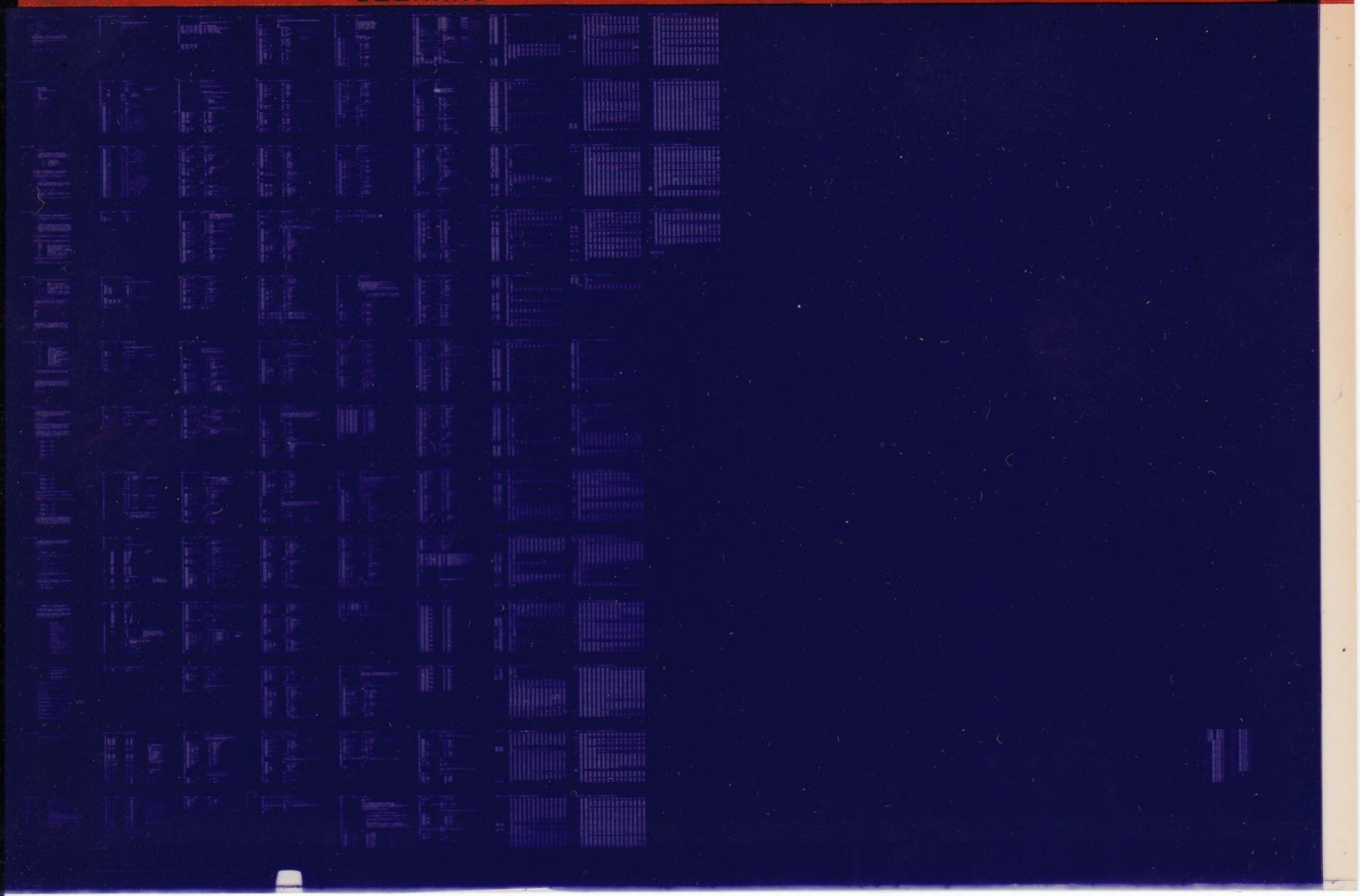
LN01

PRINTER DIAG
CZLNAA0

AH-T344A-MC
FICHE 1 OF 1



B 1

.REM &

IDENTIFICATION

PRODUCT CODE : AC-T343A-MC

PRODUCT NAME: CZLNAAO LN01 PRINTER DIAG

MAINTAINER: SMALL SYSTEMS DIAGNOSTICS

PRODUCT DATE: JAN. 1983

AUTHOR: GLENN A. PERNA

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS DIAGNOSTIC PROGRAM VERIFIES PROPER OPERATION OF THE BASIC LINE PRINTER FUNCTIONS ONLY OF THE LN01 ELECTRONIC PRINTER AND ITS ASSOCIATED M7258 CONTROL UNIT WHICH INTERFACES TO THE PDP-11 CPU. THE BROAD RANGE OF TESTS ASSURES A COMPREHENSIVE TEST OF THE FUNCTIONAL CAPABILITY OF THE PRINTER. THE INDIVIDUAL TESTS ARE IDENTIFIED AS FOLLOWS:

TEST	1	INTERFACE LOGIC
TEST	2	DATA TRANSFER PATHS
TEST	3	PRINTABLE CHARACTERS
TEST	4	NON-PRINTABLE CHARACTERS
TEST	5	PRINT CONTROL
TEST	6	MULTIPLE LINE ADVANCE
TEST	7	OVERSTRIKE TEST
TEST	8	INTERLOCK TEST

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+ OPERATING SYSTEM.

FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL. THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

1.2 SYSTEM REQUIREMENTS

1.2 SYSTEM REQUIREMENTS

A TEST STATION IS REQUIRED CONSISTING OF A PDP-11 CPU WITH A MINIMUM OF 16K WORDS OF MEMORY AND A CONSOLE TERMINAL WITH INTERFACE AT DEVICE ADDRESS 777560. THE SYSTEM ALSO REQUIRES AN XXDP SUPPORTED DEVICE SUCH AS AN RK05/RK11 DISK DRIVE TO AFFORD A MEANS TO LOAD THE DIAGNOSTIC PROGRAM.

1.3 RELATED DOCUMENTS AND STANDARDS

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THIS DIAGNOSTIC IS COMPATIBLE WITH ALL MEMBERS OF THE PDP-11 COMPUTER FAMILY. THE DIAGNOSTIC IS INTERFACED TO THE PDP-11 DIAGNOSTIC SUPERVISOR THROUGH WHICH IT INTERFACES TO THE ENVIRONMENT.

THE DIAGNOSTIC CAN BE USED IN A VARIETY OF OPERATING SYSTEMS TO FULFILL DIFFERENT REQUIREMENTS. THE DIAGNOSTIC CAN BE

LOADED USING XXDP IN A FIELD SERVICE OPERATION, LOADED USING THE APT/ACT/SLIDE DIAGNOSTIC MONITORS IN A MANUFACTURING ENVIRONMENT, OR MANUALLY LOADED USING PAPER TAPE.

THE APPLICABLE PDP-11 CPU, MEMORY, AND PERIPHERALS SHOULD BE RUN TO VALIDATE PROPER OPERATION OF THE SYSTEM BEFORE RUNNING THIS DIAGNOSTIC.

1.5 ASSUMPTIONS

THE PRINTERS UNDER TEST SHOULD HAVE POWER APPLIED AND BE PLACED ON LINE IN READINESS FOR TESTING. EACH LINE PRINTER MUST HAVE ITS OWN M7258 CONTROLLER SET UP AT A DIFFERENT DEVICE ADDRESS. THE DIAGNOSTIC PROVIDES A DEFAULT DEVICE ADDRESS OF 777514 WHICH CAN BE USED WHEN A SINGLE LINE PRINTER IS BEING TESTED OR FOR THE FIRST UNIT WHEN MULTIPLE LINE PRINTERS ARE UNDER TEST. IT WILL BE NECESSARY FOR THE OPERATOR TO RUN THE LINE PRINTER OFF LINE IN THE SELF TEST MODE BEFORE RUNNING THE DIAGNOSTIC. EACH PRINTER SHOULD BE IN THE 8 BIT MODE

2.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES. FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

2.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES (SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".

