[1011..11.10..11.01..011..000...0.0,1.0,0.0...0.0000...0..0000,0...11.000...010.100.001]
18542 7241: 1(FREE)
18543 D0500A:
18544 P2-T, D_SR-IOR-000100, SAVE-D[C], |FORCE BIT D<06> TO A ONE
18545 |THIS MANUEVER SHOULD FORCE INSTR5 DIAGNOSTIC (E88)
18546 |FROM ADDRESSES (152), (125) WHEN PATTERN GOES INTO IR
18547 P2-U, IR_DBUF, |JUST HAPPENS, DONT CARE NOW
18548 P3, DBUF_D, |PUT PATTERN JUST SET IN D INTO DBUF
18549 NEXT, J/DTOIRB
18550 (7241) DCS[0.00,0.0,0.0] BM[1110..11.11..00.00..000..111...0.1,0.0,0.0...1.1010...0..0000,0...11.000...010.100.011]
18551 7242: 1(FREE)
18552 DZERO:
18553 P2-U, IR_DBUF-[I], |JUST HAPPENS, DONT CARE
18554 P3, DBUF_D-[I], |COPY PATTERN IN D TO DBUF

```

```
18555 NEXT, J/DTOIRB
(7242) DCS[0.00,0.0,0.0] BM[0100..00.00..00,01..000..100...0,0,0..0...1.1011...0..0000,0...11,000...010,100,011]
18556
18557 7243: 1(FREE)
18558 DTOIRB:
18559 P2=U, TP,DBUF, INOW COPY PATTERN IN DBUF TO IR
18560 P3, DBUF,D, IJUST HAPPENS, DONT CARE
18561 NEXT, J/BUSINSTRS IGO DO INSTRS "BUT" ON PATTERN IN IR
(7243) DCS[0.00,0.0,0.0] BM[0000..00.00..00,00,000..000...0,0,0..0...1.1010...0..0000,0...11,000...011,000,001]
18562
18563
18564
18565
18566
18567 1.PAGE=====
18568
18569
18570 .TOC * UCON SUBROUTINES (FLAGS, PS, FPS, CUA, SERVICE, JAN, PBA)
18571
18572
18573 THESE SUBROUTINES MANIPULATE THE PROCESSOR UCONS, AND VARIOUS OTHER BUSDIN DRIVERS
18574
18575 ASPLO(17) = A TEMPORARY LOCATION, DESTROYED
18576
18577 CSP(02) = VALUE EXPECTED TO BE READ
18578 CSP(03) = ACTUAL VALUE READ
18579 CSP(04) = MASK VALUE
18580
18581 AT RETURN, D = (BUSDINSELECT,AND,MASKVALUE),XOR,EXPECTEDVALUE
18582
18583
18584 7244: 1(FREE)
18585 CLRSERVICETOD:
18586 P3, CSPD[02]_EMIT, EMIT/000340, SERVICE PORT OF STATUS: ALL BITS RESET
18587 NEXT, J/CJESERVICETOD
(7244) DCS[0.00,0.0,0.0] BM[0000..10.00..00,11..100..000...0,0,0..0...0.1101...1..0000,0...11,000...010,100,111]
18588
18589 7245: 1(FREE)
18590 DATISERVICETOD:
18591 P3, CSPD[02]_EMIT, EMIT/100340, SERVICE PORT OF STATUS: ONLY DATA[1]H SET
18592 NEXT, J/CJESERVICETOD
(7245) DCS[0.00,0.0,0.0] BM[1000..10.00..00,11..100..000...0,0,0..0...0.1101...1..0000,0...11,000...010,100,111]
18593
18594 7246: 1(FREE)
18595 DATOSERVICETOD:
18596 P3, CSPD[02]_EMIT, EMIT/002340, SERVICE PORT OF STATUS: ONLY DATA[1]H SET
18597 NEXT, J/CJESERVICETOD
(7246) DCS[0.00,0.0,0.0] BM[0000..10.01..00,11..100..000...0,0,0..0...0.1101...1..0000,0...11,000...010,100,111]
18598
18599 7247: 1(FREE)
18600 CJESERVICETOD:
18601 P3, CLR-JAM-ERRORS-[1], SPECIAL ENTRY POINT TO CLEAR OUT JAM
18602 NEXT, J/CJESERVICETOD I PORT OF STATUS MUX, BEFORE READ SERVICE
(7247) DCS[0.00,0.0,0.0] BM[0100..00.00..10,00..000..000...0,0,0..0...1.1011...0..0000,0...11,000...010,101,000]
```

```
18603
18604 7250: 1(FREE)
18605 SERVICETOD:
18606 P0, BUSDIN_SERVICE-[1], SERVICE (PORT 1) OF STATUS MUX
18607 NEXT, J/GETPROC DAT
(7250) DCS[0.00,0.0,0.0] BM[0100..01.00..00,00..000..000...0,0,0..0...1.1001...0..0000,0...11,000...011,111,011]
18608
18609
18610
18611 7251: 1(FREE)
18612 PBATOD:
18613 P0, BUSDIN_PBA-[1], PBA (PORT 3) OF STATUS MUX
18614 NEXT, J/GETPROC DAT
(7251) DCS[0.00,0.0,0.0] BM[1100..01.00..00,00..000..000...0,0,0..0...1.1001...0..0000,0...11,000...011,111,011]
18615
18616
18617
18618 7252: 1(FREE)
18619 FLAGPPSTOD:
18620 P0, BUSDIN_FLAGS#FPS-[1], PUT FLAGS, FPS ON BUSDIN
18621 NEXT, J/GETPROC DAT
(7252) DCS[0.00,0.0,0.0] BM[0000..00.00..11,01..000..000...0,0,0..0...1.1001...0..0000,0...11,000...011,111,011]
18622
18623
18624
18625 7253: 1(FREE)
18626 PSTOD:
18627 P0, BUSDIN_PS-[1], PUT PS ON BUSDIN
18628 NEXT, J/GETPROC DAT
(7253) DCS[0.00,0.0,0.0] BM[0000..00.00..10,01..000..000...0,0,0..0...1.1001...0..0000,0...11,000...011,111,011]
18629
18630
18631
18632 7254: 1(FREE)
18633 ODDJAMTOD:
18634 P3, CSPD[02]_EMIT, EMIT/101004, JAM PORT OF STATUS: ONLY ODD-ADDR[1]H SET
18635 NEXT, J/JAMTOD
(7254) DCS[0.00,0.0,0.0] BM[1000..10.00..10,00..000..100...0,0,0..0...0.1101...1..0000,0...11,000...010,101,110]
18636
18637 7255: 1(FREE)
18638 CLRJAMTOD:
18639 P3, CSPD[02]_EMIT, EMIT/001000, JAM PORT OF STATUS: ALL BITS RESET
18640 NEXT, J/JAMTOD
(7255) DCS[0.00,0.0,0.0] BM[0000..10.00..10,00..000..000...0,0,0..0...0.1101...1..0000,0...11,000...010,101,110]
18641
18642 7256: 1(FREE)
18643 JAMTOD:
18644 P0, BUSDIN_JAM-[1], PUT JAM REG (STATUS MUX PORT 2)
18645 NEXT, J/GETPROC DAT IONTO BUSDIN
(7256) DCS[0.00,0.0,0.0] BM[1100..00.00..00,00..000..000...0,0,0..0...1.1001...0..0000,0...11,000...011,111,011]
18646
18647
18648
18649 7257: 1(FREE)
```

```

18650 CUATOD1
18651 PO, BUSDIN_CUA-[I], INPUT CUA REG (HBMUX PORT 2)
18652 NEXT, J/GETPROC DAT IONTO BUSDIN
(7257) DCS[0.00,0.0,0.0] BM[0000..00.00..01.01..000..000..0.0,0.0,0.0,1.1001...0..0000,0...11,000...011,111,011]

18653
18654
18655
18656 7373:
18657 GETPROC DAT:
18658 P3, CSPD[03]_BUSDIN, IGET PREVIOUSLY ENABLED PROC DATA
18659 NEXT, J/GETMSKPROC DAT
(7373) DCS[0.00,0.0,0.0] BM[0000..10.00..00.00..000..000..0.0,0.0,0.0,0.1100...1..0000,0...11,000...010,110,000]

18660
18661 7260: I(FREE)
18662 GETMSKPROC DAT:
18663 P2-T, D_CSPD[04], D[C]_0, IGET MASK VALUE
18664 P3, ASPLO[17]_D, IINTO A=00DE
18665 NEXT, J/MSKPROC DAT
(7260) DCS[0.00,0.0,0.0] BM[1010..10.00..00.01..011..000..0.0,0.0,0.0,0.1011...0..0001,0...11,000...010,110,001]

18666
18667 7261: I(FREE)
18668 MSKPROC DAT:
18669 P2-T, D_ASPLO[17]-AND-CSPD[03], SAVE-D[C], IMASK OUT UNWANTED BITS
18670 P3, ASPLO[17]_D, IAND WRITE BACK
18671 NEXT, J/CMPPPROC DAT
(7261) DCS[0.00,0.0,0.0] BM[1011..10.00..10.01..011..111...0.0,0.0,0.0,0.1100...0..0001,0...11,000...010,110,010]

18672
18673 7262: I(FREE)
18674 CMPPROC DAT:
18675 P2-T, D_ASPLO[17]-XOR-CSPD[03], SAVE-D[C], ICOMPARE OBTAINED, EXPECTED BITWISE
18676 NEXT, J/RESETPROC DAT
(7262) DCS[0.00,0.0,0.0] BM[1010..10.00..10.01..011..111...0.0,0.0,0.0,0.1101...0..0000,0...11,000...010,110,011]

18677
18678 7263: I(FREE)
18679 RESETPROC DAT:
18680 PO, BUSDIN_EMIT-[I], IRESET PROC UCON
18681 EN-CLK-IR[15-00], I
18682 NEXT, J/BUTD-IS-ZERO IAND GO TEST D<15:00>
(7263) DCS[0.00,0.0,0.0] BM[0000..00.00..00.01..000..100...0.0,0.0,0.0,1.1001...0..0000,0...11,000...011,100,001]

18693
18694
18695
18696
18697
18698
18699
18700
18701
18702
18703 7264: I(FREE)
18704 LOADFPSCC:
18705 P2-T, SP_ALL-ONES, ISET BIT<00> FOR JAMUPP EXPECTED
18706 P3, BUTA(DIAGNOSE), ISTART THE XFR TO BM SEQUENCE
18707 NEXT, J/LOADFPSCC2
(7264) DCS[0.00,0.0,0.0] BM[1111..00.00..11.01..101..000...0.0,0.0,0.0,0.0000...0..0000,0...11,011...010,110,101]

18708
18709 7265: I(FREE)
18710 LOADFPSCC2:
18711 NEXT, XFR-TO-BM[LOADNZW4] ILOAD PAGE, POINT UPF AT BM CODE
(7265) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..100...0.0,0.0,0.0,0.0000...0..0000,0...11,100...011,011,000]

18712
18713 ILOADNZW4:
18714 P2-T, D_CSPB[16], I(THIS WORD ACTUALLY COMES OUT OF BM ROMS)
18715 NEXT, J/LOADNZW5 IPUT "MD" = CSP[16] INTO D
18716
18717 ILOADNZW5:
18718 P2, FPS[CC]_D[3-0], ILOAD FPS<3:0>
18719 P3, BUTA(DIAGNOSE), IBEGIN XFR SEQUENCE BACK TO DCS ROMS
18720 NEXT, J/NZERO2
18721
18722 INZERO2:
18723 NEXT, GOTO-PAGE(4),
18724 J/XXXXXX
18725
18726 I(CONTROL NOW RETURNS TO DCS AT "JAMUPP001" WORD)
18727 I4777:
18728 IJAMUPP001:
18729 I *** SEE FLOWS ON SUBSEQUENT PAGE ***
18730
18731
18732
18733
18734
18735
18736
18737
18738
18739
18740
18741
18742
18743
18744
18745
18746
18747
18748

```

```

18698 (1) LOADING SR MUST BE SETUP
18699 (2) CSP[16] SETUP WITH BITS TO LOAD
18700 (3) CSP[00] CONTAINS RETURN MICROADDRESS IN BITS<14:0>
18701
18702
18703 7264: I(FREE)
18704 LOADFPSCC:
18705 P2-T, SP_ALL-ONES, ISET BIT<00> FOR JAMUPP EXPECTED
18706 P3, BUTA(DIAGNOSE), ISTART THE XFR TO BM SEQUENCE
18707 NEXT, J/LOADFPSCC2
(7264) DCS[0.00,0.0,0.0] BM[1111..00.00..11.01..101..000...0.0,0.0,0.0,0.0000...0..0000,0...11,011...010,110,101]

18708
18709 7265: I(FREE)
18710 LOADFPSCC2:
18711 NEXT, XFR-TO-BM[LOADNZW4] ILOAD PAGE, POINT UPF AT BM CODE
(7265) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..100...0.0,0.0,0.0,0.0000...0..0000,0...11,100...011,011,000]

18712
18713 ILOADNZW4:
18714 P2-T, D_CSPB[16], I(THIS WORD ACTUALLY COMES OUT OF BM ROMS)
18715 NEXT, J/LOADNZW5 IPUT "MD" = CSP[16] INTO D
18716
18717 ILOADNZW5:
18718 P2, FPS[CC]_D[3-0], ILOAD FPS<3:0>
18719 P3, BUTA(DIAGNOSE), IBEGIN XFR SEQUENCE BACK TO DCS ROMS
18720 NEXT, J/NZERO2
18721
18722 INZERO2:
18723 NEXT, GOTO-PAGE(4),
18724 J/XXXXXX
18725
18726 I(CONTROL NOW RETURNS TO DCS AT "JAMUPP001" WORD)
18727 I4777:
18728 IJAMUPP001:
18729 I *** SEE FLOWS ON SUBSEQUENT PAGE ***
18730
18731
18732
18733
18734
18735
18736
18737
18738
18739
18740
18741
18742
18743
18744
18745
18746
18747
18748

```

```

18749      NEXT, J/DINTOIR
(7266) DCS[0.00,0.0,0.0] BM[1111.00,00.00,00.000...111...0.1.0.0.0...0.0000...0.0000,0...11,000...010,110,111]
18750
18751      7267: 1(FREE)
18752      DINTOIR:
18753      P2-U, IR_DBUF-[I],
18754      P3-U, DBUF_D-[I],
18755      NEXT, J/DBUFINTOIR
(7267) DCS[0.00,0.0,0.0] BM[1000.00,00.00,01.000...100...0.0.0.0.0...1.1011...0.0000,0...11,000...010,111,000]
18756
18757      7270: 1(FREE)
18758      DBUFINTOIR:
18759      EMITC, EMIT/125200,
18760      P2-U, IR_DBUF,
18761      P3-U, DBUF_D,
18762      NEXT, J/RESETUCOMP
(7270) DCS[0.00,0.0,0.0] BM[1010.00,10.10,10.000...000...0.0.0.0.0...1.1010...0.0000,0...11,000...010,111,001]
18763
18764      7271: 1(FREE)
18765      RESETUCOMP:
18766      SELECT, UCON-PROC,
18767      ENARLF, EN-CLK-IR(15-00),
18768      BUSDIN_EMIT(15-00),
18769      PO, SET-UCON-CONTROL,
18770      NEXT, BUTA(RETURN),
18771      J/BUTERROR?
(7271) DCS[0.00,0.0,0.0] BM[0000.00,00.00,01.000...100...0.0.0.0.0...1.1001...0.0000,0...11,111...011,111,110]
18772
18773      I - - - - -
18774
18775      7272: 1(FREE)
18776      SRINTOIRS:
18777      P2-T, D_SR, SAVE-D[C],
18778      NEXT, J/DINTOIRS
(7272) DCS[0.00,0.0,0.0] BM[1111.00,00.00,00.000...111...0.1.0.0.0...0.0000...0.0000,0...11,000...010,111,011]
18779
18780      7273: 1(FREE)
18781      DINTOIRS:
18782      P2-U, IR_DBUF-[I],
18783      P3-U, DBUF_D-[I],
18784      NEXT, J/DBUFINTOIRS
(7273) DCS[0.00,0.0,0.0] BM[1000.00,00.00,01.000...100...0.0.0.0.0...1.1011...0.0000,0...11,000...010,111,100]
18785
18786      7274: 1(FREE)
18787      DBUFINTOIRS:
18788      EMITC, EMIT/125200,
18789      P2-U, IR_DBUF,
18790      P3-U, DBUF_D,
18791      NEXT, J/RESETUCOMP5
(7274) DCS[0.00,0.0,0.0] BM[1010.00,10.10,10.000...000...0.0.0.0.0...1.1010...0.0000,0...11,000...010,111,101]
18792
18793      7275: 1(FREE)
18794      RESETUCOMP5:
18795

```

```

18796      SELECT, UCON-PROC,
18797      ENARLF, EN-CLK-IR(15-00),
18798      BUSDIN_EMIT(15-00),
18799      PO, SET-UCON-CONTROL,
18800      NEXT, J/BUTINSTRS
(7275) DCS[0.00,0.0,0.0] BM[0000.00,00.00,01.000...100...0.0.0.0.0...1.1001...0.0000,0...11,000...011,000,001]
18801
18802
18803
18804
18805
18806
18807      I PAGE=====
18808
18809
18810      .TOC * JAM UPP SERVICE SUBROUTINE
18811
18812      I
18813      I IF A JAMUPP TO 0777/4777 OCCURS, ACTION DEPENDS UPON SR<00> CONTENTS:
18814      I
18815      I SR<00>=1 -> RESET SR TO (000000), DO A BUTA(RETURN) TO ADDRESS IN CSP(00)
18816      I
18817      I SP<00>=0 -> SIGNAL ERROR, LOG STATUS, RETURN VIA "CUA" CONTENTS
18818      I ERROR-CODE := INDICATES LAST TEST EXECUTED
18819      I
18820
18821      4777:
18822      JAMUPP001:
18823      P3-T, SR_D,
18824      NEXT, BUTR(SR00),
18825      IF SR<00>=1, EXPECTED JAM, GOTO "JAMUPP002B"
18826      IF SR<00>=0, NO JAM WAS EXPECTED, FORCE ERROR AT "JAMUPP003"
18827      J/JAMUPP003
(4777) DCS[0.00,0.0,0.0] BM[1111.00,00.01,01.000...000...1.0.1.0.0...0.0000...0.0000,0...00,000...111,101,110]
18828
18829      4757:
18830      JAMUPP002B:
18831      P2-T, D_CSPD(D00), SAVE-D[C],
18832      NEXT, J/JAMUPP002C
(4757) DCS[0.00,0.0,0.0] BM[1010.10,00.00,00.000...111...0.1.0.0.0...0.1111...0.0000,0...11,000...100,011,100]
18833
18834      4434: 1(FREE)
18835      JAMUPP002C:
18836      PO, RETURN_D[14-03], PAGE(7),
18837      P2-T, D_SR, SAVE-D[C],
18838      NEXT, J/JAMUPP002D
(4434) DCS[0.00,0.0,0.0] BM[1111.00,00.00,00.000...111...0.1.0.0.0...0.0000...0.0000,0...11,101...010,001,011]
18839
18840      7213: 1(FREE)
18841      JAMUPP002D:
18842      P2-T, SR_ZERO,
18843      NEXT, BUTA(RETURN),
18844      J/BUTERROR?

```