2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL

SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH.
IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

```
START/TESTS:1-5/PASS:1000/EOP:100
```

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR

CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBR*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXR*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

```
/FLAGS:LOE:IER:BOE
```

2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL). YOU WILL THEN BE ASKED THE FOLLOWING QUESTIONS FOR EACH UNIT.

```
#UNITS (D) ? 1

UNIT 1
LP11 ADDRESS: (O)  (177514) ?
INTERRUPT VECTOR : (O)  (200) ?
```

2.5 SOFTWARE QUESTIONS

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE PARAMETERS. THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC OPERATION MODES. YOU WILL BE PROMPTED BY "CHANGE SW (L) ?" IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING "Y". THE SOFTWARE QUESTIONS AND THE DEFAULT VALUES ARE DESCRIBED IN THE NEXT PARAGRAPH(S).

```
RUN MANUAL INTERVENTION TESTS (N) ?     DEFAULT IS NO
TESTING IN U.S.A. (Y) ?
AUTODROP ERROR COUNT (D) 5 ?            DROPS ANY UNIT FROM TEST WHICH EXCEEDS SPECIFIED NO. OF ERRORS
```

2.6 EXTENDED P-TABLE DIALOGUE

WHEN YOU ANSWER THE HARDWARE QUESTIONS, YOU ARE BUILDING ENTRIES IN A TABLE THAT DESCRIBES THE DEVICES UNDER TEST. THE SIMPLEST WAY TO BUILD THIS TABLE IS TO ANSWER ALL QUESTIONS FOR EACH UNIT TO BE TESTED. IF YOU HAVE A MULTIPLEXED DEVICE SUCH AS A MASS STORAGE CONTROLLER WITH SEVERAL DRIVES OR A COMMUNICATION DEVICE WITH SEVERAL LINES, THIS BECOMES TEDIOUS SINCE MOST OF THE ANSWERS ARE REPETITIOUS.

TO ILLUSTRATE A MORE EFFICIENT METHOD, SUPPOSE YOU ARE TESTING A FICTIONAL DEVICE, THE XY11. SUPPOSE THIS DEVICE CONSISTS OF A CONTROL MODULE WITH EIGHT UNITS (SUB-DEVICES) ATTACHED TO IT. THESE UNITS ARE DESCRIBED BY THE OCTAL NUMBERS 0 THROUGH 7. THERE IS ONE HARDWARE PARAMETER THAT CAN VARY AMONG UNITS CALLED THE Q-FACTOR. THIS Q-FACTOR MAY BE 0 OR 1. BELOW IS A SIMPLE WAY TO BUILD A TABLE FOR ONE XY11 WITH EIGHT UNITS.

```
        # UNITS (D) ?  8<CR>

        UNIT 1
        CSR ADDRESS (O) ?  160000<CR>
        SUB-DEVICE # (O) ?  0<CR>
        Q-FACTOR (O) 0 ?  1<CR>

        UNIT 2
        CSR ADDRESS (O) ?  160000<CR>
        SUB-DEVICE # (O) ?  1<CR>
        Q-FACTOR (O) 1 ?  0<CR>

        UNIT 3
        CSR ADDRESS (O) ?  160000<CR>
        SUB-DEVICE # (O) ?  2<CR>
        Q-FACTOR (O) 0 ?  <CR>

        UNIT 4
        CSR ADDRESS (O) ?  160000<CR>
        SUB-DEVICE # (O) ?  3<CR>
        Q-FACTOR (O) 0 ?  <CR>

        UNIT 5
        CSR ADDRESS (O) ?  160000<CR>
```

```
SUB-DEVICE # (0) ?  4<CR>
Q-FACTOR (0) 0 ?  <CR>

UNIT 6
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  5<CR>
Q-FACTOR (0) 0 ?  <CR>

UNIT 7
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  6<CR>
Q-FACTOR (0) 0 ?  1<CR>

UNIT 8
CSR ADDRESS (0)  160000<CR>
SUB-DEVICE # (0) ?  7<CR>
Q-FACTOR (0) 1 ?  <CR>
```

NOTICE THAT THE DEFAULT VALUE FOR THE Q-FACTOR CHANGES WHEN A NON-DEFAULT RESPONSE IS GIVEN. BE CAREFUL WHEN SPECIFYING MULTIPLE UNITS!

AS YOU CAN SEE FROM THE ABOVE EXAMPLE, THE HARDWARE PARAMETERS DO NOT VARY SIGNIFICANTLY FROM UNIT TO UNIT. THE PROCEDURE SHOWN IS NOT VERY EFFICIENT.

THE RUNTIME SERVICES CAN TAKE MULTIPLE UNIT SPECIFICATIONS HOWEVER. LET'S BUILD THE SAME TABLE USING THE MULTIPLE SPECIFICATION FEATURE.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  0,1<CR>
Q-FACTOR (0) 0 ?  1,0<CR>

UNIT 3
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  2-5<CR>
Q-FACTOR (0) 0 ?  0<CR>

UNIT 7
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  6,7<CR>
Q-FACTOR (0) 0 ?  1<CR>
```

AS YOU CAN SEE IN THE ABOVE DIALOGUE, THE RUNTIME SERVICES WILL BUILD AS MANY ENTRIES AS IT CAN WITH THE INFORMATION GIVEN IN ANY ONE PASS THROUGH THE QUESTIONS. IN THE FIRST PASS, TWO ENTRIES ARE BUILT SINCE TWO SUB-DEVICES AND Q-FACTORS WERE SPECIFIED. THE SERVICES ASSUME THAT THE CSR ADDRESS IS 160000 FOR BOTH SINCE IT WAS SPECIFIED ONLY ONCE. IN THE SECOND PASS, FOUR ENTRIES WERE BUILT. THIS IS BECAUSE FOUR SUB-DEVICES WERE SPECIFIED. THE "-" CONSTRUCT TELLS THE RUNTIME SERVICES TO INCREMENT THE DATA FROM THE FIRST NUMBER TO THE SECOND. IN THIS CASE, SUB-DEVICES

2, 3, 4 AND 5 WERE SPECIFIED. (IF THE SUB-DEVICE WERE SPECIFIED BY ADDRESSES, THE INCREMENT WOULD BE BY 2 SINCE ADDRESSES MUST BE ON AN EVEN BOUNDARY.) THE CSR ADDRESSES AND Q-FACTORS FOR THE FOUR ENTRIES ARE ASSUMED TO BE 160000 AND 0 RESPECTIVELY SINCE THEY WERE ONLY SPECIFIED ONCE. THE LAST TWO UNITS ARE SPECIFIED IN THE THIRD PASS.

THE WHOLE PROCESS COULD HAVE BEEN ACCOMPLISHED IN ONE PASS AS SHOWN BELOW.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0-7<CR>
Q-FACTOR (O) 0 ?  0,1,0,,,,1,1<CR>
```

AS YOU CAN SEE FROM THIS EXAMPLE, NULL REPLIES (COMMAS ENCLOSING A NULL FIELD) TELL THE RUNTIME SERVICES TO REPEAT THE LAST REPLY.

2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. TYPE "R NAME", WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
3. TYPE "START"
4. ANSWER THE "CHANGE HW" QUESTION WITH "Y"
5. ANSWER ALL THE HARDWARE QUESTIONS
6. ANSWER THE "CHANGE SW" QUESTION WITH "N"

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. THESE DEFAULTS ARE DESCRIBED IN SECTIONS 2.3 AND 2.5.

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE "IER" FLAG IS SET (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

```
NAME  TYPE  NUMBER  ON UNIT NUMBER  TST NUMBER PC:XXXXXX
ERROR MESSAGE
```

```
.WHERE; NAME = DIAGNOSTIC NAME
        TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
        NUMBER = ERROR NUMBER
```

UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE "IER" OR "IBR" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE "IER", "IBR" OR "IXR" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

3.2 SPECIFIC ERROR MESSAGES

ERROR	DESCRIPTION
1	"PRINTER ERROR" ERROR CONDITION IN THE PRINTER.
2	"PRINTER NOT READY" PRINTER NOT READY TO ACCEPT DATA.
3	"PRINTER DID NOT INTERRUPT" FAILURE IN INTERFACE LOGIC.
4	"LOADING PRINTER BUFFER DOES NOT CLEAR READY" FAILURE IN INTERFACE LOGIC.
5	"PRINTER INTERRUPTED AT SAME LEVEL AS THE PROCESSOR" FAILURE IN INTERFACE LOGIC.
6	"PRINTER ERROR" ERROR CONDITION IN THE PRINTER.
7	"PRINTER NOT READY" PRINTER NOT READY TO ACCEPT DATA.
8	"PAPER OUT INTERLOCK SWITCH FAILURE" FAULTY INTERLOCK SWITCH
9	"PAPER TRAY HANDLE INTERLOCK SWITCH FAILURE" FAULTY INTERLOCK SWITCH
10	"FRONT DOOR INTERLOCK SWITCH FAILURE" FAULTY INTERLOCK SWITCH
	"NOTE" ERROR MESSAGES #11 THRU #12 HAVE BEEN ELIMINATED
13	"INTERRUPT SERVICING FOR THE FOLLOWING DEVICE DID NOT OCCUR"

GLOBAL ERROR INDICATING INTERRUPT FOR
DATA TRANSFER DID NOT OCCUR.

14 "PRINTER STATUS ERROR"
GLOBAL ERROR INDICATING PRINTER ERROR
CONDITION.

15 "OUTPUT TIMEOUT ERROR"
GLOBAL ERROR INDICATING TRANSMISSION
OF LAST CHARACTER DID NOT OCCUR
WITHIN A GIVEN TIME.

4.0 PERFORMANCE AND PROGRESS REPORTS

PERFORMANCE AND PROGRESS REPORTS ARE NOT SUPPLIED.

5.0 DEVICE INFORMATION TABLES

DEVICE INFORMATION APPEARS IN THE GLOBAL DATA SECTION.

6.0 TEST SUMMARIES

TEST 1
INTERFACE LOGIC
VERIFIES OPERATION OF INTERFACE LOGIC BETWEEN THE PRINTER AND THE CPU.

TEST 2
DATA TRANSFER PATHS
CHECKS THE DATA TRANSFER PATHS FROM THE PRINTER OUTPUT TO
THE PROCESSOR INTERFACE.

TEST 3
PRINTABLE CHARACTERS
CHECKS FOR PROPER PRINTING OF ALL PRINTABLE CHARACTERS.

TEST 4
NON-PRINTABLE CHARACTERS
CHECKS FOR PROPER DETECTION OF ALL NON-PRINTABLE CHARACTERS.

TEST 5
PRINT CONTROL
CHECKS THAT CHARACTERS IN EXCESS OF 132 CHARACTERS ON A LINE
ARE DISREGARDED.

TEST 6
MULTIPLE LINE ADVANCE
CHECKS THE MULTIPLE LINE ADVANCE FOR PROPER PAPER MOVEMENT.