```
(7213) DCS[0.00.0.0.0.0] RM[0011..00.00..00.00..0000..0.0.1..0..0.0000..0..0000.0...11.111...011.111.110]
18845
18846 4756:
18847 JAMUPP003:
18848 SELECT, BUSDIN_CUA-[I], IENABLE READ JAMUPP MICROADDR
18849 NEXT, BUT[ERROR], I WAS ERROR PREVIOUSLY SET ?
18850 J/JAMUPP005 I YES/JAMUPP004, NO/JAMUPP005
(4756) DCS[0.00.0.1.0.0] BM[0000..00.00..01.01..0000..0000..0.0.0..0..0.1.1001..0..0000.0...11.000...111.100.111]
18851
18852 1* ENTER HERE IF "ERROR" WAS PREVIOUSLY SET (INHIBIT LOG)
18853 4746:
18854 JAMUPP004:
18855 P3, CSPD[00]_LOG-CUA, IGET RETURN LOC <- SAVED CUA
18856 P3-T, BUT[CUA-TRACK], ISTART CUA GOING AGAIN
18857 NEXT, J/JAMUPP005 IINHIBIT LOG IF PREV ERROR
(4746) DCS[0.00.0.0.0.0] BM[0000..10.00..00.00..0000..0000..1.0.0..0..0.1111..1..0000.0...11.001...111.101.111]
18858
18859 1* ENTER HERE IF "ERROR" NOT YET SET (LOG FIRST MODE)
18860 4747:
18861 JAMUPP005:
18862 P0, DCS-CTR(C0.), IFORCE ERROR WITH PREV ERROR-CODE/ENUA, TNUA=(4747)
18863 P3, CSPD[00]_LOG-CUA, IRETURN LOC <- RETURN ADDR FROM CUA
18864 NEXT, J/JAMUPP006 IAND GO LOG REGISTERS THIS TIME
(4747) DCS[0.00.1.0.0.0] BM[1111..10.00..00.00..0000..0000..0.0.0..0..0.1111..1..0000.0...11.000...100.011.101]
18865
18866 4435: 1(FREE)
18867 JAMUPP006:
18868 SELECT, BUSDIN_SERVICE-[I], ISERVICE REGISTER
18869 NEXT, J/JAMUPP007
(4435) DCS[0.00.0.0.0.0] BM[0100..01.00..00.00..0000..0000..0.0.0..0..0.1.1001..0..0000.0...11.000...100.011.110]
18870
18871 4436: 1(FREE)
18872 JAMUPP007:
18873 P3, CSPD[01]_LOG-SERVICE, ILOG SERVICE INFO REGISTER
18874 NEXT, J/JAMUPP010
(4436) DCS[0.00.0.0.0.0] BM[0000..10.00..00.00..0000..0000..0.0.0..0..0.1110..1..0000.0...11.000...100.011.111]
18875
18876 4437: 1(FREE)
18877 JAMUPP010:
18878 SELECT, BUSDIN_JAM-[I], IJAM REGISTER
18879 NEXT, J/JAMUPP011
(4437) DCS[0.00.0.0.0.0] BM[1100..00.00..00.00..0000..0000..0.0.0..0..0.1.1001..0..0000.0...11.000...100.100.000]
18880
18881 4440: 1(FREE)
18882 JAMUPP011:
18883 P3, CSPD[02]_LOG-JAM, ILOG JAMUPP CAUSE INFO
18884 BUT[CUA-TRACK], IRESET CUA TO TRACK, IN CASE ANOTHER JAM COMES
18885 NEXT, J/JAMUPP002B IGO TO TOP OF THIS PAGE, AND RETURN INLINE
(4440) DCS[0.00.0.0.0.0] BM[0000..10.00..00.00..0000..0000..0.0.0..0..0.1101..1..0000.0...11.001...111.101.111]
18886
18887
18888
18889
18890
```

```
18891 1.PAGE=====
18892
18893
18894 .TDC * MICROBRANCH [BUT] TAKEOFF WORDS
18895
18896 !=====
18897 1*
18898 1* UNWORDS: 000 + 056
18899 1*
18900 1* FUNCTION: THESE WORDS ARE THE INACTIVE-"BUT" (BRANCHING TYPE)
18901 1* "TAKEOFF", OR SUBROUTINE MICROWORDS, ANY TEST WHICH
18902 1* REQUIRES A SPECIFIC "BUT" CONDITION TO BE TESTED WILL USE
18903 1* ONE OF THESE MICROWORDS AS A TAKEOFF POINT INTO THE "BUT
18904 1* TARGET TABLE" (DESCRIBED NEXT), WHERE AN ENUNTIUA
18905 1* COMPARISON WILL HAVE BEEN PREVIOUSLY ENABLED (VIA SET-
18906 1* TING THE DIAGNOSTIC COUNTER TO THE APPROPRIATE VALUE.
18907 1*
18908 !=====
18909
18910
18911
18912
18913 1** BUT 00 **
18914 IFULL WIDTH IS BUT[SR<3:0>]
18915 7276: 1(FREE)
18916 BUTSR3=0:
18917 NEXT, BUT(SR3=0), ITO (400)-(417), M4
18918 J/ZTARGET400 INO MASK
(7276) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0..0..0.0000..0..0000.0...00.000...100.000.000]
18919
18920
18921 1** BUT 01 **
18922 IFULL WIDTH IS BUT[IR<15:12>]
18923 7300: 1(FREE)
18924 BUTIR15=12:
18925 NEXT, BUT(IR15=12), ITO (400)-(417), M4
18926 J/ZTARGET400 INO MASK
(7300) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0..0..0.0000..0..0000.0...00.001...100.000.000]
18927
18928
18929
18930 1** BUT 02 **
18931 IFULL WIDTH IS BUT[INSTR 5]
18932 7301: 1(FREE)
18933 BUTINSTR5:
18934 NEXT, BUT(INSTR5), ITO (400)-(437), M5
J/ZTARGET400 INO MASK
(7301) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0..0..0.0000..0..0000.0...00.010...100.000.000]
18935
18936
18937 1** BUT 03 **
18938 IFULL WIDTH IS BUT[IR11FLTP3<3:0>]
18939 7302: 1(FREE)
18940 BUTIR11FLTP3=0:
18941 NEXT, BUT(IR11FLTP3=0), ITO (400)-(437), M5
```

```
18942 J/ZTARGET400 NO MASK
(7302) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...011...100,000,000]
18943
18944
18945 *** BUT 04 ***
18946 IFULL WIDTH IS BUT[IR<9:6>]
18947 7304: 1(FREE)
18948 RUTIR9=6:
18949 NEXT, BUT(IR9=6), ITO (400)=(417), W4
18950 J/ZTARGET400 NO MASK
(7304) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...010...100,000,000]
18951
18952
18953 *** BUT 05 ***
18954 IFULL WIDTH IS BUT[MOV-DR7:IR<5:3>]
18955 7305: 1(FREE)
18956 BUTMOVDR7IPS=3:
18957 NEXT, BUT(MOV-DR7:IR5=3), ITO (400)=(417), W4
18958 J/ZTARGET400 NO MASK
(7305) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...010...100,000,000]
18959
18960
18961 *** BUT 06 ***
18962 IFULL WIDTH IS BUT[INSTR 1] *** N.B.: THIS BUT IS ALSO ACTIVE ***
18963 7306: 1(FREE)
18964 RUTINSTR1:
18965 NEXT, BUTA(INSTR1), ITO (408)=(777), W8
18966 J/ZTARGET400 NO MASK
(7306) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...010...100,000,000]
18967
18968
18969 *** BUT 07 ***
18970 IFULL WIDTH IS BUT[BG-SERV-H+FP-SERV-H#D(C)FPPRET<1:0>]
18971 7307: 1(FREE)
18972 BUTRGFPSEV:
18973 NEXT, BUT(BGSEV-PPSERV#D(C)FPPRET), ITO (407)=(417), W1
18974 J/ZTARGET407 MASK OUT D(C), FPPRET<1:0>
(7307) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...011...100,000,111]
18975
18976 7310: 1(FREE)
18977 BUTD(C)C:
18978 NEXT, BUT(BGSEV-PPSERV#D(C)FPPRET), ITO (413),(417), W1
18979 J/ZTARGET413 MASK OUT BG-SERV-H+FP-SERV-H, FPPRET<1:0>
(7310) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...011...100,001,011]
18980
18981
18982 *** BUT 10 ***
18983 IFULL WIDTH IS BUT[COUNT7:DOUT07:FPS05]
18984 7311: 1(FREE)
18985 RUTCOUNT7DOUT7:
18986 NEXT, BUT(COUNT7:DOUT07:FPS05), ITO (401),(403),(405),(407), W2
18987 J/ZTARGET401 MASK OUT FPS05
(7311) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...01,000...100,000,001]
18988
```

```
18989 7312: 1(FREE)
18990 BUTFPS05:
18991 NEXT, BUT(COUNT07:DOUT07:FPS05), ITO (406),(407), W1
18992 J/ZTARGET406 MASK OUT COUNT7, DOUT7
(7312) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...01,000...100,000,110]
18993
18994
18995 *** BUT 11 ***
18996 IFULL WIDTH IS BUT[DM0:SM0:BYTE]
18997 7313: 1(FREE)
18998 BUTDM0SM0BYTE:
18999 NEXT, BUT(DM0:SM0:BYTE), ITO (400)=(407), W3
19000 J/ZTARGET400 NO MASK
(7313) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...01,001...100,000,000]
19001
19002
19003 *** BUT 12 ***
19004 IFULL WIDTH IS BUT[GD<3:2>]
19005 7314: 1(FREE)
19006 RUTGD3=2:
19007 NEXT, BUT(GD3=2), ITO (400)=(403), W2
19008 J/ZTARGET400 NO MASK
(7314) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...01,010...100,000,000]
19009
19010
19011 *** BUT 13 ***
19012 IFULL WIDTH IS BUT[SR<1:0>:COUNT-IS=377] *** N.B.: THIS BUT IS ALSO ACTIVE ***
19013 7315: 1(FREE)
19014 BUTSR1=0:
19015 NEXT, BUTA(SR1=0:COUNT-IS=377), ITO (401),(403),(405),(407), W2
19016 J/ZTARGET401 MASK OUT COUNT-IS=377
(7315) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...01,011...100,000,001]
19017
19018
19019 *** BUT 14 ***
19020 IFULL WIDTH IS BUT[BG-SERVICE-L#MFSS#MULTIPLE]
19021 7316: 1(FREE)
19022 BUTBGSEVLI:
19023 NEXT, BUT(BG-SERVICE-L#MFSS#MULTIPLE), ITO (403),(407), W1
19024 J/ZTARGET403 MASK OUT MFSS, MULTIPLE
(7316) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...01,100...100,000,011]
19025
19026 7317: 1(FREE)
19027 BUTMASKPS(T):
19028 SETUP, TEST(MASKED-PS(T)), ISELECT MULTIPLE BUT
19029 NEXT, J/BUTMNT000 INEXT
(7317) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...11,000...011,010,000]
19030
19031 7320: 1(FREE)
19032 BUTMNT000:
19033 SETUP, TEST(MASKED-PS(T)), ISELECT MULTIPLE BUT
19034 NEXT, BUT(BG-SERVICE-L#MFSS#MULTIPLE), ITO (406),(407), W1
19035 J/ZTARGET406 MASK OUT BG-SERVICE-L, MFSS
(7320) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000..0.0,0.0..0.0000...0..0000,0...01,100...100,000,110]
```

```
19036 7321: 1(FREE)
19037 BUTM001:
19038 SETUP, TEST(D00), ISELECT MULTIPLE BUT
19039 NEXT, J/BUTMNX001 INEXT
19040 (7321) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..001...0.0.0..0...0.0000...0..0000,0...11,000...011,010,010]

19041 7322: 1(FREE)
19042 BUTMNX001:
19043 SETUP, TEST(D00), ISELECT MULTIPLE BUT
19044 NEXT, BUT(BG-SERVICE-L,MFSS#MULTIPLE), ITO (406),(407), W1
19045 J/ZTARGET406 IMASK OUT BG-SERVICE-L, MFSS
19046 (7322) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..001...0.0.0..0...0.0000...0..0000,0...01,100...100,000,110]

19047 7323: 1(FREE)
19048 BUTMPS(N):
19049 SETUP, TEST(PS(N)), ISELECT MULTIPLE BUT
19050 NEXT, J/BUTMNX002 INEXT
19051 (7323) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..010...0.0.0..0...0.0000...0..0000,0...11,000...011,010,100]

19052 7324: 1(FREE)
19053 BUTMNX002:
19054 SETUP, TEST(PS(N)), ISELECT MULTIPLE BUT
19055 NEXT, BUT(BG-SERVICE-L,MFSS#MULTIPLE), ITO (406),(407), W1
19056 J/ZTARGET406 IMASK OUT BG-SERVICE-L, MFSS
19057 (7324) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..010...0.0.0..0...0.0000...0..0000,0...01,100...100,000,110]

19058 7325: 1(FREE)
19059 BUTMFLAG7:
19060 SETUP, TEST(FLAG7), ISELECT MULTIPLE BUT
19061 NEXT, J/BUTMNX003 INEXT
19062 (7325) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..011...0.0.0..0...0.0000...0..0000,0...11,000...011,010,110]

19063 7326: 1(FREE)
19064 BUTMNX003:
19065 SETUP, TEST(FLAG7), ISELECT MULTIPLE BUT
19066 NEXT, BUT(BG-SERVICE-L,MFSS#MULTIPLE), ITO (406),(407), W1
19067 J/ZTARGET406 IMASK OUT BG-SERVICE-L, MFSS
19068 (7326) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..011...0.0.0..0...0.0000...0..0000,0...01,100...100,000,110]

19069 7327: 1(FREE)
19070 BUTMEXFLAG1:
19071 SETUP, TEST(EXFLAG1), ISELECT MULTIPLE BUT
19072 NEXT, J/BUTMNX004 INEXT
19073 (7327) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..100...0.0.0..0...0.0000...0..0000,0...11,000...011,011,000]

19074 7330: 1(FREE)
19075 BUTMNX004:
19076 SETUP, TEST(EXFLAG1), ISELECT MULTIPLE BUT
19077 NEXT, BUT(BG-SERVICE-L,MFSS#MULTIPLE), ITO (406),(407), W1
19078 J/ZTARGET406 IMASK OUT BG-SERVICE-L, MFSS
19079 (7330) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..100...0.0.0..0...0.0000...0..0000,0...01,100...100,000,110]

19080 7331: 1(FREE)
```

```
19082 BUTMFLTPIS:
19083 SETUP, TEST(FLTPIS), ISELECT MULTIPLE BUT
19084 NEXT, J/BUTMNX005 INEXT
19085 (7331) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..101...0.0.0..0...0.0000...0..0000,0...11,000...011,011,010]

19086 7332: 1(FREE)
19087 BUTMNX005:
19088 SETUP, TEST(FLTPIS), ISELECT MULTIPLE BUT
19089 NEXT, BUT(BG-SERVICE-L,MFSS#MULTIPLE), ITO (406),(407), W1
19090 J/ZTARGET406 IMASK OUT BG-SERVICE-L, MFSS
19091 (7332) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..101...0.0.0..0...0.0000...0..0000,0...01,100...100,000,110]

19092 7333: 1(FREE)
19093 BUTMEXFLAG2:
19094 SETUP, TEST(EXFLAG2), ISELECT MULTIPLE BUT
19095 NEXT, J/BUTMNX006 INEXT
19096 (7333) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..110...0.0.0..0...0.0000...0..0000,0...11,000...011,011,100]

19097 7334: 1(FREE)
19098 BUTMNX006:
19099 SETUP, TEST(EXFLAG2), ISELECT MULTIPLE BUT
19100 NEXT, BUT(BG-SERVICE-L,MFSS#MULTIPLE), ITO (406),(407), W1
19101 J/ZTARGET406 IMASK OUT BG-SERVICE-L, MFSS
19102 (7334) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..110...0.0.0..0...0.0000...0..0000,0...01,100...100,000,110]

19103 7336: 1(FREE)
19104 BUTMINITJAM:
19105 SETUP, TEST(INIT-JAM), ISELECT MULTIPLE BUT
19106 NEXT, J/BUTMNX007 INEXT
19107 (7336) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..111...0.0.0..0...0.0000...0..0000,0...11,000...011,100,000]

19108 7340: 1(FREE)
19109 BUTMNX007:
19110 SETUP, TEST(INIT-JAM), ISELECT MULTIPLE BUT
19111 NEXT, BUT(BG-SERVICE-L,MFSS#MULTIPLE), ITO (406),(407), W1
19112 J/ZTARGET406 IMASK OUT BG-SERVICE-L, MFSS
19113 (7340) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..111...0.0.0..0...0.0000...0..0000,0...01,100...100,000,110]

19114 *** BUT 15 ***
19115 IFULL WIDTH IS BUT(D14:000=00D15)
19116 7341: 1(FREE)
19117 BUTD-15-ZERO:
19118 NEXT, BUT(D14=00EQ0D15), ITO (400)-(403), W2
19119 J/ZTARGET400 INO MASK
19120 (7341) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0.0..0...0.0000...0..0000,0...01,101...100,000,000]

19121 *** BUT 16 ***
19122 IFULL WIDTH IS BUT(IR118PS15)
19123 7342: 1(FREE)
19124 BUTIR118:
19125 NEXT, BUT(IR118PS15), ITO (401),(403), W1
19126 J/ZTARGET401 IMASK OUT PS15
```

```
(7342) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...01.110...100.000,001]
19129
19130 7343: 1(FREE)
19131 BUTPB15:
19132 NEXT, BUT(IR11#PB15), ITO (402),(403), W1
19133 J/ZTARGET402 IMASK OUT IR11
(7343) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...01.110...100.000,010]
19134
19135
19136 1*** BUT 17 ***
19137 IFULL WIDTH IS BUT(COUNT-IS-377#D[C]) *** N.B.: THIS BUT IS ALSO ACTIVE ***
19138 7344: 1(FREE)
19139 BUTD(C)A:
19140 NEXT, BUTA(COUNT-IS-377#D[C]), ITO (402),(403), W1
19141 J/ZTARGET402 IMASK OUT COUNT-IS-377
(7344) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...01.111...100.000,010]
19142
19143
19144 1*** BUT 20 ***
19145 IFULL WIDTH IS BUT(PREFETCH-L#SERVICE) *** N.B.: THIS BUT IS ALSO ACTIVE ***
19146 7345: 1(FREE)
19147 BUTSERVICE:
19148 NEXT, BUTA(PREFETCH-L#SERVICE), ITO (402),(403), W1
19149 J/ZTARGET402 IMASK OUT PREFETCH-L
(7345) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.000...100.000,010]
19150
19151
19152 1*** BUT 21 ***
19153 IFULL WIDTH IS BUT(VECTOR-LOAD#DR6/7L)
19154 7346: 1(FREE)
19155 BUTVECTLOAD:
19156 NEXT, BUT(VECTOR-LOAD#DR6-7L), ITO (401),(403), W1
19157 J/ZTARGET401 IMASK OUT DR6/7-L
(7346) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.001...100.000,001]
19158
19159 7347: 1(FREE)
19160 BUTDR6-7L:
19161 NEXT, BUT(VECTOR-LOAD#DR6-7L), ITO (402),(403), W1
19162 J/ZTARGET402 IMASK OUT VECTOR-LOAD
(7347) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.001...100.000,010]
19163
19164
19165 1*** BUT 22 ***
19166 ITHIS IS AN ACTIVE BUT - NO BRANCH MODIFICATION
19167 I(THIS BUT IS NOT USED WITH THE TARGET TABLE)
19168
19169
19170 1*** BUT 23 ***
19171 IFULL WIDTH IS BUT(D[C]#BA00)
19172 7350: 1(FREE)
19173 BUTD(C)B:
19174 NEXT, BUT(D[C]#BA00), ITO (401),(403), W1
19175 J/ZTARGET401 IMASK OUT BA00
(7350) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.011...100.000,001]
```

```
19176
19177 7351: 1(FREE)
19178 BUTBA00:
19179 NEXT, BUT(D[C]#BA00), ITO (402),(403), W1
19180 J/ZTARGET402 IMASK OUT D[C]
(7351) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.011...100.000,010]
19181
19182
19183 1*** BUT 24 ***
19184 IFULL WIDTH IS BUT(OTHER-JAM#FP-PROC)
19185 7352: 1(FREE)
19186 BUTOTHERJAM:
19187 NEXT, BUT(OTHER-JAM#FP-PROC), ITO (401),(403), W1
19188 J/ZTARGET401 IMASK OUT FLTPF-PROC-H
(7352) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.100...100.000,001]
19189
19190 7354: 1(FREE)
19191 BUTFPFPROC:
19192 NEXT, BUT(OTHER-JAM#FP-PROC), ITO (402),(403), W1
19193 J/ZTARGET402 IMASK OUT OTHER-JAM
(7354) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.100...100.000,010]
19194
19195
19196 1*** BUT 25 ***
19197 IFULL WIDTH IS BUT(COUNT-IS-377) *** N.B.: THIS BUT IS ALSO ACTIVE ***
19198 I(THIS BUT IS NOT USED WITH THE TARGET TABLE)
19199
19200
19201 1*** BUT 26 ***
19202 IFULL WIDTH IS BUT(INTR-HIGH#INSTR-BRANCH-L)
19203 7355: 1(FREE)
19204 BUTINTRHIGH:
19205 NEXT, BUT(INTR-HIGH#INSTR-BRANCH-L), ITO (401),(403), W1
19206 J/ZTARGET401 IMASK OUT INSTR-BRANCH-L
(7355) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.110...100.000,001]
19207
19208 7356: 1(FREE)
19209 BUTINSTRBRANCH:
19210 NEXT, BUT(INTR-HIGH#INSTR-BRANCH-L), ITO (402),(403), W1
19211 J/ZTARGET402 IMASK OUT INTR-HIGH-H
(7356) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.110...100.000,010]
19212
19213
19214 1*** BUT 27 ***
19215 IFULL WIDTH IS BUT(PREFETCH-JAM#FP-FD)
19216 7360: 1(FREE)
19217 BUTPREFETCHJAM:
19218 NEXT, BUT(PREFETCH-JAM#FP-FD), ITO (401),(403), W1
19219 J/ZTARGET401 IMASK OUT FLTPF-FD-H
(7360) DCS[0.00,0.0,0.0] BM[0000..00.00..00.00..000..000...0.0,0.0,0.0...0.0000...0..0000.0...10.111...100.000,001]
19220
19221 7371: 1(FREE)
19222 BUTFPFD:
19223 NEXT, BUT(PREFETCH-JAM#FP-FD), ITO (402),(403), W1
```

```
19224 J/ZTARGET402 MASK OUT PREFETCH-JAM-W
(7371) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.10.111.100.000.010]
19225
19226
19227
19228
19229
19230 1.PAGE=====
19231
19232
19233 .TDC * MICROBRANCH [BUT] TARGET WORDS
19234
19235 |*****
19236 |*
19237 |* UNWORDS: 000 + 400
19238 |*
19239 |* FUNCTION: TARGET BUT TABLE
19240 |* ALL THE ABOVE BUTS TARGET INTO THIS TABLE OF MICROWORDS,
19241 |* ALL OF WHICH DO A BUTA(RETURN). IN THIS MANNER, ANY OF
19242 |* THE ABOVE BRANCHES MAY BE EXECUTED, AND CONTROL WILL ALWAYS
19243 |* RETURN TO WHERE THE "BUT TMS" SUBROUTINE WAS CALLED.
19244 |*
19245 |*****
19246
19247
19248
19249 |*** THE TARGET BUT TABLE ***
19250
19251 7400:
19252 ZTARGET400: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7400) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19253 7401:
19254 ZTARGET401: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7401) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19255 7402:
19256 ZTARGET402: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7402) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19257 7403:
19258 ZTARGET403: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7403) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19259 7404:
19260 ZTARGET404: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7404) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19261 7405:
19262 ZTARGET405: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7405) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19263 7406:
19264 ZTARGET406: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7406) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19265 7407:
19266 ZTARGET407: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7407) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19267 7410:
19268 ZTARGET410: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
```

```
(7410) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19269 7411:
19270 ZTARGET411: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7411) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19271 7412:
19272 ZTARGET412: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7412) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19273 7413:
19274 ZTARGET413: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7413) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19275 7414:
19276 ZTARGET414: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7414) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19277 7415:
19278 ZTARGET415: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7415) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19279 7416:
19280 ZTARGET416: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7416) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19281 7417:
19282 ZTARGET417: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7417) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19283 7420:
19284 ZTARGET420: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7420) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19285 7421:
19286 ZTARGET421: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7421) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19287 7422:
19288 ZTARGET422: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7422) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19289 7423:
19290 ZTARGET423: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7423) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19291 7424:
19292 ZTARGET424: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7424) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19293 7425:
19294 ZTARGET425: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7425) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19295 7426:
19296 ZTARGET426: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7426) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19297 7427:
19298 ZTARGET427: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7427) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19299 7430:
19300 ZTARGET430: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7430) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19301 7431:
19302 ZTARGET431: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7431) DCS[0.00.0.0.0.0] BM[0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0.0.0.0.11.111.011.111.110]
19303 7432:
19304 ZTARGET432: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
```

[illegible]

```

(7454) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19341 7455;
19342 ZTARGET455; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7456) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19343 7456;
19344 ZTARGET456; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7456) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19345 7457;
19346 ZTARGET457; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7457) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19347 7460;
19348 ZTAPGET460; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7460) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19349 7461;
19350 ZTARGET461; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7461) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19351 7462;
19352 ZTARGET462; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7462) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19353 7463;
19354 ZTAPGET463; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7463) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19355 7464;
19356 ZTARGET464; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7464) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19357 7465;
19358 ZTARGET465; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7465) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19359 7466;
19360 ZTAPGET466; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7466) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19361 7467;
19362 ZTARGET467; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7467) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19363 7470;
19364 ZTARGET470; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7470) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19365 7471;
19366 ZTARGET471; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7471) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19367 7472;
19368 ZTARGET472; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7472) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19369 7473;
19370 ZTARGET473; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7473) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19371 7474;
19372 ZTARGET474; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7474) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19373 7475;
19374 ZTARGET475; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7475) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..000..0.0.0..0.0..0.0000...0..0000.0...11.111...011.111.110]
19375 7476;
19376 ZTARGET476; NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN

```

```

(7476) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19377 7477:
19378 ZTARGET477: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7477) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19379 7500:
19380 ZTARGET500: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7500) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19381 7501:
19382 ZTARGET501: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7501) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19383 7502:
19384 ZTARGET502: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7502) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19385 7503:
19386 ZTARGET503: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7503) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19387 7504:
19388 ZTARGET504: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7504) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19389 7505:
19390 ZTARGET505: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7505) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19391 7506:
19392 ZTARGET506: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7506) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19393 7507:
19394 ZTARGET507: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7507) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19395 7510:
19396 ZTARGET510: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7510) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19397 7511:
19398 ZTARGET511: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7511) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19399 7512:
19400 ZTARGET512: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7512) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19401 7513:
19402 ZTARGET513: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7513) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19403 7514:
19404 ZTARGET514: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7514) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19405 7515:
19406 ZTARGET515: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7515) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19407 7516:
19408 ZTARGET516: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7516) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19409 7517:
19410 ZTARGET517: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7517) DCS(0.00,0.0,0.0) BM(0000.00.00.00.00.000.000.0.0.0.0.0.0.0000.0.0.0000.0.11.111.011.111.110)
19411 7520:
19412 ZTARGET520: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN

```

[illegible]

[illegible][illegible]

[illegible]

```

(7674) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19629 76751
19630 ZTARGET6751 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7675) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19631 76761
19632 ZTARGET6761 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7676) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19633 76771
19634 ZTARGET6771 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7677) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19635 77001
19636 ZTARGET7001 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7700) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19637 77011
19638 ZTARGET7011 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7701) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19639 77021
19640 ZTARGET7021 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7702) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19641 77031
19642 ZTARGET7031 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7703) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19643 77041
19644 ZTARGET7041 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7704) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19645 77051
19646 ZTARGET7051 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7705) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19647 77061
19648 ZTARGET7061 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7706) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19649 77071
19650 ZTARGET7071 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7707) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19651 77101
19652 ZTARGET7101 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7710) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19653 77111
19654 ZTARGET7111 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7711) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19655 77121
19656 ZTARGET7121 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7712) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19657 77131
19658 ZTARGET7131 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7713) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19659 77141
19660 ZTARGET7141 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7714) DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19661 77151
19662 ZTARGET7151 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7715) DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..0000..0.0.0.0..0.0.0000..0.0000.0...1111...011.111.110]
19663 77161
19664 ZTARGET7161 NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN

```

[illegible]

```

19701      7740: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0000..0.0000.0...111...011.111.110]
19702      ZTARGET741: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19703      7741: DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19704      ZTARGET742: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19705      7742: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19706      ZTARGET743: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19707      7743: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19708      ZTARGET744: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19709      7744: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19710      ZTARGET745: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19711      7745: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19712      ZTARGET746: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19713      7746: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19714      ZTARGET747: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19715      7747: DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19716      ZTARGET750: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19717      7750: DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19718      ZTARGET751: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19719      7751: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19720      ZTARGET752: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19721      7752: DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19722      ZTARGET753: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19723      7753: DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19724      ZTARGET754: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19725      7754: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19726      ZTARGET755: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19727      7755: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19728      ZTARGET756: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19729      7756: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19730      ZTARGET757: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19731      7757: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19732      7760: ZTARGET760: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19733      7760: DCS[0.00.0.0.0.0] RM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19734      ZTARGET761: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
19735      7761: DCS[0.00.0.0.0.0] BM[0000..00.00..00.00..0000..00.0.0.0.0..0.0.0000..0.0000.0...111...011.111.110]
19736      7762: ZTARGET762: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN

```