TEST 7
OVERSTRIKE
THIS TEST CHECKS THE MACHINE'S OVERSTRIKE CAPABILITY AS WELL AS PAGE
BUFFER AND LINE BUFFER LIMITATIONS.

```
                                                                M 1
INTERLOCK TEST 8
THIS TEST MAKES SURE THAT THE ERROR BIT IN THE PRINTER
INTERFACE IS SET WHENEVER INTERLOCK SWITCHES ARE TRIPPED
IN THE PRINTER.
                                                        &
```

```
                                .TITLE  CZLNAAO LN01 DIAGNOSTIC
                                .ENABL AMA
                                .SBTTL  IDENTIFICATION

                                ; PRODUCT CODE:         AC-T343A-MC

                                ; PRODUCT NAME:         CZLNAAO LN01 DIAG

                                ; MAINTAINER:           SMALL SYSTEMS DIAGNOSTICS

                                ; AUTHORS:              GLENN A. PERNA

                                ; DATE                  JAN     1983

                                ;COPYRIGHT (C) 1983,  BY
                                ;DIGITAL EQUIPMENT CORPORATION, MAYNARD MASSACHUSSETTS 01754
                                ;
                                ; THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON  A
                                ; SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH THE INCLU-
                                ; SION OF THE ABOVE COPYRIGHT NOTICE.  THIS  SOFTWARE,  OR  ANY
                                ; OTHER  COPIES  THEREOF, MAY NOT BE PROVIDED OR OTHERWISE MADE
                                ; AVAILABLE TO ANY OTHER PERSON EXCEPT FOR USE ON  SUCH  SYSTEM
                                ; AND  TO  ONE WHO AGREES TO THESE LICENSE TERMS.  TITLE TO AND
                                ; OWNERSHIP OF THE SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.
                                ;
                                ; THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT
                                ; NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
                                ; EQUIPMENT CORPORATION.
                                ;
                                ; DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR  RELIABILITY  OF
                                ; ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.
```

```
                                ;++
                                ; FUNCTIONAL DESCRIPTION
                                ;
                                ; THIS DIAGNOSTIC PROGRAM VERIFIES PROPER OPERATION OF THE LN01
                                ; LINE PRINTER, AND IT'S ASSOCIATED INTERFACE MODULE.
                                ;A TOTAL OF 16 PRINTERS CAN BE TESTED.
                                ;
                                ; THE PROGRAM CONSISTS OF T-B-S TESTS,
                                ;
                                ; THE PROGRAM IS COMPATIBLE TO THE PDP-11 DIAGNOSTIC SUPERVISOR, ACT/SLIDE, AND
                                ; XXDP+.
                                ;
                                ;--



                                ; VERSION        A-0    JAN  1983               GLENN A. PERNA

                                ; HISTORY       REV. A-0        INITIAL RELEASE
```

```
                                        .TITLE  CZLNAA0 LN01 TEST
                                .SBTTL PROGRAM HEADER

                                        .MCALL  SVC
000000'                                 SVC                             ;INITIALIZE SUPERVISOR MACROS
                                        .MCALL  STRUCT
000000'                                 STRUCT                          ;STRUCTURED MACRO PACKAGE
        000000                          $LSTIN=         0               ; LIST ASSY CODE LEFT
        000000                          $LSTTAG=        0               ; LIST TAGS LEFT
        177777                          $LOCTAG=        -1

        000000                  SVCINS= 0       ;LIST INSTRUCTIONS
        000000                  SVCTST= 0       ;LIST TEST TAGS
        000000                  SVCSUB= 0       ;LIST SUBTEST TAGS
        000000                  SVCGBL= 0       ;LIST GLOBAL TAGS
        000000                  SVCTAG= 0       ;LIST OTHER TAGS

                                .ENABL AMA
                                .ENABL ABS
                                .ENABL LC
        002000                          .=2000

002000                                  BGNMOD
002000                                  POINTER BGNSW,BGNSFT

002000                                  HEADER  CZLNA,A,0,60,1,340
(4) 002000              L$NAME::                ;DIAGNOSTIC NAME
(4) 002000      103                     .ASCII /C/
(4) 002001      132                     .ASCII /Z/
(4) 002002      114                     .ASCII /L/
(4) 002003      116                     .ASCII /N/
(4) 002004      101                     .ASCII /A/
(6) 002005      000                     .BYTE   0
(6) 002006      000                     .BYTE   0
(5) 002007      000                     .BYTE   0
(5) 002010              L$REV::                 ;REVISION LEVEL
(4) 002010      101                     .ASCII  /A/
(5) 002011              L$DEPO::                ;0
(4) 002011      060                     .ASCII  /0/
(5) 002012              L$UNIT::                ;NUMBER OF UNITS
(4) 002012  000000                      .WORD   0
(5) 002014              L$TIML::                ;LONGEST TEST TIME
(4) 002014  000060                      .WORD   60
(5) 002016              L$HPCP::                ;POINTER TO H.W. QUES.
(4) 002016  022140                      .WORD   L$HARD
(5) 002020              L$SPCP::                ;POINTER TO S.W. QUES.
(4) 002020  022220                      .WORD   L$SOFT
(5) 002022              L$HPTP::                ;PTR. TO DEF. H.W. PTABLE
(4) 002022  002222                      .WORD   L$HW
(5) 002024              L$SPTP::                ;PTR. TO S.W. PTABLE
(4) 002024  002234                      .WORD   L$SW
(5) 002026              L$LADP::                ;DIAG. END ADDRESS
(4) 002026  022370                      .WORD   L$LAST
(5) 002030              L$STA::                 ;RESERVED FOR APT STATS
(4) 002030  000000                      .WORD   0
(5) 002032              L$CO::
```

```
(4)  002032  000000                    .WORD   0
(5)  002034                     L$DTYP::        ;DIAGNOSTIC TYPE
(4)  002034  000001                    .WORD   1
(5)  002036                     L$APT::         ;APT EXPANSION
(4)  002036  000000                    .WORD   0
(5)  002040                     L$DTP::         ;PTR. TO DISPATCH TABLE
(4)  002040  002132                    .WORD   L$DISPATCH
(5)  002042                     L$PRIO::        ;DIAGNOSTIC RUN PRIORITY
(4)  002042  000340                    .WORD   340
(5)  002044                     L$ENV1::        ;FLAGS DESCRIBE HOW IT WAS SETUP
(4)  002044  000000                    .WORD   0
(5)  002046                     L$EXP1::        ;EXPANSION WORD
(4)  002046  000000                    .WORD   0
(5)  002050                     L$MREV::        ;SVC REV AND EDIT #
(4)  002050     003                    .BYTE   C$REVISION
(3)  002051     003                    .BYTE   C$EDIT
(5)  002052                     L$EF::          ;DIAG. EVENT FLAGS
(4)  002052  000000                    .WORD   0
(5)  002054  000000                    .WORD   0
(5)  002056                     L$SPC::
(4)  002056  000000                    .WORD   0
(5)  002060                     L$DEVP::        ; POINTER TO DEVICE TYPE LIST
(4)  002060  002212                    .WORD   L$DVTYP
(5)  002062                     L$REPP::        ;PTR. TO REPORT CODE
(4)  002062  000000                    .WORD   0
(5)  002064                     L$EXP4::
(4)  002064  000000                    .WORD   0
(5)  002066                     L$EXP5::
(4)  002066  000000                    .WORD   0
(5)  002070                     L$AUT::         ;PTR. TO ADD UNIT CODE
(4)  002070  000000                    .WORD   0
(5)  002072                     L$DUT::         ;PTR. TO DROP UNIT CODE
(4)  002072  000000                    .WORD   0
(5)  002074                     L$LUN::         ;LUN FOR EXERCISERS TO FILL
(4)  002074  000000                    .WORD   0
(5)  002076                     L$DESP::        ;POINTER TO DIAG. DESCRIPTION
(4)  002076  002152                    .WORD   L$DESC
(5)  002100                     L$LOAD::        ;GENERATE SPECIAL AUTOLOAD EMT
(4)  002100  104035                    EMT     E$LOAD
(5)  002102                     L$ETP::         ;POINTER TO ERRTBL
(4)  002102  000000                    .WORD   0
(5)  002104                     L$ICP::         ;PTR. TO INIT CODE
(4)  002104  005456                    .WORD   L$INIT
(5)  002106                     L$CCP::         ;PTR. TO CLEAN-UP CODE
(4)  002106  007232                    .WORD   L$CLEAN
(5)  002110                     L$ACP::         ;PTR. TO AUTO CODE
(4)  002110  002226                    .WORD   L$AUTO
(5)  002112                     L$PRT::         ;PTR. TO PROTECT TABLE
(4)  002112  002122                    .WORD   L$PROT
(5)  002114                     L$TEST::        ;TEST NUMBER
(4)  002114  000000                    .WORD   0
(5)  002116                     L$DLY::         ;DELAY COUNT
(4)  002116  000000                    .WORD   0
(5)  002120                     L$HIME::        ;PTR. TO HIGH MEM
(4)  002120  000000                    .WORD   0
681
```

```
                                        ;
                                        ; THE FOLLOWING IS A LOAD PROTECTION TABLE
                                        ;
                                                BGNPROT
                                        L$PROT::
                                                .WORD   0
                                                .WORD   -1
                                                .WORD   -1
                                                ENDPROT
```

```
                                        .SBTTL  DISPATCH TABLE

                                        ;++
                                        ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                                        ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                                        ;--

002130                                          DISPATCH        8               ;X= NUMBER OF TESTS
(4) 002130  000010                              .WORD   8
(3) 002132                              L$DISPATCH::
(6) 002132  007426                              .WORD   T1
(6) 002134  010722                              .WORD   T2
(6) 002136  011356                              .WORD   T3
(6) 002140  012020                              .WORD   T4
(6) 002142  013050                              .WORD   T5
(6) 002144  014302                              .WORD   T6
(6) 002146  014770                              .WORD   T7
(6) 002150  016522                              .WORD   T8