```

(7767) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19737 7763:
19738 ZTARGET763: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7763) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19739 7764:
19740 ZTARGET764: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7764) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19741 7765:
19742 ZTARGET765: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7765) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19743 7766:
19744 ZTARGET766: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7766) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19745 7767:
19746 ZTARGET767: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7767) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19747 7770:
19748 ZTARGET770: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7770) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19749 7771:
19750 ZTARGET771: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7771) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19751 7772:
19752 ZTARGET772: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7772) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19753 7773:
19754 ZTARGET773: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7773) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19755 7774:
19756 ZTARGET774: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7774) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19757 7775:
19758 ZTARGET775: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7775) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19759 7776:
19760 ZTARGET776: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7776) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19761 7777:
19762 ZTARGET777: NEXT, BUTA(RETURN), J/BUTERROR7 ICOMPARE MICROADDRESS, THEN RETURN
(7777) DCS(0.00,0.0,0.0) BM(0000.00,00.00,00.000.000.000.0.0.0.0.0.0.0.0000.0.0.0000.0.0.11.111.011.111.10)
19763
19764
19765 I*****
19766 .TOC * END OF KD11-K MICRODIAGNOSTIC CODE
19767 I*****
19768
19769 .END

```

[illegible]

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
13851	6404	COUNTER534G	14359	7147	CR11-537A	14311	7137	CR3-537A	14335	7143	CR7-537A
9574	7035	CSP16XORFLTTOIRS	9568	7031	CSP16XORBERTOIPS	14352	5372	CSP16A21A	9627	6222	CUA372A
9678	5555	CUA372B	18650	7257	CUATOD	18643	7261	D0500A	18529	7237	D1106A
18514	7235	D1512A	18590	7245	DATISERVICETOD	18595	7246	DATOSERVICETOD	18788	7270	DBUFINTOIR
18788	7274	DBUFINTOIRS	17958	4415	DELAY768C	18782	7267	DINTOIR	8359	7363	DINTOIRISO
18782	7273	DINTOIRS	8408	4563	DINTOIRAB80	18410	7232	DISP002	18416	7223	DISP003
18422	7224	DISP004	18428	7225	DISP005	18434	7226	DISP006	18404	7221	DISPLAY
10718	4531	DNONZPRO410	11282	6153	DOFCCS03A	15611	4177	DOFCCS04A	10408	5440	DOIT376A
12009	7072	DOIT506A	14785	5313	DOIT610A1	14838	5321	DOIT610A2	14898	5327	DOIT610B1
14953	5336	DOIT610B2	15016	5344	DOIT610C1	15087	5352	DOIT610C2	15124	5357	DOIT610D1
15163	5363	DOIT610D2	7613	4163	DOPA133A1	7725	4173	DOPA134A1	7838	4203	DOPA135A1
7944	4212	DOPA136A1	9926	5225	DOWRITE374A1	9946	5230	DOWRITE374A2	9999	5234	DOWRITE374B1
10019	5217	DOWRITE374B2	10072	5243	DOWRITE374C1	10092	5247	DOWRITE374C2	10145	5253	DOWRITE374D1
10165	5256	DOWRITE374D2	10218	5262	DOWRITE374E1	10239	5265	DOWRITE374E2	10291	5271	DOWRITE374F1
10311	5274	DOWRITE374F2	14365	7150	DR11-537A	14387	7140	DR3-537A	14341	7144	DR7-537A
18558	7243	DTOIRB	17836	4411	DM11L762A	18852	7248	DZERO	10728	6833	DZEP0410
18538	7240	D[05-00]	18524	7236	D[11-06]	18811	7234	D[15-12]	12452	6270	ENFLAGS10D
18230	4431	EOP001	18237	6452	EOP002	18263	6810	EOP003	18250	6811	EOP004
18256	7211	EOP005	18265	6776	EOP006	18275	6812	EOP007	18282	7212	EOP010
14371	7151	ER3-537A	2827	6543	ERROR010	18426	5615	ERROR621A	15803	6567	ERROR624A
8853	6003	EXPEC361B	9969	6014	EXPEC362B	9167	6033	EXPEC365A	9236	6041	EXPEC366A
9295	6046	EXPEC366C	9914	5222	EXPEC374A1	9987	5237	EXPEC374B1	10060	5557	EXPEC374C1
10133	5567	EXPEC374D1	10206	5515	EXPEC374E1	10270	5227	EXPEC374F1	10785	7363	EXPEC410A
10822	6110	EXPEC410B	10851	6113	EXPEC410C	10886	6117	EXPEC410D	10918	6122	EXPEC410E
11108	6136	EXPEC500C	11171	6143	EXPEC500E	11268	6150	EXPEC503A	11414	6163	EXPEC503E
11599	6175	EXPEC504A	11723	6206	EXPEC504E	11860	6215	EXPEC505A	11989	6612	EXPEC506A
12151	6602	EXPEC507A	13893	6407	EXPEC535A	13934	6463	EXPEC535B	13986	6420	EXPEC536A
14026	6424	EXPEC536B	14116	6434	EXPEC536D	15802	5408	EXPEC621E	15528	5407	EXPEC621F
15688	5421	EXPEC623	15846	5430	EXPEC624C	15939	7166	EXPEC701A	15988	6455	EXPEC701C
16052	5437	EXPEC702A	16096	6460	EXPEC702B	16243	6465	EXPEC710C	16272	6467	EXPEC710D
16381	6473	EXPEC711B	16411	6475	EXPEC711C	16562	6501	EXPEC712D	16648	5452	EXPEC713B
16697	6504	EXPEC713D	17156	4336	EXPEC730B	18013	4417	EXPEC762E	18069	4422	EXPEC763A
17953	4414	FILL762C	10526	7046	FIRST375A	11056	6827	FIXPAT500	12446	6267	FLAGS10D
12571	7075	FLAGFP802	12577	7076	FLAGFP803	12582	7077	FLAGFP804	12566	7073	FLAGFP85EQL0D
18619	7252	FLAGFP870D	14382	7152	FR3-537A	11992	6231	FUDGEPS8506A	12177	6244	FUDGEPS8507A
18674	4314	GENADR712A	5954	4010	GETALT715A1	18535	5410	GETCUA621F	15695	5422	GETCUA623
15852	5431	GETCUA624C	17109	4332	GETD0UF730A	17335	4353	GETD0UF731A	8118	4222	GETDC320A
8176	4227	GETDC320C	8237	4234	GETDC320E	14784	5314	GETIT610A1	14844	5322	GETIT610A2
14907	5330	GETIT610B1	14962	5337	GETIT610B2	15028	5345	GETIT610C1	15076	5353	GETIT610C2
15133	5360	GETIT610D1	15172	5364	GETIT610D2	17189	4340	GETIT730B	17265	4347	GETIT730D
17376	4357	GETIT731B	17853	7210	GETIT7762A	15324	5371	GETJAM620C	15508	5406	GETJAM621E
15665	5420	GETJAM622C	18662	7260	GETM8KPPROCDAT	5881	4690	GETONES114A	18657	7373	GETPROCDAT
9934	5726	GETTEM374A1	9954	5231	GETTEM374B1	10007	5235	GETTEM374B2	10027	5240	GETTEM374B2
10880	5244	GETTEM374C1	10100	5250	GETTEM374C2	10283	5254	GETTEM374D1	10173	5257	GETTEM374D2
10246	5263	GETTEM374E1	10246	5266	GETTEM374F1	10289	5272	GETTEM374F1	10319	5278	GETTEM374F2
5894	4004	GETZER0ES114A	3056	5003	GOBUT012A	3077	5004	GOBUT012B	3097	5005	GOBUT012C
3116	5006	GOBUT012D	3135	5007	GOBUT012E	3155	5010	GOBUT012F	3177	5011	GOBUT012G
3220	5013	GOBUT013A	3242	5014	GOBUT013B	3261	5016	GOBUT013C	3280	5016	GOBUT013D
3299	5017	GOBUT013E	3320	5020	GOBUT013F	3339	5021	GOBUT013G	3382	5023	GOBUT014A
14803	5024	GOBUT014B	3422	5025	GOBUT014C	3444	5026	GOBUT014D	3488	5030	GOBUT015A
3508	5031	GOBUT015B	3527	5032	GOBUT015C	3546	5034	GOBUT016A	3613	5036	GOBUT017A
3635	5037	GOBUT017B	3677	5041	GOBUT020A	3697	5042	GOBUT020B	3719	5043	GOBUT020C

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
3762	5045	GOBUT021A	3783	5046	GOBUT021B	3805	5047	GOBUT021C	3848	5051	GOBUT022A
3892	5053	GOBUT023A	3936	5055	GOBUT024A	3957	5056	GOBUT024B	4001	5060	GOBUT025A
4045	5062	GOBUT026A	4089	5064	GOBUT027A	4133	5066	GOBUT030A	4155	5067	GOBUT030B
4174	5070	GOBUT030C	4197	5071	GOBUT030D	4240	5073	GOBUT031A	4261	5074	GOBUT031B
4304	5076	GOBUT032A	4325	5077	GOBUT032B	4376	5081	GOBUT033A	4414	5103	GOBUT034A
4457	5105	GOBUT035A	4502	5107	GOBUT036A	4546	5111	GOBUT037A	4589	5113	GOBUT040A
4611	5114	GOBUT040B	4654	5116	GOBUT041A	4675	5117	GOBUT041B	4718	5121	GOBUT042A
4739	5122	GOBUT042B	4783	5124	GOBUT043A	4804	5125	GOBUT043B	4846	5127	GOBUT044A
4891	5131	GOBUT045A	4935	5133	GOBUT046A	4959	5134	GOBUT046B	5004	5136	GOBUT047A
5027	5137	GOBUT047B	5047	5140	GOBUT047C	5093	5142	GOBUT050A	5122	5144	GOBUT050B
5626	5210	GOBUT105A1	5674	5213	GOBUT105B1	5899	4005	GOBUT114A	5913	4006	GOBUT114A2
5959	4011	GOBUT115A1	5991	4013	GOBUT115A2	6008	4014	GOBUT115A3	5975	4012	GOBUT115A4
6036	4016	GOBUT115B1	6068	4020	GOBUT115B2	6085	4021	GOBUT115B3	6052	4017	GOBUT115B4
6119	4023	GOBUT115C1	6134	4024	GOBUT115C2	6181	4025	GOBUT115C3	6178	4027	GOBUT115D1
6194	4030	GOBUT115D2	6211	4031	GOBUT115D3	6250	4033	GOBUT116A1	6264	4034	GOBUT116A2
6279	4035	GOBUT116A3	6296	4036	GOBUT116A4	6312	4037	GOBUT116A5	6340	4041	GOBUT116B
6374	4043	GOBUT116C1	6389	4044	GOBUT116C2	6404	4045	GOBUT116C3	6420	4046	GOBUT116C4
6436	4047	GOBUT116C5	6466	4051	GOBUT116D	6510	4053	GOBUT117A1	6525	4054	GOBUT117A2
6542	4055	GOBUT117A3	6557	4056	GOBUT117A4	6585	4060	GOBUT117B1	6600	4061	GOBUT117B2
6617	4062	GOBUT117B3	6647	4064	GOBUT117C1	6662	4065	GOBUT117C2	6700	4067	GOBUT120A1
6715	4070	GOBUT120A2	6732	4071	GOBUT120A3	6760	4073	GOBUT120B1	6775	4074	GOBUT120B2
6792	4075	GOBUT120B3	6807	4076	GOBUT120B4	6847	4103	GOBUT121A1	6862	4104	GOBUT121A2
6878	4105	GOBUT121A3	6893	4106	GOBUT121A4	6921	4110	GOBUT121B1	6936	4111	GOBUT121B2
6953	4112	GOBUT121B3	6968	4113	GOBUT121B4	7008	4115	GOBUT121C1	7017	4116	GOBUT121C2
7037	4117	GOBUT121C3	7048	4120	GOBUT121C4	7076	4122	GOBUT121D1	7091	4123	GOBUT121D2
7108	4124	GOBUT121D3	7123	4125	GOBUT121D4	7163	4127	GOBUT122A1	7179	4130	GOBUT122A2
7194	4131	GOBUT122A3	7215	4133	GOBUT122A4	7306	4136	GOBUT130A1	7321	4137	GOBUT130A2
7357	4142	GOBUT130B1	7372	4143	GOBUT130B2	7415	4146	GOBUT131A1	7430	4147	GOBUT131A2
7466	4152	GOBUT131B1	7481	4153	GOBUT131B2	7518	4155	GOBUT132A1	7534	4156	GOBUT132A2
7563	4160	GOBUT132B1	7579	4161	GOBUT132B2	7638	4165	GOBUT133A2	7685	4170	GOBUT133B2
7750	4175	GOBUT134A2	7798	4200	GOBUT134B2	7863	4205	GOBUT135A2	7911	4210	GOBUT135B2
7869	4214	GOBUT136A2	8017	4217	GOBUT136B2	8125	4223	GOBUT1320A	8146	4224	GOBUT1320B
8184	4230	GOBUT1320C	8204	4231	GOBUT1320D	8244	4235	GOBUT1320E	8263	4236	GOBUT1320F
8429	4243	GOBUT1350A	8456	4245	GOBUT1350B	8484	4247	GOBUT1350C	8512	4251	GOBUT1350D
8581	4256	GOBUT1351A	8603	4260	GOBUT1351B	8628	4262	GOBUT1351C	8647	4264	GOBUT1351D
8680	4266	GOBUT1352A	8704	4270	GOBUT1352B	8726	4272	GOBUT1352C	8748	4274	GOBUT1352D
8880	6005	GOBUT1361D	8902	6006	GOBUT1361E	8949	6013	GOBUT1362A	8996	6016	GOBUT1362D
9017	6017	GOBUT1362E	9056	6023	GOBUT1363A	9076	6024	GOBUT1363B	9104	6026	GOBUT1364A
9131	6030	GOBUT1364B	9194	6035	GOBUT1365B	9263	6043	GOBUT1366B	9345	6053	GOBUT1367B
9301	6060	GOBUT1370A	9412	6061	GOBUT1370B	9482	6055	GOBUT1370C	9474	6066	GOBUT1370D
9522	6073	GOBUT1371A	9545	6074	GOBUT1371B	9821	6101	GOBUT1373B	10552	7064	GOBUT1375A
10575	5301	GOBUT1375B	10626	5303	GOBUT1376A	10866	6116	GOBUT1410C	10898	6121	GOBUT1410D
10930	6124	GOBUT1410E	11084	6135	GOBUT1500A	11121	6140	GOBUT1500C	11149	6142	GOBUT1500D
11184	6145	GOBUT1500E	11306	6155	GOBUT1503A	11325	6156	GOBUT1503B	11346	6157	GOBUT1503C
11393	6162	GOBUT1503D	11366	6160	GOBUT1503DA	11440	6165	GOBUT1503F	11461	6166	GOBUT1503G
11482	6167	GOBUT1503H	11502	6170	GOBUT1503J	11520	6171	GOBUT1503J	11539	6172	GOBUT1503K
11636	6201	GOBUT1504A	11654	6202	GOBUT1504B	11675	6203	GOBUT1504C	11702	6205	GOBUT1504D
11749	6210	GOBUT1504F	11770	6211	GOBUT1504H	11791	6212	GOBUT1504H	11811	6213	GOBUT1504I
11899	6221	GOBUT1505B	11919	6223	GOBUT1505C	12022	7071	GOBUT1506A	12041	6233	GOBUT1506B
12062	6234	GOBUT1506C	12083	6235	GOBUT1506D	12112	6237	GOBUT1506E	12189	6246	GOBUT1507A
12208	6247	GOBUT1507B	12229	6250	GOBUT1507C	12250	6251	GOBUT1507D	12273	6253	GOBUT1507E
12328	6261	GOBUT1507F	12381	6264	GOBUT1510A	12403	6265	GOBUT1510B	12425	6266	GOBUT1510C

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
12462	6272	G0BUT510D	12483	6273	G0BUT510DA	12509	6275	G0BUT510E	12538	6277	G0BUT510F
13554	6351	G0BUT533B	13595	6355	G0BUT533B	13636	6360	G0BUT534A	13671	6363	G0BUT534B
13711	6366	G0BUT534C	13745	6371	G0BUT534D	13791	6375	G0BUT534E	13831	6403	G0BUT534F
13856	6405	G0BUT534G	13912	6412	G0BUT535A	13952	6416	G0BUT535B	14005	6423	G0BUT536A
14046	6427	G0BUT536B	14089	6432	G0BUT536C	14135	6437	G0BUT536D	14182	6443	G0BUT536E
14222	6447	G0BUT536F	14428	4303	G0BUT537A	14275	5586	G0BUT620A	14293	5367	G0BUT620B
15447	5402	G0BUT621B	15465	5403	G0BUT621C	15483	5404	G0BUT621D	15557	5411	G0BUT621E
15586	7160	G0BUT621H	15623	5415	G0BUT622A	15646	5417	G0BUT622B	15871	5432	G0BUT624D
15969	5434	G0BUT701B	16018	8435	G0BUT7301D	16037	6500	G0BUT712C	16671	6503	G0BUT713C
17127	4335	G0BUT730A	17175	4341	G0BUT730B	17198	4342	G0BUT730C	17219	4443	G0BUT730C1
17270	4350	G0BUT730D	17293	4351	G0BUT730E	17347	4358	G0BUT731A	17382	4360	G0BUT731B
17414	4362	G0BUT731C	17452	4366	G0BUT731D	17474	4367	G0BUT731E	17572	4776	G0BUT740A
17605	4766	G0BUT740B	17633	4375	G0BUT740C	17687	4767	G0BUT740D	17752	4405	G0BUT761A
17773	4406	G0BUT761B	17793	4407	G0BUT761C	17932	4762	G0BUT762B	17987	4416	G0BUT762D
18038	4421	G0BUT762F	18115	4425	G0BUT763B	18136	4426	G0BUT763C	18155	4427	G0BUT763D
10773	7361	G0FOR410	18019	4420	G0GET703E	10832	6112	G0PUT410B	11300	6154	G0PUT503A
11421	6164	G0PUT503E	11630	6200	G0PUT504A	11730	6207	G0PUT504E	11879	6220	G0PUT505A
5173	5147	G0TEST101A	5263	5156	G0TEST102A	5289	5160	G0TEST102B	5316	5162	G0TEST102C