                                        ;FOR USE ON REVISION C OF THE SUPERVISOR
                                        ;
002152                                          DESCRIP         <CZLNAAO LINE PRINTER DIAGNOSTIC>
(4) 002152                              L$DESC::
(3) 002152  055103  047114  040501              .ASCIZ  /CZLNAAO LINE PRINTER DIAGNOSTIC/
(3) 002160  020060  044514  042516
(3) 002166  050040  044522  052116
(3) 002174  051105  042040  040511
(3) 002202  047107  051517  044524
(3) 002210  000103
(2)                                             .EVEN
002212                                          DEVTYP          <LN01>
(4) 002212                              L$DVTYP::
(3) 002212  047114  030460      000             .ASCIZ  /LN01/
(2)         002220                              .EVEN
```

```
                                .SBTTL  DEFAULT HARDWARE P-TABLE

                                ;++
                                ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                                ; THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
                                ; IS IDENTICAL TO THE RUN-TIME P-TABLE.
                                ;--

002220                                  BGNHW   DFPTBL
(3) 002220  000002                      .WORD   L10001-L$HW/2
(3) 002222                      L$HW::
(3) 002222                      DFPTBL::
002222  177514                          .WORD   177514                  ;LP11 REGISTER ADDRESS
002224  000200                          .WORD   200                     ;LP11 INTERRUPT VECTOR

                                ;INTERRUPT VECTOR PRIORITY IS 4 AND CANNOT BE CHANGED


002226                                  ENDHW
(3) 002226                      L10001:


002226                                  BGNAUTO
(3) 002226                      L$AUTO::

002226  000240                          NOP             ; NOT USED

002230                                  ENDAUTO
(3) 002230                      L10002:
(3) 002230  104461                      TRAP    C$AUTO
```

```
                                .SBTTL  SOFTWARE P-TABLE

                                ;++
                                ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                                ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                                ;--

002232                                  BGNSW   SFPTBL
(3) 002232  000002                      .WORD   L10003-L$SW/2
(3) 002234                      L$SW::
(3) 002234                      SFPTBL::

002234  000000                  INHINT: .WORD   0                       ;0 IF NO INTERVENTION TESTS
                                                                        ;1 IF MANUAL INTERVENTION TESTS
                                                                        ;DEFAULT IS NO

002236  000005                  MAXERR: .WORD   5                       ; AUTODROP ERROR COUNT
                                ; IF ERROR COUNT EXCEEDS MAXERR THE UNIT WILL BE DROPPED FROM TEST

002240                                  ENDSW
(3) 002240                      L10003:
```

```
                .SBTTL  I/O MACRO DEFINITIONS

                .MACRO  OUTPUT  ADD,BFCNT,ERR,PRINTS
                        MOV     ADD,BUFADD              ;SAVE THE BUFFER ADDRESS
                        MOV     BFCNT,BUFCNT            ;BUFFER BYTE COUNT BFCNT
                        MOV     #-1,PRINTR              ; OUTPUT TO ALL UNITS
                .IF B   ERR
                        MOV     #LPERR,ERRSVC
                .ENDC
                .IF NB  ERR
                        MOV     ERR,ERRSVC
                .ENDC
                .IF B PRINTS
                        MOV     #1,BUFREP               ; PRINT ONCE DEFAULT
                .ENDC
                .IF NB PRINTS
                        MOV     PRINTS,BUFREP           ; SUPPLY PRINT COUNT
                .ENDC
                        JSR     PC,IOCTRL               ;CALL THE DRIVER
                .ENDM   OUTPUT


                .MACRO  OUTPUI  ADD,BFCNT,ERR,UNIT,PRINTS
                        MOV     ADD,BUFADD              ;SAVE BUFFER ADDRESS
                        MOV     BFCNT,BUFCNT            ;BUFFER BYTE COUNT BFCNT
                .IF B   ERR
                        MOV     #LPERR,ERRSVC
                .ENDC
                .IF NB  ERR
                        MOV     ERR,ERRSVC
                .ENDC
                .IF B PRINTS
                        MOV     #1,BUFREP               ; PRINT ONCE DEFAULT
                .ENDC
                .IF NB PRINTS
                        MOV     PRINTS,BUFREP           ; SUPPLY PRINT COUNT
                .ENDC
                        MOV     UNIT,PRINTR             ; SUPPLY UNIT NUMBER
                        JSR     PC,IOCTRL               ;CALL THE DRIVER
                .ENDM


                ; PRINTS IS A PARAMETER CONTROLLING THE NUMBER IF TIMES THE DATA OR
                ; MESSAGE IS TO BE PRINTED (SENT TO THE PRINTER). DEFAULT IS 1.
                ;
                ; A TIMEOUT OF 20. SECONDS IS FURNISHED BASED ON THE FOLLOWING ASSUMPTIONS :
                ;       1   A PRINTER SPEED OF 300 LPM
                ;       2   A REPEAT COUNT OF 88 MAX. ( 1 PAGE OF LINES AT 8 LPI. )
                ;       3   AN INITIAL BAND STARTUP TIME OF 2.5 SECONDS.
                ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
002240                  ENDMOD
```

```
                                .SBTTL  GLOBAL AREAS

002240                                  BGNMOD

                                ;++
                                ; THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES
                                ; THAT ARE USED IN MORE THAN ONE TEST.
                                ;--

002240                                  EQUALS
(1)                             ;
(1)                             ; BIT DIFINITIONS
(1)                             ;
(1)             100000          BIT15== 100000
(1)             040000          BIT14== 40000
(1)             020000          BIT13== 20000
(1)             010000          BIT12== 10000
(1)             004000          BIT11== 4000
(1)             002000          BIT10== 2000
(1)             001000          BIT09== 1000
(1)             000400          BIT08== 400
(1)             000200          BIT07== 200
(1)             000100          BIT06== 100
(1)             000040          BIT05== 40
(1)             000020          BIT04== 20
(1)             000010          BIT03== 10
(1)             000004          BIT02== 4
(1)             000002          BIT01== 2
(1)             000001          BIT00== 1
(1)                             ;
(1)             001000          BIT9==  BIT09
(1)             000400          BIT8==  BIT08
(1)             000200          BIT7==  BIT07
(1)             000100          BIT6==  BIT06
(1)             000040          BIT5==  BIT05
(1)             000020          BIT4==  BIT04
(1)             000010          BIT3==  BIT03
(1)             000004          BIT2==  BIT02
(1)             000002          BIT1==  BIT01
(1)             000001          BIT0==  BIT00
(1)                             ;
(1)                             ; EVENT FLAG DEFINITIONS
(1)                             ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
(1)                             ;
(1)             000040          EF.START==      32.             ; START COMMAND WAS ISSUED
(1)             000037          EF.RESTART==    31.             ; RESTART COMMAND WAS ISSUED
(1)             000036          EF.CONTINUE==   30.             ; CONTINUE COMMAND WAS ISSUED
(1)             000035          EF.NEW==        29.             ; A NEW PASS HAS BEEN STARTED
(1)             000034          EF.PWR==        28.             ; A POWER-FAIL/POWER-UP OCCURRED
(1)                             ;
(1)                             ;
(1)                             ; PRIORITY LEVEL DEFINITIONS
(1)                             ;
(1)             000340          PRI07== 340
(1)             000300          PRI06== 300
(1)             000240          PRI05== 240
```

```
(1)     000200          PRI04== 200
(1)     000140          PRI03== 140
(1)     000100          PRI02== 100
(1)     000040          PRI01== 40
(1)     000000          PRI00== 0
(1)                     ;
(1)                     ;OPERATOR FLAG BITS
(1)                     ;
(1)     000004          EVL==         4
(1)     000010          LOT==        10
(1)     000020          ADR==        20
(1)     000040          IDU==        40
(1)     000100          ISR==       100
(1)     000200          UAM==       200
(1)     000400          BOE==       400
(1)     001000          PNT==      1000
(1)     002000          PRI==      2000
(1)     004000          IXE==      4000
(1)     010000          IBE==     10000
(1)     020000          IER==     20000
(1)     040000          LOE==     40000
(1)     100000          HOE==    100000
817
821     000012          LF==12
822     000014          FF==14
823     000015          CR==15
824     000177          DEL==177
825                     ;
826                     ;GLOBAL ERROR CODES FOR USE BY GENERAL ERROR ROUTINE
827                     ;
828     000001          STATER= 1                       ;TRANSMITTER STATUS ERROR IN OUTPUT
829     000002          TIMOUT= 2                       ;TIMEOUT ERROR IN IO DRIVER MODULE
830                                                     ;THIS ERROR INDICATES THE LAST CHARACTER
831                                                     ;WAS NOT TRANSMITTED WITHIN A GIVEN TIME
832     000003          NOINTR= 3                       ;GROSS TIME OUT ERROR. THE SPECIFIED DID NOT
833                                                     ;INTERRRUPT. THEREFORE IO DRIVER MODULE WAS
834                                                     ;NOT CALLED
835                     ;
836                     .SBTTL  GENERAL REGISTER USAGE DEFINITIONS
837                     ;
838                     ;R0     RESERVED FOR USE BY THE MACRO PACKAGES
839                     ;R1     MAXIMUM NUMBER OF UNITS TO TEST L$UNIT-1
840                     ;R2     UNIT NUMBER BY 2. USED TO CALCULATE OFFSET INTO PROPER
841                     ;       PRINTER TABLE
842                     ;R3     TEMPORARY STORAGE
843                     ;R4          "       "
844                     ;R5
845                     ;R6     STACK POINTER
846                     ;R7     PROGRAM COUNTER
847                     ;
848
849
850                     ;
851                     ; LP STATUS TABLE BIT DEFINITIONS
852                     ;
853     100000          ERROR = BIT15
```

```
                        040000          DROPED = BIT14
                        020000          ACTIVE = BIT13
                        000377          LOBYTE = 377    ; BIT MASK FOR CLEARING LOBYTE (COUNTER)
```

```
                                .SBTTL  GLOBAL DATA SECTION

002240  000000          FLAG:   .WORD   0               ;<CR> FLAG FOR USE BY SUPERVISOR
002242  000000          LINCNT: .WORD   0               ;LINE COUNTER
002244  000000          LSTCNT: .WORD   0
002246  000000          COUNT:  .WORD   0
002250  000000          CCNT:   .WORD   0
002252  000000          STRCNT: .WORD   0
002254  000000          CHRGEN: .WORD   0
002256  000000          UNIT:   .WORD   0               ;UNIT COUNTER FOR SINGLE UNIT TESTING
002260  000000          LUNIT:  .WORD   0               ;UNIT COUNTER FOR ERRORS
                                                        ;AND TESTS NOT USING THE OUTPUT
                                                        ;MACROS.
002262  000000          PTABAD: .WORD   0               ;P-TABLE ADDRESS RETURNED BY GPHARD
002264  000000          PRINTR: .WORD   0               ;SELECTED LINE NO.
                                                        ;MACRO
002266  000000          CLKTYP: .WORD   0               ;CLOCK TYPE CONTROL WORD
                                                        ;1= NO CLOCK AVAILABLE
                                                        ;2= KW11-L LINE CLOCK
                                                        ;3= KW11-P PROGRAMABLE CLOCK
002270  000000          CLOCKP: .WORD   0               ; CLOCK P-TABLE ADDRESS
002272  000000          CLKCSR: .WORD   0               ;CLOCK CSR ADDRESS
002274  000000          CLKSET: .WORD   0               ; CLOCK TIME SET REG ADDRESS
002276  000000          CLKVEC: .WORD   0               ;CLOCK VECTOR ADDRESS
002300  000000          CLKENA: .WORD   0               ;CLOCK ENABLE BITS
002302  000000          ERRCOD: .WORD   0               ;ERROR CODE TYPE FOR GENERAL
                                                        ;ERROR ROUTINE
002304  000000          ERRFLG: .WORD   0               ;EXPECTED ERROR INDICATOR
002306  000000          UUT:    .WORD   0               ; # UNITS ACTUALLY UNDER TEST
                                                        ;EXITS BACK TO IO DRIVER EQUAL
                                                        ;1 IF ERROR WAS EXPECTED.

002310  000000          INDEX:  .WORD   0
002312  000000          VFUCMD: .WORD   0
                        ;
                        ;MACRO  VARIABLES

002314  000000          BUFADD: .WORD   0               ;BUFFER ADDRESS OF DATA TO BE SENT
                                                        ;TO THE PRINTER
002316  000000          BUFCNT: .WORD   0               ;NUMBER OF BYTES TO TRANSFER

002320  000000          BUFREP: .WORD   0               ; NUMBER OF TIMES TO PRINT
                        ;
                        ;
                        ;
                        ;LN01 PARAMETER WORD TABLES
                        ;
002322  000020          LPCSR:  .REPT   16.             ; ADDRESS OF CSR FOR EACH LP11
                                .WORD   0
                                .ENDR
002362  000016          LPVEC:  .REPT   16              ; INTERRUPT VECTOR ADDRESS
                                .WORD   0
                                .ENDR
002416  000020          LPBUF:  .REPT 16.               ; DATA BUFFER REGISTER ADDRESS
```

```
                                        .WORD   0
                                        .ENDR
002456  000020                  STATUS: .REPT 16.               ; UNIT STATUS
                                        .WORD   0
                                        .ENDR
002516  000020                  CURADD: .REPT 16.               ; CURRENT ADDRESS OF OUTPUT DATA BYTE
                                        .WORD   0
                                        .ENDR
002556  000020                  MSGCNT: .REPT 16.               ; INITIAL BYTE COUNT OF MSG FOR REPEAT RESTORE
                                        .WORD 0
                                        .ENDR
002616  000020                  REPCNT: .REPT   16.             ; NO. OF TIMES TO REPEAT MESSAGE
                                        .WORD   0
                                        .ENDR
002656  000020                  MSGADR: .REPT   16.             ; ADDRESS OF DATA TO PRINT START OF DATA
                                        .WORD   0
                                        .ENDR
002716  000020                  CURCNT: .REPT 16.               ; CURRENT COUNT REMAINING TO OUTPUT
                                        .WORD   -1
                                        .ENDR
002756  000020                  LFINTR: .REPT 16.               ; INTERRUPT ROUTINE ADDRESS
                                        .WORD   0
                                        .ENDR
003016  000020                  DELCNT: .REPT 16.
                                        .WORD   0               ; TIMEOUT DELAY COUNTER
                                        .ENDR
003056  000000                  ERRSVC: .WORD   0               ; ERROR ROUTINE DISPATCH ADDRESS
003060  000020                  ERRTBL::        .REPT 16.       ; ERROR COUNT FOR EACH UNIT
                                        .WORD   0
                                        .ENDR

003120  000000                  WORK::  .WORD   0               ; WORK AREA
003122  000000                  WORK1:  .WORD   0


                                .SBTTL  OUTPUT BUFFER
                                ;
                                ;150 BYTES IS RESERVED FOR THE OUTPUT BUFFER AREA
                                ;

                                        .EVEN
003124  000226                  OUTBUF: .REPT   150.
                                        .BYTE   0
                                        .ENDR
```

```
                                        .SBTTL  GLOBAL TEXT SECTION

                                        .NLIST BEX
                                        .++
                                        ; THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                        ; MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                                        ; MORE THAN ONE TEST.
                                        .--
003352  051120  047111  042524  CSRERR: .ASCIZ /PRINTER ERROR/
003370  051120  047111  042524  RDYERR: .ASCIZ /PRINTER NOT READY/
003412  040520  042520  020122  PAPSWI: .ASCIZ /PAPER OUT INTERLOCK SWITCH FAILURE/
003455     120  050101  051105  HANSWI: .ASCIZ /PAPER TRAY HANDLE INTERLOCK SWITCH FAILURE/
003530  051106  047117  020124  DOOSWI: .ASCIZ /FRONT DOOR INTERLOCK SWITCH FAILURE/
003574  051124  047101  046523  INTER1: .ASCIZ  /TRANSMIT INTERRUPT TIMEOUT/
003627     120  044522  052116  TXERR:  .ASCIZ  /PRINTER STATUS ERROR/
003654  052517  050124  052125  OUTTIM: .ASCIZ  /OUTPUT TIMEOUT ERROR/
003701     125  044516  020124  TXNOIN: .ASCIZ  /UNIT FAILED TO INTERRUPT/
003732  046101  020114  047125  UUTEQ0: .ASCIZ  /ALL UNITS HAVE BEEN DROPPED..RESTART../
        004002                  .EVEN


                                        ;
                                        ;

                                        .LIST BEX
                                        ;
                                        ; FORMAT STATEMENTS USED IN PRINT CALLS
                                        ;
004002  040445  050114  030461  LPDROP: .ASCIZ  /%ALP11 UNIT %D2%A DROPPED FROM TEST%N/
004010  052440  044516  020124
004016  042045  022462  020101
004024  051104  050117  042520
004032  020104  051106  046517
004040  052040  051505  022524
004046  000116
```

```
                                .SBTTL  GLOBAL SUBROUTINES  SECTION

                                ;++
                                ; THE GLOBAL SUBROUTINE SECTION CONTAINS THE SUBROUTINES
                                ; THAT ARE USED BY MORE THAN ONE TEST.
                                ;--


                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;    SUBROUTINE TO PRINT THE GENERAL ERROR INFORMATION.
                                ;       PRINTS THE ERROR MESSAGE IN THE FOLLOWING FORMAT:
                                ;
                                ;       "ERROR AT CSR XXXXXX UNIT YY"
                                ;
                                ;       WHERE XXXXXX=   DEVICE CSR ADDRESS
                                ;                 YY=   UNIT NUMBER THAT FAILED
                                ;
                                ;       CALLING SEQUENCE
                                ;               JSR PC,LPERR
                                ;       REQUIRED PARAMETERS
                                ;               ERRCOD MUST BE SET TO ONE OF THE ERROR CODES DESCRIBED
                                ;               UNDER ERROR CODES.
                                ;
                                ;--
                                ;
                                ;
                                ;               R2 IS USED INTERNAL TO THE ROUTINE.
                                ;               THE ROUTINE DOES A SAVE ON R2
                                ;               AND RESTORES IT PRIOR TO EXITING.
                                ;

004050                          LPERR:  SELECT ERRCOD OF 3 VERIFY       ;SELECT PROPER MESSAGE FORMAT
(2)  004050  013746  002302             MOV     ERRCOD,-(SP)
(6)  004054  002455                     BLT     50005$
(3)  004056  023727  002302  000003     CMP     ERRCOD,#3
(7)  004064  003051                     BGT     50005$
(2)  004066  006316                     ASL     (SP)
(3)  004070  062716  004076             ADD     #50000$,(SP)
(2)  004074  013607                     MOV     @(SP)+,PC
(3)  004076                     50000$:
(5)  004076  004214                     .WORD   50004$
(5)  004100  004106                     .WORD   50003$
(5)  004102  004134                     .WORD   50002$
(5)  004104  004162                     .WORD   50001$

004106                                  CASE 1                  ;STATUS ERROR
(5)  004106                     50003$:
004106                                  LET ERRTBL(R2) := ERRTBL(R2) + #1
(7)  004106  005262  003060             INC     ERRTBL(R2)
004112                                  LET L$LUN := R2 SHIFT -1
(5)  004112  010237  002074             MOV     R2,L$LUN
(8)  004116  006237  002074             ASR     L$LUN
004122                                  ERRHRD  14,TXERR
(4)  004122  104456                     TRAP    C$ERHRD
(5)  004124  000016                     .WORD   14
```

```
   (5)  004126  003627                              .WORD   TXERR
   (5)  004130  000000                              .WORD   0
  1036
  1037  004132                                      CASE 2                  ;OUTPUT TIMEOUT ERROR
   (4)  004132  000430                              BR      50006$
   (5)  004134                              50002$:
  1038  004134                                      LET ERRTBL(R2) := ERRTBL(R2) + #1
   (7)  004134  005262  003060                      INC     ERRTBL(R2)
  1039  004140                                      LET L$LUN := R2 SHIFT -1
   (5)  004140  010237  002074                      MOV     R2,L$LUN
   (8)  004144  006237  002074                      ASR     L$LUN
  1040  004150                                      ERRHRD  15,OUTTIM       ;