5342	5164	G0TEST102D	5409	5173	G0TEST103A	5435	5178	G0TEST103B	5462	5177	G0TEST103C
5488	5201	G0TEST103D	5528	5203	G0TEST104A	5555	5205	G0TEST104B	5606	5207	G0TEST105A
5654	5212	G0TEST105B	5702	5215	G0TEST105C	5729	5217	G0TEST105D	5750	5220	G0TEST105E
7619	4164	G0TEST133A1	7669	4167	G0TEST133B1	7731	4174	G0TEST134A1	7781	4177	G0TEST134B1
7844	4204	G0TEST135A1	7894	4207	G0TEST135B1	7980	4213	G0TEST136A1	8000	4216	G0TEST136B1
9967	5232	G0TEST374A2	10040	5241	G0TEST374B2	10113	5251	G0TEST374C2	10186	5260	G0TEST374D2
10259	5267	G0TEST374E2	10332	5276	G0TEST374F2	12761	6550	G0TEST511A1	12772	7107	G0TEST511A2
12783	7111	G0TEST511A3	12794	7113	G0TEST511A4	12817	6554	G0TEST511B1	12828	7115	G0TEST511B2
12839	7117	G0TEST511B3	12850	7121	G0TEST511B4	13049	6311	G0TEST512A1	13060	6312	G0TEST512A2
13100	6315	G0TEST512B1	13115	6316	G0TEST512B2	13189	6321	G0TEST512C1	13164	6322	G0TEST512C2
13204	6325	G0TEST512D1	13219	6326	G0TEST512D2	13283	6331	G0TEST512E1	13268	6332	G0TEST512E2
13346	6335	G0TEST520A	13374	6337	G0TEST520B	13393	6341	G0TEST520C	13431	6343	G0TEST520D
13460	6348	G0TEST520E	14487	4564	G0TEST551A	14511	4560	G0TEST551B	14528	7157	G0TEST551B1
14552	5626	G0TEST551C	15949	7171	G0TEST701A	15988	4457	G0TEST701C	16074	5444	G0TEST702A
16113	4463	G0TEST702B	16219	6464	G0TEST710B	16248	4466	G0TEST710C	16284	6471	G0TEST710D
16301	6472	G0TEST710E	16387	6474	G0TEST711B	16416	6476	G0TEST711C	16435	6477	G0TEST711D
16519	5450	G0TEST711B	16567	6502	G0TEST712D	16583	5483	G0TEST713B	16702	6505	G0TEST713D
16796	4506	G0TEST720B	16820	4317	G0TEST720E	16901	4321	G0TEST721B	16925	4322	G0TEST721C
17012	4325	G0TEST722B	17036	4326	G0TEST722C	14388	7183	GR3-537A	14394	7154	HR3-537A
13535	6666	INIT533A	13576	6352	INIT535B	13687	6586	INIT535A	13980	6654	INIT536A
14110	6433	INIT536D	13542	6347	INIT535B	13582	6353	INIT535B	13623	6664	INIT534A
13658	6361	INIT534B	13698	6662	INIT534C	13732	6367	INIT534D	13778	6373	INIT534E
13817	6401	INIT534F	13899	6410	INIT535B	13940	6414	INIT535B	13993	6421	INIT536A
14033	6425	INIT536B	14075	6652	INIT536C	14123	6438	INIT536D	14168	6441	INIT536E
14209	6445	INIT536F	2931	6532	INITIALIZE01	2937	7081	INITIALIZE03	2942	7003	INITIALIZE04
2948	7004	INITIALIZE05	2953	7005	INITIALIZE06	2958	7086	INITIALIZE07	2965	4101	INITIALIZE10
2977	4102	INITIALIZE11	2988	4001	INITIALIZE12	14280	7131	INIT5R537A	18367	7215	INSERT02
18378	7217	INSERT03	18384	7220	INSERT04	18562	7214	INSERT0PREVNO	18373	7216	INSERTREYNB
10802	6107	INT01P410A	18643	7256	JANTOD	18623	4277	JANUPP001	18830	4757	JANUPP002B
18835	4434	JAMUPP002C	18841	7213	JANUPP002D	18847	4786	JANUPP003	18854	4746	JANUPP004
18861	4747	JAMUPP005	18867	4435	JAMUPP006	18872	4436	JANUPP007	18877	4437	JANUPP010
18882	4440	JAMUPP011	18176	4763	KILL764A	12933	7114	KTRCDBST01	12938	7123	KTRCDBST02
12944	7124	KTRCDBST03	12951	7125	KTRCDBST04	12958	7636	KTRCDBST04B	12967	7335	KTRCDBST05
12973	7337	KTRCDBST06	12981	7127	KTRCDBST07	12987	7130	KTRCDBST08	5252	5154	LOAD00-102A

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
5232	5530	LOAD01-102A	14758	5306	LOAD01-610A1	14819	5316	LOAD01-610A2	14882	5324	LOAD01-610B1
14937	5333	LOAD01-610B2	15000	5341	LOAD01-610C1	15081	5347	LOAD01-610C2	15108	5510	LOAD01-610D1
2809	4100	LOAD010	2854	5001	LOAD011	3049	5002	LOAD012A	3213	5744	LOAD013A
3376	5742	LOAD014A	3427	4300	LOAD01537A	3481	5740	LOAD015A	3563	5736	LOAD016A
3605	5732	LOAD017A	5237	5191	LOAD02-102A	3670	5734	LOAD020A	3754	5720	LOAD021A
3841	5716	LOAD022A	3885	5714	LOAD023A	3929	5796	LOAD024A	3994	5704	LOAD025A
4038	5702	LOAD026A	4082	5700	LOAD027A	4125	5676	LOAD030A	4232	5672	LOAD031A
4297	5666	LOAD032A	4363	5664	LOAD033A	4406	5662	LOAD034A	4466	4277	LOAD03537A
4450	5660	LOAD035A	4494	5656	LOAD036A	4539	5654	LOAD037A	5242	5152	LOAD04-102A
4582	5650	LOAD040A	4646	5644	LOAD041A	4721	5640	LOAD042A	4775	5634	LOAD043A
4841	5632	LOAD044A	4886	5630	LOAD045A	4928	5626	LOAD046A	4997	5550	LOAD047A
8544	4567	LOAD05-351	14764	4507	LOAD05-610A1	15045	5346	LOAD05-610C2	5086	5552	LOAD050A
5116	5143	LOAD050B	8554	4254	LOAD06-351	14769	5310	LOAD06-610A1	14813	5315	LOAD06-610A2
5393	5170	LOAD07-103A	14832	5332	LOAD07-610B2	5247	5153	LOAD10-102A	5388	5167	LOAD13-103A
14259	4276	LOAD14537A	5383	5166	LOAD15-103A	5277	5346	LOAD16-103A	14252	4574	LOAD16537A
5398	5171	LOAD17-103A	9633	6703	LOAD372A	9884	5803	LOAD372B	11854	6620	LOAD505A
16170	4307	LOADADRY10A	15944	7170	LOADBA701A	15993	6486	LOADBA701C	16069	5443	LOADBA702A
16107	6462	LOADBA702B	5168	5520	LOADD105A	5258	5185	LOADD102A	5284	5157	LOADD102B
5311	5161	LOADD102C	5337	5163	LOADD105B	5404	5172	LOADD103A	5430	5174	LOADD103B
5457	5176	LOADD103C	5483	5200	LOADD103D	5523	5540	LOADD104A	5580	5204	LOADD104B
17096	4330	LOADDA7A730D	17240	4343	LOADDA7A730D	12015	6104	LOADDC5506A	12184	6245	LOADDC5507A
11276	6152	LOADFCC503A	11605	6176	LOADFCC504A	11246	6414	LOADFLAG503A	11595	6622	LOADFLAG504A
11255	6147	LOADFP5803A	11593	6174	LOADFP5804A	18704	7264	LOADFP58CC	18710	7265	LOADFP58CC02
8384	7357	LOADHI350	8393	7024	LOADHIA350	14476	4305	LOADIR551A	11017	6132	LOADIR500
11386	4161	LOADIR603D	11695	6204	LOADIR504B	14782	5532	LOADIR610A1	14876	5534	LOADIR610B1
14926	5331	LOADIR610B2	14994	5544	LOADIR610C1	16743	7172	LOADIR720A	17091	4754	LOADIR730A
17435	4363	LOADIR731D	17551	4765	LOADIR740A	17886	4372	LOADIR740B	17647	4400	LOADIR740D
12375	6263	LOADPRI0510A	8841	6001	LOADRES361A	9337	6011	LOADRES362A	9161	6032	LOADRES365A
9224	6037	LOADRES366A	9332	6051	LOADRES367A	9375	6055	LOADRES370A	9505	6070	LOADRES371A
9907	6103	LOADRET737A	16164	5472	LOADRET711A	16335	5470	LOADRET711A	16467	5466	LOADRET712A
16602	5464	LOADRET713A	16737	5462	LOADRET722A	16847	4320	LOADRET721A	16958	5456	LOADRET722A
5600	5542	LOADSR105A	5649	5211	LOADSR105B	5697	5214	LOADSR105C	5733	5216	LOADSR105D
8848	6002	LOADSR361A	9340	6052	LOADSR367A	14162	6650	LOADSR536E	14203	6444	LOADSR536F
8350	4241	LOADSR365D	11999	7067	LOADUCON506A	15369	5374	LOADSRK621A	15747	5425	LOADSRK624A
13772	6660	LOADNTR534E	13812	6400	LOADNTR534F	15988	5480	LOADFLG621A	15731	5562	LOADING624A
9613	6540	LOADP372A	17252	4345	WANGLED730D	17446	4365	WANGLED731D	11270	6151	NASK503A
15317	5370	NASK620C	15933	5476	NASK701A	16047	8474	NASK702A	17842	7202	NASK762A
12686	6232	NF5801	12692	6302	NF5801	12699	6303	NF5803	12706	6304	NF5804
12715	6552	NF5805	12725	6553	NF5806	12738	6545	NF58EXPECO	12742	6547	NF58EXPECI
18648	7261	NKPPROCDAY	14291	6643	NKPPROCDAY	2767	7303	NEXT007	2833	5245	NEXT010
16193	4445	NEXT7110A	16355	4446	NEXT7111A	16492	4447	NEXT7112A	16621	7365	NEXT7113A
16769	7364	NEXT7220A	16875	7362	NEXT7221A	16985	4524	NEXT7222A	17891	4467	NEXTD762A
8376	7023	NEXTPAT350	10737	7065	NEXTPAT410	11039	6377	NEXTPAT500	9536	4253	NEXTPAT350
18633	7254	ODDJAMTOD	7659	4166	OPA133B1	7712	4512	OPA134A1	7771	4176	OPA134B1
7825	4514	OPA135A1	7884	4206	OPA135B1	7990	4215	OPA136B1	7607	4510	OPB

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
12623	7105	P8SEQL0D04	18626	7253	P8TOD	18082	4423	READYECT763A	8530	4703	RESETIR350
8435	7022	RESETRIR350A	8461	7026	RESETRIR350B	8489	7027	RESETRIR350C	18679	7261	RESETPROCADAT
18765	7271	RESETRUCOMP	18795	7275	RESETRUCOMP5	8769	4718	RESTORE01	8776	7030	RESTORE02
8782	7032	RESTORE03	8788	7033	RESTORE04	9745	8662	RETURN373A	3185	5012	SCOPE012
1347	5022	SCOPE013	3452	5027	SCOPE014	3533	5033	SCOPE018	3578	5035	SCOPE016
1642	5040	SCOPE017	3726	5044	SCOPE020	3812	5050	SCOPE021	3886	5082	SCOPE022
3899	5054	SCOPE023	3965	5057	SCOPE024	4008	5061	SCOPE025	4053	5063	SCOPE026
4095	5065	SCOPE027	4204	5072	SCOPE030	4288	5078	SCOPE031	4333	5100	SCOPE032
4378	5102	SCOPE033	4421	5104	SCOPE034	4468	5106	SCOPE035	4510	5110	SCOPE036
4554	5112	SCOPE037	4618	5115	SCOPE040	4688	5120	SCOPE041	4747	5123	SCOPE042
4811	5126	SCOPE043	4855	5130	SCOPE044	4900	5132	SCOPE045	4967	5138	SCOPE046
5056	5141	SCOPE047	5129	5145	SCOPE050	5160	5150	SCOPE101	5350	5165	SCOPE102
5496	5202	SCOPE103	5563	5206	SCOPE104	5759	5221	SCOPE105	5920	4007	SCOPE114A
6092	4022	SCOPE115B	6217	4032	SCOPE118D	6348	4042	SCOPE116C	6474	4052	SCOPE116D
6670	4066	SCOPE117C	6813	4077	SCOPE120B	6976	4114	SCOPE121C	7129	4126	SCOPE121D
7223	4134	SCOPE122A	7380	4144	SCOPE130B	7489	4194	SCOPE131B	7587	4162	SCOPE132B
7693	4171	SCOPE133B	7806	4201	SCOPE134B	7919	4211	SCOPE135B	8025	4220	SCOPE136B
8270	4237	SCOPE1320	8520	4252	SCOPE1350	8655	4268	SCOPE1351	8756	4275	SCOPE1352
9553	6075	SCOPE1371	9700	5501	SCOPE1372B	9828	6102	SCOPE1373	9975	5233	SCOPE1374A
10048	5242	SCOPE1374B	10121	5252	SCOPE1374C	10194	5261	SCOPE1374D	10267	5270	SCOPE1374E
10340	5277	SCOPE1374F	10633	5304	SCOPE1376	10937	6126	SCOPE1410	11192	6146	SCOPE1300F
11550	6173	SCOPE1303	11820	6214	SCOPE1504	11955	6284	SCOPE1505	12121	6240	SCOPE1506
12338	6262	SCOPE1507	12549	6300	SCOPE1510	12800	6301	SCOPE1511A	12856	6305	SCOPE1511B
13066	6313	SCOPE1512A	13170	6323	SCOPE1512C	13275	6333	SCOPE1512E	13469	6346	SCOPE1520
13601	6356	SCOPE1533B	13677	6364	SCOPE1534B	13752	6372	SCOPE1534D	13866	6406	SCOPE1534G
13959	6417	SCOPE1535B	14052	6430	SCOPE1538B	14141	6440	SCOPE1536D	14228	6450	SCOPE1536F
14434	6404	SCOPE1537A	14559	6305	SCOPE1551	14852	5323	SCOPE1610A	14970	5340	SCOPE1610B
15084	6354	SCOPE1610C	15180	6365	SCOPE1610D	15702	5493	SCOPE1710	15884	5224	SCOPE1624
16027	5436	SCOPE1701	16120	5433	SCOPE1702	16310	5445	SCOPE1710	16443	5446	SCOPE1711
16576	5447	SCOPE1712	16711	5451	SCOPE1713	16934	5484	SCOPE1721	17045	4327	SCOPE1722
17299	4352	SCOPE1730	17481	4370	SCOPE1731	17875	4462	SCOPE1740	17801	4410	SCOPE1761
18161	4430	SCOPE1763	18533	7061	SECONDR725A	18605	7250	SERVICETOD	18748	7174	SETADR720A
18853	7173	SETADR721A	18964	4470	SETADR722A	18361	5373	SETBRK621A	18738	5424	SETBRK624A
14780	5312	SETBUSA610A1	14830	5320	SETBUSA610A2	14893	5326	SETBUSA610B1	14948	5335	SETBUSA610B2
15011	5343	SETBUSA610C1	15062	5351	SETBUSA610C2	15119	5358	SETBUSA610D1	15158	5362	SETBUSA610D2
10673	6540	SETBYTE6410	10680	7060	SETBYTE6410	10688	7066	SETBYTE6410	10691	6000	SETBYTE6410
10697	6105	SETBYTE6410	12320	6260	SETDCIN507F	7210	4132	SETD1224A	8111	4221	SETD320A
8169	4226	SETD320C	8230	4233	SETD320E	15760	6306	SETD624A	9622	6076	SETD327A
8471	7000	SETDDC372B	8932	4010	SETDDC382A	9040	6020	SETDDC363A	9099	6025	SETDDC364A
9126	6027	SETDDC364B	9285	6044	SETDDC365C	9436	6082	SETDDC370C	14246	6651	SETDIT537A
10976	6676	SETFETCH8500	10982	6126	SETFETCH8500	10987	6137	SETFETCH8500	10992	6130	SETFETCH8500
10998	6131	SETFETCH8500	11867	6216	SETFLAG605A	12457	6271	SETFLAG510D	15382	5377	SETFLG621A
14765	6453	SETFLG624A	9751	7034	SETIR373A	10602	5302	SETIR376A	13033	6864	SETIR512A1
13088	6536	SETIR512B1	13137	6317	SETIR512C1	13192	6425	SETIR512D1	13241	6327	SETIR512E1
3039	5760	SETIRA	16175	4312	SETJAM720A	16783	7175	SETJAM720A	16858	7200	SETJAM721A
16969	4323	SETJAM722A	16062	5442	SETKT702A	16401	6461	SETKT702B	16057	5441	SETLED702A
8105	4700	SETONE320A	8164	4225	SETONE320C	8234	4232	SETONE320E	12370	6606	SETONE5510A
17007	4463	SETPR6-762A	17912	4413	SETPRI762A	17986	7286	SETPRI762C	12301	6256	SETPR507F
13039	6310	SETPSFLAG512A1	13094	6314	SETPSFLAG512B1	13143	6320	SETPSFLAG512C1	13198	6324	SETPSFLAG512D1
13247	6330	SETPSFLAG512E1	8834	6570	SETRES364A	8884	6607	SETRES362A	9153	6031	SETRES365A
9216	6036	SETRES366A	9324	6050	SETRES367A	9367	6054	SETRES370A	9497	6067	SETRES371A
14285	7135	SETRESA537A	14377	7370	SETRESB637A	15358	5375	SETRET621A	15784	5426	SETRET624A

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
10519	5506	SETSP375A	9757	7037	SETSR373A	12310	6287	SETSR507F	14481	4306	SETSR551A
14506	4565	SETSR551B	14546	4561	SETSR551C	15393	5401	SETSR621A	15771	6454	SETSR624A
11965	6225	SETUCON506A	13001	6555	SETUPS12A	13007	6307	SETUPS12B	13336	6334	SETUPS20A
13367	6336	SETUPS20B	13396	6340	SETUPS20C	13474	6342	SETUPS20D	13452	6344	SETUPS20E
5862	7015	SETUPCSP00A	5852	7013	SETUPCSP05A	5857	7014	SETUPCSP07A	5847	7012	SETUPCSP12A
5842	7011	SETUPCSP14A	5837	7010	SETUPCSP16A	5832	7002	SETUPCSP17A	5826	5843	SETUPCSP17A
15197	7156	SETUPP8CCADC	15202	7161	SETUPP8CCADC02	17245	4344	SETZER730F	10415	5746	SFDPTOSR
10428	7047	SFDPTOSRA	10423	7040	SFDPTOSRBA	10434	7050	SFDPTOSRBB	10440	7051	SFDPTOSRBC
10446	7052	SFDPTOSRBD	10452	7053	SFDPTOSRBE	10488	7054	SFDPTOSRBF	10464	7055	SFDPTOSRGG
10470	7046	SFDPTOSRH	10476	7057	SFDPTOSRI	9048	6021	SHIF1363A	9230	6040	SHIF1366A
9290	6045	SHIF1366C	9381	6056	SHIF1370A	9442	6063	SHIF1370C	9511	6071	SHIF1371A
19747	7266	SRINTOIR	18777	7272	SRINTOIRS	8343	4240	START350	2816	7000	SUBR010
9639	6077	SUBRA372A	9689	5223	SUBRA372B	13482	7132	SUCBRTST01	13487	7132	SUCBRTST02
13497	7133	SUCBRTST03	2633	4000	TEST001	2684	6062	TEST002	2673	6631	TEST003
2692	5525	TEST004	2711	5146	TEST005	2730	4474	TEST006	2749	4377	TEST007