   (4)  004150  104456                              TRAP    C$ERHRD
   (5)  004152  000017                              .WORD   15
   (5)  004154  003654                              .WORD   OUTTIM
   (5)  004156  000000                              .WORD   0
  1041
  1042  004160                                      CASE 3
   (4)  004160  000415                              BR      50006$
   (5)  004162                              50001$:
  1043                                                      ;  NEVER RECIEVED THE INTERRUPT
  1044  004162                                              LET ERRTBL(R2) := ERRTBL(R2) + #1
   (7)  004162  005262  003060                      INC     ERRTBL(R2)
  1045  004166                                              LET L$LUN := R2 SHIFT -1
   (5)  004166  010237  002074                      MOV     R2,L$LUN
   (8)  004172  006237  002074                      ASR     L$LUN
  1046  004176                                              ERRHRD 16,TXNOIN
   (4)  004176  104456                              TRAP    C$ERHRD
   (5)  004200  000020                              .WORD   16
   (5)  004202  003701                              .WORD   TXNOIN
   (5)  004204  000000                              .WORD   0
  1047
  1048
  1049
  1050  004206                              ENDSELECT
   (3)  004206  000402                              BR      50006$
   (3)  004210                              50005$:
   (2)  004210  062706  000002                      ADD     #2,SP
   (3)  004214                              50004$:
   (3)  004214                              50006$:
  1051
  1052  004214                                      IF ERRTBL(R2) GT MAXERR THEN
   (6)  004214  026237  003060  002236              CMP     ERRTBL(R2),MAXERR
  (10)  004222  003402                              BLE     50007$
  1053  004224  004737  005314                          JSR PC,DROPIT       ; MAXIMUM ERROR COUNT EXCEEDED !
  1054  004230                                      ENDIF
   (4)  004230                              50007$:
  1055  004230                                      LET STATUS(R2) :- STATUS(R2) CLR.BY #ERROR
   (7)  004230  042762  100000  002456              BIC     #ERROR,STATUS(R2)
  1056  004236                                      LET ERRCOD := #0
   (4)  004236  005037  002302                      CLR     ERRCOD
  1057  004242                                      LET @LPCSR(R2) := #100          ; CLEAR THE ERROR BIT AND ENABLE INTERRUPTS
   (4)  004242  012772  000100  002322              MOV     #100,@LPCSR(R2)
  1058  004250  000207                              RTS     PC                      ;AND EXIT
  1059
  1060                                      ;==============================================
```

```
1061                                    ; BIN2DA        BINARY TO DECIMAL ASCII CONVERSION ROUTINE
1062                                    ;               ENTER WITH NUMBER TO BE CONVERTED ON THE STACK
1063                                    ;               FOLLOWED BY THE ADDRESS OF A 5 BYTE BUFFER
1064                                    ;               FOR THE ASCII STRING. 5 DIGITS WILL BE CONVERTED
1065                                    ;               LEADING ZEROES WILL BE CONVERTED TO SPACES.
1066                                    ;               CALL BY JSR  PC,BIN2DA
1067                                    ;=============================================================
1068
1069  004252                            BIN2DA: PUSH R4,R5
 (2)  004252  010446                            MOV     R4,-(SP)
 (3)  004254  010546                            MOV     R5,-(SP)
1070  004256                                    LET R4 := 6(SP)         ; GET ADDRESS FOR ASCII STRING
 (4)  004256  016604  000006                    MOV     6(SP),R4
1071  004262                                    LET R5 := #TABLDA       ; GET ADDRESS OF DECIMAL TABLE
 (4)  004262  012705  004444                    MOV     #TABLDA,R5
1072  004266                                    LET FLAGDA := #0        ; LEADING ZERO FLAG
 (4)  004266  005037  004456                    CLR     FLAGDA
1073  004272                                    LET COUNTD := #0
 (4)  004272  005037  004460                    CLR     COUNTD
1074                                            ; 8.(SP) HAS NUMBER TO BE CONVERTED
1075  004276                                    DECR DIGITS FROM #4 TO #0 BY #1 ; DO 5 DIGITS
 (5)  004276  012737  000004  004462            MOV     #4,DIGITS
 (7)  004304  000402                            BR      50010$
 (6)  004306                            50011$:
(10)  004306  005337  004462                    DEC     DIGITS
 (7)  004312                            50010$:
 (7)  004312  005737  004462                    TST     DIGITS
 (9)  004316  002435                            BLT     50012$
1076  004320                                        WHILE 8.(SP) GE (R5) DO     ; CREATE A DIGIT
 (4)  004320                            50013$:
 (6)  004320  026615  000010                    CMP     8.(SP),(R5)
(10)  004324  002405                            BLT     50014$
1077  004326                                    LET 8.(SP) := 8.(SP) - (R5)
 (7)  004326  161566  000010                    SUB     (R5),8.(SP)
1078  004332                                    LET COUNTD := COUNTD + #1
 (7)  004332  005237  004460                    INC     COUNTD
1079  004336                                ENDDO
 (4)  004336  000770                            BR      50013$
 (3)  004340                            50014$:
1080                                        ; CONVERT DIGIT TO ASCII OR SUPPLY A SPACE
1081  004340                                IF COUNTD GT #0 OR FLAGDA GT #0 THEN
 (6)  004340  005737  004460                    TST     COUNTD
 (8)  004344  003003                            BGT     50015$
 (6)  004346  005737  004456                    TST     FLAGDA
(10)  004352  003410                            BLE     50016$
 (6)  004354                            50015$:
1082  004354                                    LET COUNTD := COUNTD SET.BY #60
 (7)  004354  052737  000060  004460            BIS     #60,COUNTD
1083  004362                                    LET (R4)+ :B= COUNTD
 (4)  004362  113724  004460                    MOVB    COUNTD,(R4)+
1084  004366                                    LET FLAGDA := FLAGDA + #1
 (7)  004366  005237  004456                    INC     FLAGDA
1085  004372                                ELSE
 (4)  004372  000402                            BR      50017$
 (3)  004374                            50016$:
1086  004374                                    LET (R4)+ :B= #40
```

```
 (4)  004374  112724  000040                    MOVB    #40,(R4)+
1087  004400                                        ENDIF
 (4)  004400                              50017$:
1088                                               ; DO THE NEXT DIGIT
1089  004400                                       LET R5 := R5 + #2
 (7)  004400  062705  000002                    ADD     #2,R5
1090  004404                                       LET COUNTD := #0
 (4)  004404  005037  004460                    CLR     COUNTD
1091  004410                                     ENDDECR
 (5)  004410  000736                            BR      50011$
 (4)  004412                              50012$:
1092                                               ; IF NUMBER WAS A ZERO PRINT A '0'
1093  004412                                       IF FLAGDA EQ #0 THEN
 (6)  004412  005737  004456                    TST     FLAGDA
 (10) 004416  001002                            BNE     50020$
1094  004420                                       LET -(R4) :B= #60
 (4)  004420  112744  000060                    MOVB    #60,-(R4)
1095  004424                                     ENDIF
 (4)  004424                              50020$:
1096                                               ; CLEAN UP THE STACK AND EXIT
1097  004424                                       LET 8.(SP) := 4(SP)
 (4)  004424  016666  000004  000010            MOV     4(SP),8.(SP)
1098  004432                                     POP     R5,R4
 (2)  004432  012605                            MOV     (SP)+,R5
 (3)  004434  012604                            MOV     (SP)+,R4
1099  004436                                       LET SP := SP + #4
 (7)  004436  062706  000004                    ADD     #4,SP
1100  004442  000207                            RTS     PC
1101
1102
1103  004444  023420  001750  000144  TABLDA: .WORD   10000.,1000.,100.,10.,1
      004452  000012  000001
1104  004456  000000                  FLAGDA: .WORD   0
1105  004460  000000                  COUNTD: .WORD   0
1106  004462  000000                  DIGITS: .WORD   0
1107
```

```
1109                                    .SBTTL  I/O DRIVER
1110
1111                                    ;
1112                                    ;
1113                                    ;++
1114                                    ;THE I/O DRIVER ROUTINE IS INVOKED BY MEANS OF THE INTERRUPT SYSTEM.
1115                                    ;CALL TO IT IS JMP  IODRV.
1116                                    ;RETURN   RTI.
1117                                    ;ENTER ROUTINE WITH R2 SET UP TO DESIRED UNIT *2. R2 IS USED
1118                                    ;TO CALCULATE OFFSET INTO PROPER TABLE.
1119                                    ;R1 EQUALS MAXIMUM NUMBER OF UNITS ON SYSTEM UNDER TEST.
1120                                    ;
1121                                    ;--
1122                                    ;
1123                                    ; CHECK FOR ERROR FLAG IN STATUS REG.
1124                                    ;
1125  004464                            IODRV:  IF #BIT15 NOTSETIN @LPCSR(R2) THEN
 (6)  004464  032772  100000  002322            BIT     #BIT15,@LPCSR(R2)
 (10) 004472  001061                            BNE     50021$
1126                                    ;
1127                                    ; IF COUNT NOT ZERO SEND NEXT BYTE
1128                                    ;
1129  004474                                        IF CURCNT(R2) GT #0 THEN
 (6)  004474  005762  002716                    TST     CURCNT(R2)
 (10) 004500  003416                            BLE     50022$
1130  004502                                    LET @LPBUF(R2) :B= @CURADD(R2)
 (4)  004502  117272  002516  002416            MOVB    @CURADD(R2),@LPBUF(R2)
1131  004510                                    LET CURADD(R2) := CURADD(R2) + #1
 (7)  004510  005262  002516                    INC     CURADD(R2)
1132                                    ;
1133                                    ; ENABLE INTERRUPT FOR NEXT BYTE
1134                                    ;
1135  004514                                    LET STATUS(R2) := STATUS(R2) SET.BY #ACTIVE
 (7)  004514  052762  020000  002456            BIS     #ACTIVE,STATUS(R2)
1136  004522                                    LET CURCNT(R2) := CURCNT(R2) - #1