2802	4775	TEST010	2847	5777	TEST011	3031	5775	TEST012A	3069	5773	TEST012B
3088	5771	TEST012C	3108	5767	TEST012D	3127	5768	TEST012E	3147	5764	TEST012F
3169	5763	TEST012G	3205	5761	TEST013A	3233	5757	TEST013B	3253	5755	TEST013C
3277	5753	TEST013D	3291	5751	TEST013E	3312	5750	TEST013F	3331	5747	TEST013G
3369	5745	TEST014A	3394	5752	TEST014B	3414	5762	TEST014C	3436	5710	TEST014D
3473	5743	TEST015A	3500	5772	TEST015B	3519	5722	TEST015C	3555	5741	TEST016A
3598	5737	TEST017A	3626	5735	TEST017B	3662	5733	TEST020A	3689	5731	TEST020B
3711	5727	TEST020C	3746	5725	TEST021A	3775	5723	TEST021B	3797	5724	TEST021C
3833	5721	TEST022A	3877	5717	TEST023A	3920	5715	TEST024A	3949	5711	TEST024B
3986	5707	TEST025A	4029	5705	TEST026A	4074	5703	TEST027A	4117	5701	TEST030A
4147	5712	TEST030B	4166	5713	TEST030C	4189	5704	TEST030D	4224	5677	TEST031A
4253	5675	TEST031B	4288	5673	TEST032A	4317	5671	TEST032B	4354	5667	TEST033A
4398	5665	TEST034A	4442	5663	TEST035A	4486	5661	TEST036A	4531	5657	TEST037A
4574	5655	TEST040A	4602	5653	TEST040B	4638	5651	TEST041A	4667	5647	TEST041B
4703	5645	TEST042A	4731	5643	TEST042B	4767	5641	TEST043A	4796	5637	TEST043B
4833	5635	TEST044A	4878	5633	TEST045A	4920	5631	TEST046A	4950	5770	TEST046B
4988	5561	TEST047A	5019	5670	TEST047B	5039	5730	TEST047C	5077	5551	TEST050A
5108	5726	TEST050B	5161	5553	TEST101A	5225	5521	TEST102A	5277	5574	TEST102B
5301	5864	TEST102C	5330	5554	TEST102D	5370	5531	TEST103A	5423	5572	TEST103B
5449	5576	TEST103C	5476	5674	TEST103D	5516	5547	TEST104A	5543	5546	TEST104B
5592	5541	TEST105A	5619	5610	TEST105B	5642	5682	TEST105B	5666	5600	TEST105B1
5690	5642	TEST105C	5716	5636	TEST105D	5743	5620	TEST105E	5874	4607	TEST114A
5906	4603	TEST114A2	5939	4601	TEST115A1	5983	4616	TEST115A2	6000	4615	TEST115A3
5967	4604	TEST115A4	6021	4614	TEST115B1	6060	4613	TEST115B2	6077	4612	TEST115B3
6044	4602	TEST115B4	6104	4611	TEST115C1	6126	4617	TEST115C2	6143	4627	TEST115C3
6164	4626	TEST115D1	6186	4625	TEST115D2	6203	4624	TEST115D3	6235	4623	TEST116A1
6257	4621	TEST116A2	6271	4620	TEST116A3	6288	4637	TEST116A4	6305	4636	TEST116A5
6326	4606	TEST116B	6359	4635	TEST116C1	6381	4633	TEST116C2	6396	4632	TEST116C3
6413	4631	TEST116C4	6429	4630	TEST116C5	6482	4645	TEST116D	6498	4647	TEST117A1
6517	4644	TEST117A2	6534	4643	TEST117A3	6549	4642	TEST117A4	6570	4641	TEST117B1
6592	4640	TEST117B2	6609	4637	TEST117B3	6633	4656	TEST117C1	6655	4677	TEST117C2
6685	4667	TEST120A1	6707	4655	TEST120A2	6724	4654	TEST120A3	6745	4653	TEST120B1
4767	4652	TEST120B2	6784	4651	TEST120B3	6799	4650	TEST120B4	6832	4665	TEST121A1
6854	4676	TEST121A2	6871	4675	TEST121A3	6885	4674	TEST121A4	6906	4663	TEST121B1
6928	4662	TEST121B2	6945	4661	TEST121B3	6960	4660	TEST121B4	6987	4707	TEST121C1
7009	4605	TEST121C2	7025	4717	TEST121C3	7040	4716	TEST121C4	7061	4727	TEST121D1
7083	4726	TEST121D2	7100	4737	TEST121D3	7115	4736	TEST121D4	7148	4671	TEST122A1

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
7172	4570	TEST122A2	7188	4734	TEST122A3	7202	4735	TEST122A4	7283	4673	TEST130A1
7114	4550	TEST130A2	7334	4530	TEST130B1	7368	4632	TEST130B2	7392	4641	TEST131A1
7423	4544	TEST131A2	7443	4546	TEST131B2	7494	4780	TEST131B2	7502	4521	TEST132A1
7527	4516	TEST132A2	7547	4517	TEST132B1	7572	4472	TEST132B2	7600	4523	TEST133A1
7630	4500	TEST133A2	7652	4501	TEST133B1	7678	4502	TEST133B2	7705	4511	TEST134A1
7742	4503	TEST134A2	7764	4504	TEST134B1	7791	4505	TEST134B2	7818	4513	TEST135A1
7855	4506	TEST135A2	7877	4507	TEST135B1	7904	4477	TEST135B2	7931	4515	TEST136A1
7961	4476	TEST136A2	7983	4475	TEST136B1	8010	4473	TEST136B2	8098	4543	TEST1320A
8138	4552	TEST1320B	8157	4562	TEST1320C	8197	4517	TEST1320D	8217	4584	TEST1320E
8256	4556	TEST1320F	8337	4701	TEST1350	8417	4782	TEST1350A	8444	4753	TEST1350B
8471	4742	TEST1350C	8499	4743	TEST1350D	8569	4739	TEST1351A	8591	4722	TEST1351B
8612	4723	TEST1351C	8634	4712	TEST1351D	8670	4705	TEST1352A	8692	4713	TEST1352B
8713	4724	TEST1352C	8735	4725	TEST1352D	8827	6573	TEST1361A	8873	6771	TEST1361D
8894	6767	TEST1361E	8917	6765	TEST1362A	8962	6743	TEST1362B	8989	6757	TEST1362D
9010	6755	TEST1362E	9033	6753	TEST1363A	9069	6751	TEST1363B	9091	6747	TEST1364A
9119	6745	TEST1364B	9146	6743	TEST1365A	9187	6741	TEST1365B	9209	6737	TEST1366A
9256	6735	TEST1366B	9278	6733	TEST1366C	9317	6731	TEST1367A	9360	6725	TEST1370A
9405	6723	TEST1370B	9429	6721	TEST1370C	9467	6717	TEST1370D	9490	6715	TEST1371A
9538	6713	TEST1371B	9606	6571	TEST1372A	9658	6701	TEST1372B	9738	6541	TEST1373A
9814	6572	TEST1373B	9898	6563	TEST1374	9959	6604	TEST1374A2	10032	5606	TEST1374B2
10105	5524	TEST1374C2	10178	5500	TEST1374D2	10251	5502	TEST1374E2	10324	5504	TEST1374F2
10512	5517	TEST1375A	10563	5625	TEST1375B	10595	5513	TEST1376A	10619	5614	TEST1376A1
10666	5507	TEST1410	10790	6711	TEST1410A	10814	6704	TEST1410B	10844	6707	TEST1410C
10879	4727	TEST1410D	10911	6705	TEST1410E	10968	6561	TEST1500	11070	6673	TEST1500A
11100	6726	TEST1500C	11136	6746	TEST1500D	11164	6736	TEST1500E	11238	6677	TEST1503A
11318	6630	TEST1503B	11338	6632	TEST1503C	11379	6670	TEST1503D	11359	6601	TEST1503DA
11407	6700	TEST1503E	11433	6710	TEST1503F	11493	6720	TEST1503G	11474	6732	TEST1503H
11495	6742	TEST1503I	11513	6756	TEST1503J	11531	6446	TEST1503K	11578	6615	TEST1504A
11647	6730	TEST1504B	11667	6740	TEST1504C	11688	6750	TEST1504D	11716	6760	TEST1504E
11742	6672	TEST1504F	11762	6702	TEST1504G	11783	6706	TEST1504H	11804	6716	TEST1504I
11847	6623	TEST1505A	11891	6712	TEST1505B	11912	6722	TEST1505C	11952	6621	TEST1506A
12034	6754	TEST1506B	12054	6744	TEST1506C	12076	6734	TEST1506D	12099	6605	TEST1506E
12144	6613	TEST1507A	12201	6724	TEST1507B	12221	6611	TEST1507C	12243	6557	TEST1507D
12265	6674	TEST1507E	12286	6647	TEST1507F	12363	6603	TEST1510A	12396	6675	TEST1510B
12418	6616	TEST1510C	12438	6600	TEST1510D	12478	6617	TEST1510DA	12497	6604	TEST1510E
12524	6610	TEST1510F	12555	6607	TEST1511A	12587	7074	TEST1511A2	12778	7110	TEST1511A3
12789	7112	TEST1511A4	12812	6551	TEST1511B1	12823	7106	TEST1511B2	12834	7116	TEST1511B3
12845	7120	TEST1511B4	13026	6542	TEST1512A1	13085	6544	TEST1512A2	13081	6565	TEST1512B1
13108	6546	TEST1512B2	13130	6556	TEST1512B1	13187	6556	TEST1512C2	13185	6537	TEST1512D1
13212	6576	TEST1512D2	13234	6577	TEST1512E1	13261	6538	TEST1512E2	13322	6627	TEST1520A
13360	6764	TEST1520B	13388	6772	TEST1520C	13417	6762	TEST1520D	13445	6775	TEST1520E
13528	6625	TEST1533A	13569	6633	TEST1533B	13615	6647	TEST1534A	13650	6634	TEST1534B
13691	6665	TEST1534C	13725	6635	TEST1534D	13769	6663	TEST1534E	13805	6636	TEST1534F
13844	6752	TEST1534G	13879	6661	TEST1535A	13926	6637	TEST1535B	13973	6657	TEST1536A
14019	6640	TEST1536B	14067	6655	TEST1536C	14183	6641	TEST1536D	14154	6653	TEST1536E
14196	6642	TEST1536F	14413	4572	TEST1537A	14468	4575	TEST1551A	14498	4576	TEST1551B
14539	4566	TEST1551C	14744	5627	TEST1610A1	14805	5628	TEST1610A2	14869	5533	TEST1610B1
14910	5612	TEST1610B2	14987	5535	TEST1610C1	15085	5602	TEST1610C2	15101	5545	TEST1610D1
15145	5624	TEST1610D2	15262	5511	TEST1620A	15286	5483	TEST1620B	15309	5621	TEST1620C
15344	5617	TEST1621A	15440	5613	TEST1621B	15485	5611	TEST1621C	15476	5607	TEST1621D
15495	5605	TEST1621E	15521	5603	TEST1621F	15600	5746	TEST1621G	15572	5754	TEST1621H
15605	5601	TEST1622A	15634	5577	TEST1622B	15687	5575	TEST1622C	15680	5573	TEST1623

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
15724	5571	TEST624A	15817	5565	TEST624B	15839	5776	TEST624C	15864	5756	TEST624D
15925	5563	TEST701A	15962	5523	TEST701B	15981	6770	TEST701C	16011	5512	TEST701D
16040	5477	TEST702A	16089	6761	TEST702B	16157	5475	TEST710A	16212	6644	TEST710B
16236	6645	TEST710C	16265	6574	TEST710D	16294	6575	TEST710E	16328	5473	TEST711A
16374	6534	TEST711B	16404	6530	TEST711C	16437	6526	TEST711D	16460	5471	TEST712A
16512	4505	TEST712B	16530	6524	TEST712C	16559	6583	TEST712D	16594	5467	TEST713A
16640	5616	TEST713B	16664	6522	TEST713C	16689	6581	TEST713D	16730	5465	TEST720A
16788	4520	TEST720B	16813	4760	TEST720C	16840	4571	TEST721A	16894	4573	TEST721B
16918	4577	TEST721C	16951	5463	TEST722A	17004	4484	TEST722B	17029	4465	TEST722C
17084	4471	TEST730A	17149	4466	TEST730B	17188	4460	TEST730C	17211	4450	TEST730C1
17233	4461	TEST730D	17286	4462	TEST730E	17321	4785	TEST731A	17363	4457	TEST731B
17401	4456	TEST731C	17428	4455	TEST731D	17467	4454	TEST731E	17556	4744	TEST740A
17591	4453	TEST740B	17619	4452	TEST740C	17683	4451	TEST740D	17726	4745	TEST761A
17765	4771	TEST761B	17786	4774	TEST761C	17829	4773	TEST762A	17899	4553	TEST762A1
17924	4751	TEST762B	17946	4740	TEST762C	17980	4741	TEST762D	18006	4730	TEST762E
18031	4711	TEST762F	18062	4761	TEST763A	18108	4720	TEST763B	18128	4721	TEST763C
18144	4710	TEST763D	18192	6714	TESTA503A	18221	6766	TESTA504A	18270	6773	TESTD410
18288	6671	TESTD500	18331	6777	TESTINHBSP500	18339	6624	UCONS20A	18089	4424	TESTVECT763A
18400	5522	URR621A	18376	6255	URR624A	18399	6624	UCONS20A	18103	4733	VECTLOAD763A
18481	6774	VFY001	18380	4412	VFY002	18313	4255	VFY003	18321	4433	VFY004
18333	7375	VFY005	18340	7374	VFY006	18340	7374	VFY006	8364	4002	WRITEDF350
8371	7016	WRITEDF350	17861	4315	ZAPD762A	17122	4334	ZAPD762A	9919	5536	ZERO374A1
9941	5227	ZERO374A2	9992	5556	ZERO374B1	10014	5236	ZERO374B2	10065	5566	ZERO374C1
10087	5246	ZERO374C2	10138	5514	ZERO374D1	10160	5255	ZERO374D2	10211	5526	ZERO374E1
10233	5264	ZERO374E2	10284	5516	ZERO374F1	10306	5273	ZERO374F2	12504	6274	ZEROD510E
17734	4772	ZEROD8761A	12531	6276	ZEROD8761B	15824	5427	ZEROD8761B	17739	4403	ZEROIT761A
10386	7044	ZEROD8F	10365	7036	ZEROD8F02DF04	10370	7041	ZEROD8F04DF02	10392	7045	ZEROD8F
10375	7042	ZEROD8FDF	10381	7043	ZEROD8FDF	19252	7400	ZTARGET400	19254	7401	ZTARGET401
19256	7402	ZTARGET402	19258	7403	ZTARGET403	19260	7404	ZTARGET404	19262	7405	ZTARGET405
19264	7406	ZTARGET406	19266	7407	ZTARGET407	19268	7410	ZTARGET410	19270	7411	ZTARGET411
19272	7412	ZTARGET412	19274	7413	ZTARGET413	19276	7414	ZTARGET414	19278	7415	ZTARGET415
19280	7416	ZTARGET416	19282	7417	ZTARGET417	19284	7420	ZTARGET420	19286	7421	ZTARGET421
19288	7422	ZTARGET422	19290	7423	ZTARGET423	19292	7424	ZTARGET424	19294	7425	ZTARGET425
19296	7426	ZTARGET426	19298	7427	ZTARGET427	19300	7430	ZTARGET430	19302	7431	ZTARGET431
19304	7432	ZTARGET432	19306	7433	ZTARGET433	19308	7434	ZTARGET434	19310	7435	ZTARGET435
19312	7436	ZTARGET436	19314	7437	ZTARGET437	19316	7440	ZTARGET440	19318	7441	ZTARGET441
19320	7442	ZTARGET442	19322	7443	ZTARGET443	19324	7444	ZTARGET444	19326	7445	ZTARGET445
19328	7446	ZTARGET446	19330	7447	ZTARGET447	19332	7450	ZTARGET450	19334	7451	ZTARGET451
19336	7452	ZTARGET452	19338	7453	ZTARGET453	19340	7454	ZTARGET454	19342	7455	ZTARGET455
19344	7456	ZTARGET456	19346	7457	ZTARGET457	19348	7460	ZTARGET460	19350	7461	ZTARGET461
19352	7462	ZTARGET462	19354	7463	ZTARGET463	19356	7464	ZTARGET464	19358	7465	ZTARGET465
19360	7466	ZTARGET466	19362	7467	ZTARGET467	19364	7470	ZTARGET470	19366	7471	ZTARGET471
19368	7472	ZTARGET472	19370	7473	ZTARGET473	19372	7474	ZTARGET474	19374	7475	ZTARGET475
19376	7476	ZTARGET476	19378	7477	ZTARGET477	19380	7500	ZTARGET500	19382	7501	ZTARGET501
19384	7502	ZTARGET502	19386	7503	ZTARGET503	19388	7504	ZTARGET504	19390	7505	ZTARGET505
19392	7506	ZTARGET506	19394	7507	ZTARGET507	19396	7510	ZTARGET510	19398	7511	ZTARGET511
19400	7512	ZTARGET512	19402	7513	ZTARGET513	19404	7514	ZTARGET514	19406	7515	ZTARGET515
19408	7516	ZTARGET516	19410	7517	ZTARGET517	19412	7520	ZTARGET520	19414	7521	ZTARGET521
19416	7522	ZTARGET522	19418	7523	ZTARGET523	19420	7524	ZTARGET524	19422	7525	ZTARGET525
19424	7526	ZTARGET526	19426	7527	ZTARGET527	19428	7530	ZTARGET530	19430	7531	ZTARGET531
19432	7532	ZTARGET532	19434	7533	ZTARGET533	19436	7534	ZTARGET534	19438	7535	ZTARGET535
19440	7536	ZTARGET536	19442	7537	ZTARGET537	19444	7540	ZTARGET540	19446	7541	ZTARGET541

LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL	LINE	LOCN	SYMBOL
19448	7542	ZTARGET542	19450	7543	ZTARGET543	19452	7544	ZTARGET544	19454	7545	ZTARGET545
19456	7546	ZTARGET546	19458	7547	ZTARGET547	19460	7550	ZTARGET550	19462	7551	ZTARGET551
19464	7552	ZTARGET552	19466	7553	ZTARGET553	19468	7554	ZTARGET554	19470	7555	ZTARGET555
19472	7556	ZTARGET556	19474	7557	ZTARGET557	19476	7560	ZTARGET560	19478	7561	ZTARGET561
19480	7562	ZTARGET562	19482	7563	ZTARGET563	19484	7564	ZTARGET564	19486	7565	ZTARGET565
19488	7566	ZTARGET566	19490	7567	ZTARGET567	19492	7570	ZTARGET570	19494	7571	ZTARGET571
19496	7572	ZTARGET572	19498	7573	ZTARGET573	19500	7574	ZTARGET574	19502	7575	ZTARGET575
19504	7576	ZTARGET576	19506	7577	ZTARGET577	19508	7600	ZTARGET600	19510	7601	ZTARGET601
19512	7602	ZTARGET602	19514	7603	ZTARGET603	19516	7604	ZTARGET604	19518	7605	ZTARGET605
19520	7606	ZTARGET606	19522	7607	ZTARGET607	19524	7610	ZTARGET610	19526	7611	ZTARGET611
19528	7612	ZTARGET612	19530	7613	ZTARGET613	19532	7614	ZTARGET614	19534	7615	ZTARGET615
19536	7616	ZTARGET616	19538	7617	ZTARGET617	19540	7620	ZTARGET620	19542	7621	ZTARGET621
19544	7622	ZTARGET622	19546	7623	ZTARGET623	19548	7624	ZTARGET624	19550	7625	ZTARGET625
19552	7626	ZTARGET626	19554	7627	ZTARGET627	19556	7630	ZTARGET630	19558	7631	ZTARGET631
19560	7632	ZTARGET632	19562	7633	ZTARGET633	19564	7634	ZTARGET634	19566	7635	ZTARGET635
19568	7636	ZTARGET636	19570	7637	ZTARGET637	19572	7640	ZTARGET640	19574	7641	ZTARGET641
19576	7642	ZTARGET642	19578	7643	ZTARGET643	19580	7644	ZTARGET644	19582	7645	ZTARGET645
19584	7646	ZTARGET646	19586	7647	ZTARGET647	19588	7650	ZTARGET650	19590	7651	ZTARGET651
19592	7652	ZTARGET652	19594	7653	ZTARGET653	19596	7654	ZTARGET654	19598	7655	ZTARGET655
19600	7656	ZTARGET656	19602	7657	ZTARGET657	19604	7658	ZTARGET658	19606	7661	ZTARGET661
19608	7662	ZTARGET662	19610	7663	ZTARGET663	19612	7664	ZTARGET664	19614	7665	ZTARGET665
19616	7666	ZTARGET666	19618	7667	ZTARGET667	19620	7670	ZTARGET670	19622	7671	ZTARGET671
19624	7672	ZTARGET672	19626	7673	ZTARGET673	19628	7674	ZTARGET674	19630	7675	ZTARGET675
19632	7676	ZTARGET676	19634	7677	ZTARGET677	19636	7700	ZTARGET700	19638	7701	ZTARGET701
19640	7702	ZTARGET702	19642	7703	ZTARGET703	19644	7704	ZTARGET704	19646	7705	ZTARGET705
19648	7706	ZTARGET706	19650	7707	ZTARGET707	19652	7710	ZTARGET710	19654	7711	ZTARGET711
19656	7712	ZTARGET712	19658	7713	ZTARGET713	19660	7714	ZTARGET714	19662	7715	ZTARGET715
19664	7716	ZTARGET716	19666	7717	ZTARGET717	19668	7720	ZTARGET720	19670	7721	ZTARGET721
19672	7722	ZTARGET722	19674	7723	ZTARGET723	19676	7724	ZTARGET724	19678	7725	ZTARGET725
19680	7726	ZTARGET726	19682	7727	ZTARGET727	19684	7730	ZTARGET730	19686	7731	ZTARGET731
19688	7732	ZTARGET732	19690	7733	ZTARGET733	19692	7734	ZTARGET734	19694	7735	ZTARGET735
19696	7736	ZTARGET736	19698	7737	ZTARGET737	19700	7740	ZTARGET740	19702	7741	ZTARGET741
19704	7742	ZTARGET742	19706	7743	ZTARGET743	19708	7744	ZTARGET744	19710	7745	ZTARGET745
19712	7746	ZTARGET746	19714	7747	ZTARGET747	19716	7750	ZTARGET750	19718	7751	ZTARGET751
19720	7752	ZTARGET752	19722	7753	ZTARGET753	19724	7754	ZTARGET754	19726	7755	ZTARGET755
19728	7756	ZTARGET756	19730	7757	ZTARGET757	19732	7760	ZTARGET760	19734	7761	ZTARGET761
19736	7762	ZTARGET762	19738	7763	ZTARGET763	19740	7764	ZTARGET764	19742	7765	ZTARGET765
19744	7766	ZTARGET766	19746	7767	ZTARGET767	19748	7770	ZTARGET770	19750	7771	ZTARGET771
19752	7772	ZTARGET772	19754	7773	ZTARGET773	19756	7774	ZTARGET774	19758	7775	ZTARGET775
19760	7776	ZTARGET776	19762	7777	ZTARGET777						

LOCN	----	-----	-----	-----	-----	-----	-----
4000	TEST001	INITIALIZE12	WRITEDP350	ALU116A	GETZEROES114A	GOBUT114A	GOBUT114A2
4010	GETALTN115A1	GOBUT115A1	GOBUT115A4	GOBUT115A2	GOBUT115A3	ALU115B1	GOBUT115B4
4020	GORUT115B2	GOBUT115B3	SCOPE115B	GOBUT115C1	GOBUT115C2	GOBUT115C3	ALU115D1
4030	GORUT115D2	GOBUT115D3	SCOPE115D	GOBUT116A1	GOBUT116A2	GOBUT116A3	GOBUT116A4
4040	ALU116B	GOBUT116B	SCOPE116C	GOBUT116C1	GOBUT116C2	GOBUT116C3	GOBUT116C4
4050	ALU116D	GOBUT116D	SCOPE116D	GOBUT117A1	GOBUT117A2	GOBUT117A3	GOBUT117A4
4060	GOBUT117B1	GOBUT117B2	GOBUT117B3	ALU117C1	GOBUT117C1	GOBUT117C2	SCOPE117C
4070	GOBUT120A2	GOBUT120A3	ALU120B1	GOBUT120B1	GOBUT120B2	GOBUT120B3	GOBUT120B4
4100	LOADD10	INITIALIZE10	INITIALIZE11	GOBUT121A1	GOBUT121A2	GOBUT121A3	GOBUT121A4
4110	GORUT121B1	GOBUT121B2	GOBUT121B3	GOBUT121B4	SCOPE121C	GOBUT121C1	GOBUT121C2
4120	GORUT121C4	ALU121D1	GOBUT121D1	GOBUT121D2	GOBUT121D3	GOBUT121D4	SCOPE121D
4130	GORUT122A2	GORUT122A3	SETD122A4	GOBUT122A4	SCOPE122A	COMP130A1	GOBUT130A1
4140	ARITH130B1	COMP130B1	GOBUT130B1	GOBUT130B2	SCOPE130B	COMP131A1	GOBUT131A1
4150	ARITH131B1	COMP131A1	GOBUT131B1	GOBUT131B2	SCOPE131B	GOBUT132A1	GOBUT132A2
4160	GOBUT132B1	GOBUT132B2	SCOPE132B	DOPB133A1	GOBUT133A1	GOBUT133A2	OPA133B1
4170	GORUT133B2	SCOPE133B	DOPB134A1	DOPB134A1	GOBUT134A1	GOBUT134A2	OPA134B1
4200	GORUT134B2	SCOPE134B	DOPB135A1	DOPB135A1	GOBUT135A1	GOBUT135A2	OPA135B1
4210	GORUT135B2	SCOPE135B	DOPB136A1	GOBUT136A1	GOBUT136A2	OPA136B1	GOBUT136B2
4220	SCOPE136B	SETD320A	SETDC320A	GOBUT320A	GOBUT320B	SETONE320C	SETDC320C
4230	GORUT320C	LOADB320D	SETONE320E	SETD320E	GOBUT320E	GOBUT320F	SCOPE320
4240	START350	LOADSRD350	BL1-537A	GOBUT350A	COMP350B	GOBUT350B	GOBUT350C
4250	COMP350D	GOBUT350D	SCOPE350	NEXTPR350	LOADD351	VFY003	GOBUT351A
4260	GOBUT351A	COMP351C	GOBUT351C	COMP351D	GOBUT351D	SCOPE351	GOBUT352A
4270	GORUT352B	COMP352C	GOBUT352C	COMP352D	GOBUT352D	SCOPE352	LOADD4537A
4300	LOADD1537A	COUNTER01	COMP937A	GOBUT937A	SCOPE937A	LOADIN551A	SETSR551A
4310	COUNTER02	BUSFCN711A	SETJAM711A	BUSFCN710A	GOBUT710A	LOADD762A	BUSFCN712A
4320	LOADDRET721A	GOBUT721B	GOBUT721C	SETJAM722A	BUSFCN722A	GOBUT722B	GOBUT722C
4330	LOADDADA730A	BUSFCN730A	GOBUT730A	LOADD730A	GOBUT730A	GOBUT730A	EXPEC730B
4340	GETIT730B	GOBUT730B	GOBUT730C	LOADD730D	GOBUT730D	GOBUT730D	BUSFCN730D
4350	GORUT730D	GOBUT730E	SCOPE730	GOBUT731A	GOBUT731A	GOBUT731A	GOBUT731B
4360	GORUT731B	COMP731C	GOBUT731C	LOADD731D	BUSFCN731D	GOBUT731D	GOBUT731E
4370	SCOPE731	BUSFCN740A	GOBUT740B	BUSFCN740C	GOBUT740C	BUSFCN740C	TEST007
4400	LOADDIR740D	BUSFCN740D	SCOPE740	GOBUT740D	GOBUT740D	GOBUT740D	GOBUT741C
4410	SCOPE761	DH11L762A	VFY002	TESTPR762A	TESTPR762A	GOBUT762B	EXPEC762E
4420	GOBUT762E	GOBUT762F	EXPEC763A	READVET763A	TESTVET763A	GOBUT763B	GOBUT763D
4430	SCOPE763	EDOP001	GOBUT763C	VFY004	JAMUPP002C	JAMUPP006	JAMUPP010
4440	JAMUPP011	TEST740C	GOBUT7730C1	TEST740C	TEST740C	NEXT7710A	NEXT7712A
4450	TEST730C1	TEST730D	TEST730E	TEST730F	TEST730F	TEST731C	TEST731B
4460	TEST730C	TEST730D	TEST730E	TEST730F	TEST730F	TEST730B	NEXTD762A
4470	SETAND722A	TEST732B	TEST732B	TEST732B	TEST732B	TEST732B	TEST732B
4500	TEST733A2	TEST733A2	TEST733A2	TEST733A2	TEST733A2	TEST733A2	TEST733A2
4510	OPA133A1	TEST734A1	OPA134A1	TEST734A1	TEST734A1	TEST734A1	TEST732B1
4520	ARITH131A1	TEST732A1	ARITH132A1	TEST732A1	TEST732A1	COUNTER03	COUNTER10
4530	TEST130B1	COUNTER10B	TEST130B2	COUNTER10A	COUNTER06	COUNTER07	TEST320D
4540	ARITH130A1	TEST131A1	OPB136A1	TEST1320A	TEST131A2	COUNTER04	TEST131B1
4550	TEST130A2	RUMPD762A	TEST1320B	TEST1320B	TEST1320B	COUNTER12	COUNTER05
4560	GOTEST951B	SETSR551C	TEST1320C	TEST1320C	GOTEST551A	SETSR551B	LOADD05-351
4570	TEST122A2	TEST721A	TEST1320D	TEST1320D	LOADD16537A	TEST551A	TEST721C
4580	GETONE8114A	TEST115A1	TEST115B4	TEST115B4	TEST115A2	TEST115A2	TEST114A
4610	ALU115A1	TEST115C1	TEST115B3	TEST115B3	TEST115B1	TEST115A3	TEST115C2
4620	TEST116A3	TEST116A2	ALU115C1	TEST115D3	TEST115D1	TEST115D2	TEST115D1
4630	TEST116C5	TEST116C4	TEST116C3	TEST116C2	ALU116A1	TEST116C1	TEST116A4

LOCN	-----0-----	-----1-----	-----2-----	-----3-----	-----4-----	-----5-----	-----6-----	-----7-----
4640	TEST117B2	TEST117B1	TEST117A4	TEST117A3	TEST117A2	TEST116D	ALU116C1	TEST117A1
4650	TEST120B4	TEST120B3	TEST120B2	TEST120B1	TEST120A3	TEST120A2	TEST117C1	TEST117B3
4660	TEST121B4	TEST121B3	TEST121B2	TEST121B1	ALU120A1	TEST121A1	ALU117A1	TEST120A1
4670	ALU121C1	TEST122A1	ALU122A1	TEST130A1	TEST121A4	TEST121A3	TEST121A2	TEST117C2
4700	SETONE320A	TEST350	COMP350A	REST350B	COMP351A	TEST352A	ALU121A1	TEST121C1
4710	TEST763D	TEST762F	TEST351D	TEST352B	COMP352A	RESTORE01	TEST121C4	TEST121C3
4720	TEST763B	TEST763C	TEST351B	TEST351C	TEST353A	TEST353D	TEST121D2	TEST121D1
4730	TEST762E	ALLOW763A	TEST351A	VECTLOAD763A	TEST354A	TEST123A4	TEST121D4	TEST121D3
4740	TEST762C	TEST762D	TEST350C	TEST350D	TEST740A	TEST761A	JANUPP004	JANUPP005
4750	TEST131B2	TEST762B	TEST350A	TEST350B	LOADIR730A	TEST731A	JANUPP003	JANUPP002B
4760	TEST720C	TEST763A	GOBUT762B	KILL764A	TEST731A	LOADIR740A	GOBUT740B	GOBUT740D
4770	.	TEST761B	ZERD08R761A	TEST762A	TEST731C	TEST010	GOBUT740A	JANUPP001

LOCN	-----0-----	-----1-----	-----2-----	-----3-----	-----4-----	-----5-----	-----6-----	-----7-----
5000	SFTDC372B	LOAD011	LOAD012A	GOBUT012A	GOBUT012B	GOBUT012C	GOBUT012D	GOBUT012E
5010	GOBUT012F	GOBUT012G	SCOPE012	GOBUT013A	GOBUT013B	GOBUT013C	GOBUT013D	GOBUT013E
5020	GOBUT013F	GOBUT013G	SCOPE013	GOBUT014A	GOBUT014B	GOBUT014C	GOBUT014D	SCOPE014
5030	GOBUT015A	GOBUT015B	GOBUT015C	SCOPE015	GOBUT016A	SCOPE016	GOBUT017A	GOBUT017B
5040	SCOPE017	GOBUT020A	GOBUT020B	GOBUT020C	SCOPE020	GOBUT021A	GOBUT021B	GOBUT021C
5050	SCOPE021	GOBUT022A	SCOPE022	GOBUT023A	SCOPE023	GOBUT024A	GOBUT024B	SCOPE024
5060	GOBUT025A	SCOPE025	GOBUT026A	SCOPE026	GOBUT027A	SCOPE027	GOBUT030A	GOBUT030B
5070	GOBUT030C	GOBUT030D	SCOPE030	GOBUT031A	GOBUT031B	SCOPE031	GOBUT032A	GOBUT032B
5100	SCOPE032	GOBUT033A	SCOPE033	GOBUT034A	SCOPE034	GOBUT035A	SCOPE035	GOBUT036A
5110	SCOPE036	GOBUT037A	SCOPE037	GOBUT040A	GOBUT040B	SCOPE040	GOBUT041A	GOBUT041B
5120	SCOPE041	GOBUT042A	SCOPE042	GOBUT043A	GOBUT043B	SCOPE043	GOBUT044A	GOBUT044B
5130	SCOPE044	GOBUT045A	SCOPE045	GOBUT046A	GOBUT046B	SCOPE046	GOBUT047A	GOBUT047B
5140	GOBUT047C	SCOPE047	GOBUT050A	LOAD050B	GOBUT050B	SCOPE050	TEST005	GOTEST101A
5150	SCOPE101	LOAD02-102A	LOAD04-102A	LOAD10-102A	LOAD00-102A	LOADD102A	GOTEST102A	LOADD102B
5160	GOTEST102B	LOADD102C	GOTEST102C	LOADD102D	GOTEST102D	SCOPE102	LOAD15-103A	LOADD13-103A
5170	LOAD07-103A	LOADD17-103A	GOTEST103A	GOTEST103B	LOADD103B	GOTEST103B	LOADD103C	GOTEST103C
5200	LOADD103D	GOTEST103D	SCOPE103	GOTEST104A	LOADD104B	GOTEST104B	SCOPE104	GOTEST105A
5210	GOBUT105A1	LOADD8105B	GOTEST105B	GOBUT105B1	LOADD8105C	GOTEST105C	LOADD8105D	GOTEST105D
5220	GOTEST105E	SCOPE105	EXPEC374A1	SUBRA374B	SCOPE624	DOWRITE374A1	GETTEN374A1	ZERO374A2
5230	DOWRITE374A2	GETTEN374A2	GOTEST374A2	EXPEC374A	DOWRITE374B1	GETTEN374B1	ZERO374B2	DOWRITE374B2
5240	GETTEN374B2	GOTEST374B2	SCOPE374B	DOWRITE374C1	GETTEN374C1	NEXT010	ZERO374C2	DOWRITE374C2
5250	GETTEN374C2	GOTEST374C2	SCOPE374C	DOWRITE374D1	GETTEN374D1	ZERO374D2	DOWRITE374D2	GETTEN374D2
5260	GOTEST374D2	SCOPE374D	DOWRITE374E1	GETTEN374E1	ZERO374E2	DOWRITE374E2	GETTEN374E2	GOTEST374E2
5270	SCOPE374E	DOWRITE374F1	GETTEN374F1	ZFR0374F2	DOWRITE374F2	GETTEN374F2	SCOPE374F	SCOPE374F
5300	CHECK375B	GOBUT375B	SETIR376A	GOBUT376A	SCOPE376	SCOPE551	LOADD1-610A1	LOADD05-610A1
5310	LOADD06-610A1	P8CC-DC610A1	SETBUSA610A1	DOIT610A1	GETT610A1	LOADD06-610A2	LOADD01-610A2	P8CC-DC610A2
5320	SETBUSA610A2	DOIT610A2	GETT610A2	SCOPE610A	LOADD01-610B1	P8CC-DC610B1	SETBUSA610B1	DOIT610B1
5330	GETT610B1	LOADD07-610B2	LOADD07-610B2	LOADD01-610B2	P8CC-DC610B2	SETBUSA610B2	DOIT610B2	GETT610B2
5340	SCOPE610A	LOADD01-610C1	P8CC-DC610C1	SETBUSA610C1	DOIT610C1	GETT610C1	LOADD08-610C2	LOADD01-610C2
5350	P8CC-DC610C2	SETBUSA610C2	DOIT610C2	GETT610C2	SCOPE610C	P8CC-DC610D1	SETBUSA610D1	DOIT610D1
5360	GETT610D1	P8CC-DC610D2	SETBUSA610D2	DOIT610D2	GETT610D2	SCOPE610D	GOBUT620A	GOBUT620B
5370	MASK620C	GETJAM620C	CSP1L621A	SETBRK621A	LOADBRK621A	SETRE7621A	BUTRRORS	SETPLG621A
5400	LOADF6G21A	SETBRK621A	GOBUT621B	GOBUT621C	GOBUT621D	EXPEC621E	GETJAM621E	EXPEC621F
5410	GETCUA621F	GOBUT621G	COMP621H	BC1FC622A	CN8T0622A	GOBUT622A	NCERC622B	GOBUT622B
5420	GETJAM622C	EXPEC623	GETCUA623	SCOPE623	SETBRK624A	LOADBRK624A	SETRE7624A	ZEROIR624B
5430	EXPEC624C	GETCUA624C	GOBUT624D	SCOPE702	GOBUT701B	GOBUT701D	SCOPE701	EXPEC702A
5440	DOIT376A	SETLED702A	SETKT702A	LOADBA702A	GOTEST702A	SCOPE710	SCOPE711	SCOPE712
5450	GOTEST712B	SCOPE713	EXPEC713B	GOTEST713B	SCOPE721	LOADRET722A	LOADRET722A	TEST713A
5460	.	.	LOADRET720A	TEST722A	LOADRET713A	TEST720A	LOADRET712A	TEST712A
5470	LOADRET711A	TEST712A	LOADRET710A	TEST711A	MASK702A	TEST710A	MASK701A	TEST702A
5500	TEST174D1	SCOPE372B	TEST374E2	LOAD372B	TEST374F2	TEST712B	SETSP375A	TEST7410
5510	LOADD1-610D1	TEST620A	TEST701D	TEST374D1	ZERO374D1	EXPEC374E1	ZERO374F1	TEST375A
5520	LOADD101A	TEST102A	UBRK621A	TEST701B	TEST374C2	TEST004	ZEP0374E1	EXPEC374F1
5530	LOADD01-102A	TEST103A	TEST610B1	TEST610B1	LOADIR610B1	TEST610C1	ZERO374A1	EXPEC374B1
5540	LOADD104A	TEST105A	LOADD8105A	SETUPC8P17A	LOADIR610C1	TEST610D1	LOAD16-103A	TEST104A
5550	LOADD047A	TEST050A	LOADD050A	TEST101A	TEST102D	CUA372B	ZERO374B1	EXPEC374C1
5560	LOADD046A	TEST047A	LOADIR624A	TEST701A	TEST102C	TEST624B	ZERO374C1	EXPEC374D1
5570	NCERC620A	TEST624A	TEST103B	TEST623	TEST102B	TEST622C	TEST103C	TEST622B
5600	TEST105B1	TEST622A	TEST610C2	TEST621F	TEST374A2	TEST621E	TEST374B2	TEST621D
5610	TEST105A1	TEST621C	TEST610B2	TEST621B	TEST376A1	ERROR621A	TEST713B	TEST621A
5620	TEST105F	TEST620C	TEST610A2	TEST620B	TEST610D2	TEST375B	GOTEST551C	TEST610A1
5630	LOADD045A	TEST046A	LOADD044A	TEST045A	LOADD043A	TEST044A	TEST105D	TEST043B

LOCN	-----0-----	-----1-----	-----2-----	-----3-----	-----4-----	-----5-----	-----6-----	-----7-----
5640	LOAD042A	TEST043A	TEST108C	TEST042B	LOAD041A	TEST042A	TEST104B	TEST041B
5650	LOAD040A	TEST041A	TEST105B	TEST040B	LOAD037A	TEST040A	LOAD036A	TEST037A
5660	LOAD035A	TEST036A	LOAD034A	TEST035A	LOAD033A	TEST034A	LOAD032A	TEST033A
5670	TEST047B	TEST032B	LOAD031A	TEST032A	TEST103B	TEST031B	LOAD030A	TEST031A
5700	LOAD027A	TEST030A	LOAD026A	TEST027A	LOAD025A	TEST026A	LOAD024A	TEST025A
5710	TEST014D	TEST024B	TEST030B	TEST030C	LOAD023A	TEST024A	LOAD022A	TEST023A
5720	LOAD021A	TEST022A	TEST015C	TEST021B	LOAD020A	TEST021A	TEST080B	TEST020C
5730	TEST047C	TEST020B	LOAD017A	TEST021B	TEST080A	TEST017B	LOAD016A	TEST017A
5740	LOAD015A	TEST016A	LOAD014A	TEST080A	TEST013A	TEST014A	8FDFT08R	TEST013G
5750	TEST013F	TEST013E	TEST014B	TEST013D	TEST0401B	TEST013C	TEST624D	TEST013B
5760	SETIRA	TEST013A	TEST014C	TEST013E	TEST013F	TEST012E	TEST621G	TEST012D
5770	TEST046B	TEST012C	TEST015B	TEST013B	TEST041C	TEST012A	TEST624C	TEST011

LOCN	-----0-----	-----1-----	-----2-----	-----3-----	-----4-----	-----5-----	-----6-----	-----7-----
6000	SETHYTEE410	LOADRES361A	LOADSR361A	EXPEC361B	COMP361B	GOBUT361D	GOBUT361E	SETRES362A
6010	SFTDDC362A	LOADRES362A	COMP362A	GOBUT362A	EXPEC362B	COMP362B	GOBUT362D	GOBUT362E
6020	SETDDC363A	SHIFT363A	COMP363A	GOBUT363A	GOBUT363B	SETDDC364A	GOBUT364A	SETDDC364B
6030	GORUT364B	SETPER365A	LOADRES365A	EXPEC365A	COMP365A	GOBUT365B	SETRES366A	LOADRES366A
6040	SHIFT366A	EXPEC366A	COMP366A	GOBUT366B	SETDDC366C	SHIFT366C	EXPEC366C	COMP366C
6050	SETRES367A	LOADRES367A	LOADSR367A	GOBUT367B	SETRES370A	LOADRES370A	SHIFT370A	COMP370A
6060	GOBUT370A	GOBUT370B	SETDDC370C	SHIFT370C	COMP370C	GOBUT370C	GOBUT370D	SETRES371A
6070	LOADRES371A	SHIFT371A	COMP371A	GOBUT371A	GOBUT371B	SCOPE371	SETDC372A	SUBRA372A
6100	PAGE372B	GOBUT373B	SCOPE373	LOADRES374	GOBUT373B	SETBYTEG410	COMP410A	INTOIR410A
6110	EXPEC410B	COMP410B	GOBUT410B	EXPEC410C	ASIDE410C	COMP410C	GOBUT410C	EXPEC410D
6120	COMP410D	GOBUT410D	EXPEC410E	COMP410E	GOBUT410E	SCOPE410	SETFETCND500	SETFETCHE500
6130	SETFETCHF400	SETFETCHG500	LOADIRS00	TESTINHBSP500	COMP500A	GOBUT500A	EXPEC500C	COMP500C
6140	GORUT500C	COMP500D	GOBUT500D	EXPEC500E	COMP500E	GOBUT500E	SCOPE500F	LOADFPS503A
6150	EXPEC503A	MASK503A	LOADFCC503A	DOFCC503A	GOBUT503A	GOBUT503A	GOBUT503B	GOBUT503C
6160	GORUT503B	LOADIRS03D	GOBUT503D	EXPEC503E	GOBUT503E	GOBUT503F	GOBUT503G	GOBUT503H
6170	GORUT503I	GOBUT503J	GOBUT503K	SCOPE503	LOADFPS504A	EXPEC504A	LOADFCC504A	DOFCC504A
6200	GORUT504A	GOBUT504A	GOBUT504B	GOBUT504C	LOADIRS04D	GOBUT504D	EXPEC504E	GOBUT504E
6210	GORUT504F	GOBUT504G	GOBUT504H	GOBUT504I	SCOPE504	EXPEC505A	SETFLAGS05A	BUTCLRS05A
6220	GORUT505A	GOBUT505B	CUA1372A	GOBUT505C	SCOPE505	SETUCONS06A	PSHIS06A	PSMID506A
6230	PSLOS06A	FUDGFP506A	MF5801	GOBUT506B	GOBUT506D	GOBUT506D	CLEAR506E	GOBUT506E
6240	SCOPE506	PSHIS07A	PSMID507A	P8LOS07A	FUDGFP507A	LOADDC507A	GOBUT507A	GOBUT507B
6250	GORUT507C	GOBUT507D	TEST002	GOBUT507E	CIN507F	UBRK624A	SETP507F	SETSR507F
6260	SETCIN507F	GOBUT507F	SCOPE507	LOADPRJ0510A	GOBUT510A	GOBUT510B	GOBUT510C	FLAGS10D
6270	FNFLAGS10D	SETFLAGS10D	GOBUT510D	GOBUT510DA	ZEROD510E	GOBUT510E	ZEROFLAGS10E	GOBUT510F
6300	SCOPE510	SCOPE511A	MF5802	GOBUT511A	MF5804	SCOPE511B	SETD624A	SETUP512B
6310	SETPSFLAGS12A1	GOTEST512A1	GOTEST512A2	SCOPE512A	SETPSFLAGS12B1	GOTEST512B1	GOTEST512B2	SETIR512C1
6320	SETPSFLAGS12C1	GOTEST512C1	GOTEST512C2	SCOPE512C	SETPSFLAGS12D1	GOTEST512D1	GOTEST512D2	SETIR512E1
6330	SETPSFLAGS12E1	GOTEST512E1	GOTEST512E2	SCOPE512E	SETUP520A	GOTEST520A	SETUP520B	GOTEST520B
6340	SFTUP520C	GOTEST520C	SETUP520D	GOTEST520D	SETUP520E	GOTEST520E	SCOPE520	INITD533A
6350	COMP533A	GOBUT533A	INITD533B	COMP533B	COMP533B	GOBUT533B	SCOPE533B	COMP534A
6360	GORUT534A	INITD534B	COMP534B	GOBUT534B	SCOPE534B	COMP534C	GOBUT534C	INITD534D
6370	COMP534D	GORUT534D	SCOPE534D	INITD534E	COMP534E	GOBUT534E	BUTERR06	NEXTPAT500
6400	LOADCNTR534F	INITD534F	COMP534F	GOBUT534F	COUNTER534G	GOBUT534G	SCOPE534G	EXPEC535A
6410	INITD535A	GOBUT535A	COMP535A	EXPEC535B	INITD535B	COMP535B	GOBUT535B	SCOPE535B
6420	EXPEC536A	INITD536A	COMP536A	GOBUT536A	EXPEC536B	INITD536B	COMP536B	GOBUT536B
6430	SCOPE536C	COMP536C	COMP536C	INITD536D	EXPEC536D	INITD536D	COMP536D	GOBUT536D
6440	SCOPE536D	INITD536E	COMP536E	GOBUT536E	LOADSR536F	INITD536F	COMP536F	GOBUT536F
6450	EXPEC536F	AP3-537A	EOP002	SETFLG624A	SETSR624A	EXPEC701C	LOADBA701C	GOTEST701C
6460	EXPEC702B	SETKT702B	LOADBA702B	GOTEST702B	GOTEST710B	EXPEC710C	GOTEST710C	EXPEC710D
6470	CLFAR710D	GOTEST710D	GOTEST710E	EXPEC711B	GOTEST711B	EXPEC711C	GOTEST711C	GOTEST711D
6500	GOBUT712C	EXPEC712D	GOTEST712D	GOBUT713C	EXPEC713D	GOTEST713B	GOTEST720B	...
6510	EOP003	EOP004	EOP007	TEST712D	TEST712C	ASSERTPOV500	TEST711D	FIXPAT500
6520	TEST720B	TEST713D	TEST713C	TEST711B	TEST711B	TEST512E2	SETIR512B1	TEST512D1
6530	TEST711C	DNONZEP0410	INITIALIZE01	DEZNO410	TEST512A2	MF58EXPEC0	TEST512B2	MF58EXPEC1
6540	LOOP372A	TEST373A	TEST512A1	ERROR010	TEST512A2	SETUP512A	TEST512B2	TEST507D
6550	GOTEST511A1	TEST511B1	MF5805	MF5806	GOTEST511B1	TEST512B1	TEST512C1	ERROR624A
6560	RETRBYTEB410	TEST500	RETURN373A	TEST374	TESTIR512A1	TEST710F	TEST512D2	TEST512E1
6570	SETRES361A	TEST372A	TEST373B	TEST361A	TEST710D	TEST510E	SETONE5510A	TEST511A
6600	TEST510D	TEST503DA	EXPEC507A	TEST510E	LOADFLAG503A	TEST504A	TEST510C	TEST510DA
6610	TEST510F	TEST507C	EXPEC506A	TEST507A	UCON520A	TEST533A	SETIR512D1	TEST520A
6620	LOADFLAG504A	TEST506A	TEST503C	TEST508A	TEST533B	TEST534B	TEST534F	TEST535B
6630	TEST503B	TEST003		TEST533B				

LOCN	-----0-----	-----1-----	-----2-----	-----3-----	-----4-----	-----5-----	-----6-----	-----7-----
6640	TEST536R	TEST536D	TEST536F	NEWCTRS37A	TEST710B	TEST710C	TEST503K	TEST507F
6650	LOADSR536E	SETEMIT537A	INITDS36C	TEST536E	INIT536A	TEST536C	INIT535A	TEST536A
6660	LOADCNTR534E	TEST535A	INITDS34C	TEST534E	INITDS34A	TEST534C	INIT533A	TEST534A
6670	TEST503D	TEST5000	TEST504F	TEST500A	TEST507E	TEST510B	SETFETCHB500	TEST503A
6700	TEST503E	TEST372B	TEST504G	LOAD372A	TEST410B	TEST410C	TEST504H	TEST410C
6710	TEST503F	TEST410A	TEST505B	TEST371B	TESTA503A	TEST371A	TEST504I	TEST410D
6720	TEST503G	TEST370C	TEST505C	TEST370B	TEST507F	TEST370A	TEST500C	TEST410D
6730	TEST504B	TEST367A	TEST503M	TEST366C	TEST506D	TEST366B	TEST500E	TEST366A
6740	TEST504C	TEST365R	TEST503I	TEST365A	TEST504C	TEST364B	TEST500D	TEST364A
6750	TEST504D	TEST363B	TEST534G	TEST363A	TEST504B	TEST362E	TEST503J	TEST362D
6760	TEST504E	TEST702B	TEST520D	TEST362B	TEST504B	TEST362A	TESTA504A	TEST361E
6770	TEST701C	TEST361D	TEST520C	TESTD410	VFY001	TEST520E	EOP006	TESTINHASP500

LOCN	-----0-----	-----1-----	-----2-----	-----3-----	-----4-----	-----5-----	-----6-----	-----7-----
7000	SUBR010	INITIALIZE03	SETUPCSP16A	INITIALIZE04	INITIALIZE05	INITIALIZE06	INITIALIZE07	ALUCARRY1
7010	SETUPCSP15A	SETUPCSP14A	SETUPCSP12A	SETUPCSP05A	SETUPCSP07A	SETUPCSP00A	WRITESF350	ALUCARRY1A
7020	ALUCARRY2	ALUCARRY2A	RESETIR350A	NEXTPAT350	LOADR350	AGAINR350	RESETIR350B	RESETIR350C
7030	PESTORE02	CSP16XORSRT01RS	RESTORE03	RESTORE04	SETIR373A	CSP16XORFLTT01R	ZEROSF02DF0A	SETSR373A
7040	SFDFPTOSRAA	ZEROSF04DF02	ZEROSFDF	ZEROSFDF	ZEROSF	ZEROSFA	FIRST375A	SFDFPTOSRA
7050	SFDFPTOSRB	SFDFPTOSRC	SFDFPTOSRD	SFDFPTOSRE	SFDFPTOSRF	SFDFPTOSRG	SFDFPTOSRH	SFDFPTOSRI
7060	SETBYTEC410	SECON0375A	BYTE375A	GOBUT375A	GOBUT375A	NEXTPAT410	SETBYTED410	LOADUCONS06A
7070	RYTEPIRST410	GOBUT506A	DOIT506A	FLAGF8SEGL0D	TEST511A2	FLAGF802	FLAGF803	FLAGF804
7100	PSRGL0D	PSRGL0D02	PSRGL0D03	PSRGL0D04	PSRGL0D05	PSRGL0D06	TEST511B2	GOTEST511A2
7110	TEST511A3	GOTEST511A3	TEST511A4	GOTEST511A4	KTSCDST01	GOTEST511B2	TEST511B3	GOTEST511B3
7120	TEST511A4	GOTEST511B4	SUCBRTST01	KTSCDST02	KTSCDST03	KTSCDST04	KTSCDST04B	KTSCDST07
7130	KTSCDST0R	INITSR37A	SUCBRTST02	SUCBRTST03	BR3-537A	SETRESA537A	BUTCOUNT-18-377	CR3-537A
7140	DR3-537A	AR3-537A	BR3-537A	CR3-537A	DR3-537A	AR11-537A	BR11-537A	CR11-537A
7150	DR11-537A	ER3-537A	FR3-537A	GR3-537A	MR3-537A	AL11-537A	SETUPP8CC-DC	GOTEST551B1
7160	GOBUT621H	SETUPP8CC-DC02	PSCTTOSR3-0	PSCTTOSR3-0AA	PSCTTOSR3-0BB	CLEAR624	EXPEC701A	BUSFCN713A
7170	LOADBA701A	GOTEST701A	LOADIR720A	SETADR721A	SETADR720A	SETJAM720A	BUSFCN720A	BDX12
7200	SETJAM721A	BUSFCN721A	MASK762A	C17X12	BDX08	C17X05	SETPRI762C	PRIO6762A
7210	GETIT762A	EOP005	EOP010	JAMUPP002D	INSERTEOPREVNO	INSERT02	INSERTREVNO	INSERT03
7220	INSFR704	DISP002	DISP003	DISP004	DISP005	DISP006	D111-06	CLEAR-1-0-A
7230	CLEAR-1-0-B	CLEAR1002	CLEAR1004	CLEAR1005	D113-12	D1512A	DATOSERVICETOD	CJESERVICETOD
7240	D105-00	D0500A	DZERO	DTOIRB	CLASERVICETOD	DATISERVICETOD	CUATOD	CUATOD
7250	SERVICETOD	PBATOD	FLAGF8STOD	PSTOD	ODDJAMTOD	CLAJAMTOD	JAMTOD	DINTOIR
7260	GETMSKPPROCDAT	MSKPPROCDAT	CMPPROCDAT	RESETPROCDAT	LOADFP8CC	LOADFP8CC02	RESETPROCDAT	BUTSR3-0
7270	DBUFINTOIR	RESETPROCDAT	SRINTOIRS	DINTOIRS	DBUFINTOIRS	LOADFP8CC02	RESETPROCDAT	BUTSR3-0
7300	BUTIR15-12	BUTINSTRS	BUTIR11FLTP73-0	NEXT007	BUTIR15-12	BUTIR15-12	BUTIR15-12	BUTIR15-12
7310	BUTD(CIC)	BUTCOUT7DOU7	BUTP805	BUTDM8M8YTE	BUTD(CIC)	BUTP805	BUTDM8M8YTE	BUTP805
7320	BUTMNT000	BUTMD00	BUTMNT001	BUTMNT002	BUTMNT003	BUTMNT004	BUTMNT005	BUTMNT006
7330	BUTMNT004	BUTMFLTP7S	BUTMNT005	BUTMFLTP7S	BUTMNT006	BUTMNT007	BUTMFLTP7S	BUTMNT008
7340	BUTMNT007	BUTD-15-ZERO	BUTIR11B	BUTP815	BUTMNT008	BUTMNT009	BUTMFLTP7S	BUTMNT010
7350	BUTD(CIC)	BUTR000	BUTOTHERJAM	EXPEC410A	BUTP815	BUTMNT010	BUTMFLTP7S	BUTMNT011
7360	BUTPREFFETCHJAM	GOFDR010	NEXT721A	GETP805	BUTP815	BUTMNT011	BUTMFLTP7S	BUTMNT012
7370	SETRES8537A	BUTP805	BUTP805	GETP805	BUTP805	BUTMNT012	BUTMFLTP7S	BUTMNT013
7400	ZTARGT400	ZTARGT401	ZTARGT402	ZTARGT403	ZTARGT404	ZTARGT405	ZTARGT406	ZTARGT407
7410	ZTARGT410	ZTARGT411	ZTARGT412	ZTARGT413	ZTARGT414	ZTARGT415	ZTARGT416	ZTARGT417
7420	ZTARGT420	ZTARGT421	ZTARGT422	ZTARGT423	ZTARGT424	ZTARGT425	ZTARGT426	ZTARGT427
7430	ZTARGT430	ZTARGT431	ZTARGT432	ZTARGT433	ZTARGT434	ZTARGT435	ZTARGT436	ZTARGT437
7440	ZTARGT440	ZTARGT441	ZTARGT442	ZTARGT443	ZTARGT444	ZTARGT445	ZTARGT446	ZTARGT447
7450	ZTARGT450	ZTARGT451	ZTARGT452	ZTARGT453	ZTARGT454	ZTARGT455	ZTARGT456	ZTARGT457
7460	ZTARGT460	ZTARGT461	ZTARGT462	ZTARGT463	ZTARGT464	ZTARGT465	ZTARGT466	ZTARGT467
7470	ZTARGT470	ZTARGT471	ZTARGT472	ZTARGT473	ZTARGT474	ZTARGT475	ZTARGT476	ZTARGT477
7500	ZTARGT500	ZTARGT501	ZTARGT502	ZTARGT503	ZTARGT504	ZTARGT505	ZTARGT506	ZTARGT507
7510	ZTARGT510	ZTARGT511	ZTARGT512	ZTARGT513	ZTARGT514	ZTARGT515	ZTARGT516	ZTARGT517
7520	ZTARGT520	ZTARGT521	ZTARGT522	ZTARGT523	ZTARGT524	ZTARGT525	ZTARGT526	ZTARGT527
7530	ZTARGT530	ZTARGT531	ZTARGT532	ZTARGT533	ZTARGT534	ZTARGT535	ZTARGT536	ZTARGT537
7540	ZTARGT540	ZTARGT541	ZTARGT542	ZTARGT543	ZTARGT544	ZTARGT545	ZTARGT546	ZTARGT547
7550	ZTARGT550	ZTARGT551	ZTARGT552	ZTARGT553	ZTARGT554	ZTARGT555	ZTARGT556	ZTARGT557
7560	ZTARGT560	ZTARGT561	ZTARGT562	ZTARGT563	ZTARGT564	ZTARGT565	ZTARGT566	ZTARGT567
7570	ZTARGT570	ZTARGT571	ZTARGT572	ZTARGT573	ZTARGT574	ZTARGT575	ZTARGT576	ZTARGT577
7600	ZTARGT600	ZTARGT601	ZTARGT602	ZTARGT603	ZTARGT604	ZTARGT605	ZTARGT606	ZTARGT607
7610	ZTARGT610	ZTARGT611	ZTARGT612	ZTARGT613	ZTARGT614	ZTARGT615	ZTARGT616	ZTARGT617
7620	ZTARGT620	ZTARGT621	ZTARGT622	ZTARGT623	ZTARGT624	ZTARGT625	ZTARGT626	ZTARGT627
7630	ZTARGT630	ZTARGT631	ZTARGT632	ZTARGT633	ZTARGT634	ZTARGT635	ZTARGT636	ZTARGT637

LOCN	-----0-----	-----1-----	-----2-----	-----3-----	-----4-----	-----5-----	-----6-----	-----7-----
7640	ZTARGET640	ZTARGET641	ZTARGET642	ZTARGET643	ZTARGET644	ZTARGET645	ZTARGET646	ZTARGET647
7650	ZTARGET650	ZTARGET651	ZTARGET652	ZTARGET653	ZTARGET654	ZTARGET655	ZTARGET656	ZTARGET657
7660	ZTARGET660	ZTARGET661	ZTARGET662	ZTARGET663	ZTARGET664	ZTARGET665	ZTARGET666	ZTARGET667
7670	ZTARGET670	ZTARGET671	ZTARGET672	ZTARGET673	ZTARGET674	ZTARGET675	ZTARGET676	ZTARGET677
7700	ZTARGET700	ZTARGET701	ZTARGET702	ZTARGET703	ZTARGET704	ZTARGET705	ZTARGET706	ZTARGET707
7710	ZTARGET710	ZTARGET711	ZTARGET712	ZTARGET713	ZTARGET714	ZTARGET715	ZTARGET716	ZTARGET717
7720	ZTARGET720	ZTARGET721	ZTARGET722	ZTARGET723	ZTARGET724	ZTARGET725	ZTARGET726	ZTARGET727
7730	ZTARGET730	ZTARGET731	ZTARGET732	ZTARGET733	ZTARGET734	ZTARGET735	ZTARGET736	ZTARGET737
7740	ZTARGET740	ZTARGET741	ZTARGET742	ZTARGET743	ZTARGET744	ZTARGET745	ZTARGET746	ZTARGET747
7750	ZTARGET750	ZTARGET751	ZTARGET752	ZTARGET753	ZTARGET754	ZTARGET755	ZTARGET756	ZTARGET757
7760	ZTARGET760	ZTARGET761	ZTARGET762	ZTARGET763	ZTARGET764	ZTARGET765	ZTARGET766	ZTARGET767
7770	ZTARGET770	ZTARGET771	ZTARGET772	ZTARGET773	ZTARGET774	ZTARGET775	ZTARGET776	ZTARGET777

PAGE	---USED---	---OPEN---
	OCTAL/DEC.	OCTAL/DEC.
4	773/507	5/ 5
5	774/508	4/ 4
6	772/506	6/ 6
7	775/509	3/ 3
TOTAL	3756/2030	22/ 18