 (7)  004522  005362  002616                    DEC     CURCNT(R2)
1137  004526                                    LET @LPCSR(R2) := @LPCSR(R2) SET.BY #100
 (7)  004526  052772  000100  002322            BIS     #100,@LPCSR(R2)
1138  004534                                        ELSE
 (4)  004534  000437                            BR      50023$
 (3)  004536                            50022$:
1139                                    ; CURRENT MSG DONE, IF PRINT COUNT NOT ZERO SEND AGAIN
1140  004536                                        LET REPCNT(R2) := REPCNT(R2) - #1
 (7)  004536  005362  002616                    DEC     REPCNT(R2)
1141  004542                                        IF REPCNT(R2) GT #0 THEN
 (6)  004542  005762  002616                    TST     REPCNT(R2)
 (10) 004546  003424                            BLE     50024$
1142  004550                                            LET CURADD(R2) := MSGADR(R2)  ; RESTORE THE MSG ADDR
 (4)  004550  016262  002656  002516            MOV     MSGADR(R2),CURADD(R2)
1143  004556                                            LET CURCNT(R2) := MSGCNT(R2)  ; RESTORE THE BYTE COUNT
 (4)  004556  016262  002556  002716            MOV     MSGCNT(R2),CURCNT(R2)
1144  004564                                            LET @LPBUF(R2) :B= @CURADD(R2)  ; RESEND THE MESSAGE
 (4)  004564  117272  002516  002416            MOVB    @CURADD(R2),@LPBUF(R2)
1145  004572                                            LET CURADD(R2) := CURADD(R2) + #1 ; BUMP THE POINTER
 (7)  004572  005262  002516                    INC     CURADD(R2)
1146  004576                                            LET CURCNT(R2) := CURCNT(R2) - #1 ; DROP BYTE COUNT
```

```
   (7)  004576  005362  002716                    DEC     CURCNT(R2)
 1147   004602                                              LET STATUS(R2) := STATUS(R2) SET.BY #ACTIVE
   (7)  004602  052762  020000  002456            BIS     #ACTIVE,STATUS(R2)
 1148   004610                                              LET @LPCSR(R2) := #100  ; RE-ENABLE INTERRUPTS
   (4)  004610  012772  000100  002322            MOV     #100,@LPCSR(R2)
 1149   004616                                          ELSE
   (4)  004616  000406                            BR      50025$
   (3)  004620                                  50024$:
 1150                                           ; CURRENT MSG DONE, REPEAT COUNT =0
 1151                                           ; CLEAR ACTIVE AND DISABLE INTERRUPTS.
 1152   004620                                              LET STATUS(R2) := STATUS(R2) CLR.BY #ACTIVE
   (7)  004620  042762  020000  002456            BIC     #ACTIVE,STATUS(R2)
 1153   004626                                              LET @LPCSR(R2) := #00
   (4)  004626  012772  000000  002322            MOV     #00,@LPCSR(R2)
 1154   004634                                          ENDIF
   (4)  004634                                  50025$:
 1155   004634                                      ENDIF
   (4)  004634                                  50023$:
 1156   004634                                      ELSE
   (4)  004634  000410                            BR      50026$
   (3)  004636                                  50021$:
 1157                                            ; CLEAR ERROR CONDITION, ENABLE INTERRUPTS
 1158                                            ; SET ERROR FLAG
 1159   004636                                          LET STATUS(R2) := STATUS(R2) SET.BY #ERROR
   (7)  004636  052762  100000  002456            BIS     #ERROR,STATUS(R2)
 1160   004644                                          LET ERRCOD := #STATER       ; STATUS ERROR
   (4)  004644  012737  000001  002302            MOV     #STATER,ERRCOD
 1161   004652  004777  176200                        JSR PC,@ERRSVC
 1162                                                  ; ERROR SERVICE SHOULD CLEAR ERROR BIT AND ENABLE INTR
 1163   004656                                      ENDIF
   (4)  004656                                  50026$:
 1164   004656                                      POP R2
   (2)  004656  012602                            MOV     (SP)+,R2
 1165   004660  000002                            RTI
```

```
                                        .SBTTL I/O CONTROL
                                        ;++
                                        ;
                                        ; THE I/O CONTROL SUBROUTINE IS A SINGLE ENTRY QUEUE MANAGER.
                                        ; THIS ROUTINE IS INVOKED BY A JSR FROM AN I/O CALL.
                                        ; INPUTS:       PRINTR  -1 FOR ALL TERMINALS
                                        ;                       N FOR PRINTER NUMBER 'N'
                                        ;               BUFADD  ADDRESS OF MESSAGE TO PRINT
                                        ;               BUFCNT  BYTE COUNT TO TRANSMIT TO PRINTER
                                        ;
                                        ;               ERRSVC  ADDRESS OF ERROR SERVICE SUBROUTINE
                                        ;               BUFREP IS NO. OF TIMES TO PRINT THE MSG
                                        ;--
                                        ;
004662                                  IOCTRL: PUSH R2,R3
(2)  004662  010246                             MOV     R2,-(SP)
(3)  004664  010346                             MOV     R3,-(SP)
                                        ;
                                        ; IF PRINTR IS -1 QUE OUTPUT TO ALL PRINTERS SELECTED
                                        ; OTHERWISE TO UNIT NUMBER IN PRINTR.
                                        ;
004666                                          IF PRINTR EQ #-1 THEN
(6)  004666  023727  002264  177777             CMP     PRINTR,#-1
(10) 004674  001005                             BNE     50027$
004676                                              LET R3 := L$UNIT
(4)  004676  013703  002012                     MOV     L$UNIT,R3
004702                                              LET L$LUN := #0
(4)  004702  005037  002074                     CLR     L$LUN
004706                                          ELSE
(4)  004706  000405                             BR      50030$
(3)  004710                             50027$:
004710                                              LET R3 := #1
(4)  004710  012703  000001                     MOV     #1,R3
004714                                              LET L$LUN := PRINTR
(4)  004714  013737  002264  002074             MOV     PRINTR,L$LUN
004722                                          ENDIF
(4)  004722                             50030$:
                                        ;
                                        ; REPEAT TILL R3 = 0
                                        ;
004722                                  CTLLOP:
004722                                          IF R3 EQ #0 THEN
(6)  004722  005703                             TST     R3
(10) 004724  001002                             BNE     50031$
004726                                              INLINE <JMP CTLEND>
(2)  004726  000137  005240                     JMP     CTLEND
004732                                          ENDIF
(4)  004732                             50031$:
                                        ;
                                        ; USE R2 AS AN INDEX INTO THE UNIT TABLES
                                        ;
004732                                          LET R2 := L$LUN SHIFT 1
(5)  004732  013702  002074                     MOV     L$LUN,R2
(8)  004736  006302                             ASL     R2
004740                                          LET ERRCOD := #0
(4)  004740  005037  002302                     CLR     ERRCOD
```

```
1205
1206                                    ;
1207                                    ; IF THE UNIT HAS BEEN DROPPED SELECT THE NEXT UNIT
1208  004744                            ;
                                                IF #DROPED NOTSETIN STATUS(R2) THEN
 (6)  004744  032762  040000  002456           BIT     #DROPED,STATUS(R2)
 (10) 004752  001123                            BNE     50032$
1209                                    ;
1210                                    ;TEST FOR DVC ERROR BIT SET
1211                                    ;
1212  004754                                        IF #BIT15 SETIN @LPCSR(R2) THEN
 (6)  004754  032772  100000  002322           BIT     #BIT15,@LPCSR(R2)
 (10) 004762  001407                            BEQ     50033$
1213  004764                                            LET ERRCOD := #STATER       ; STATUS REG ERROR BIT 15 SET IN CSR
 (4)  004764  012737  000001  002302           MOV     #STATER,ERRCOD
1214  004772                                            LET STATUS(R2) := STATUS(R2) SFT.BY #ERROR
 (7)  004772  052762  100000  002456           BIS     #ERROR,STATUS(R2)
1215  005000                                            ELSE
 (4)  005000  000455                            BR      50034$
 (3)  005002                            50033$:
1216                                    ;
1217                                    ; MAKE SURE PREVIOUS MSG IS DONE
1218                                    ;
1219  005002                                        IF CURCNT(R2) GT #0 THEN
 (6)  005002  005762  002716                    TST     CURCNT(R2)
 (10) 005006  003452                            BLE     50035$
1220  005010                                            IF #ACTIVE NOTSETIN STATUS(R2) THEN
 (6)  005010  032762  020000  002456           BIT     #ACTIVE,STATUS(R2)
 (10) 005016  001004                            BNE     50036$
1221                                    ;
1222                                    ; OUTPUT WAS QUEUED BUT I/O DRIVER WAS NEVER INVOKED (VIA INTERRUPT)
1223                                    ;
1224  005020                                            LET ERRCOD := #NOINTR       ; NO INTERRUPT
 (4)  005020  012737  000003  002302           MOV     #NOINTR,ERRCOD
1225  005026                                            ELSE
 (4)  005026  000442                            BR      50037$
 (3)  005030                            50036$:
1226  005030                                        WHILE #ACTIVE SETIN STATUS(R2) DO
 (4)  005030                            50040$:
 (6)  005030  032762  020000  002456           BIT     #ACTIVE,STATUS(R2)
 (10) 005036  001436                            BEQ     50041$
1227  005040                                            LET DELCNT(R2) := #100.       ; 220 SEC
 (4)  005040  012762  000144  003016           MOV     #100.,DELCNT(R2)
1228  005046                                            DELAY 2.        ; 200MS LOOPS
 (2)  005046  012727  000002                    MOV     #2.,(PC)+
 (2)  005052  000000                            .WORD   0
 (2)  005054  013727  002116                    MOV     L$DLY,(PC)+
 (2)  005060  000000                            .WORD   0
 (2)  005062  005367  177772                    DEC     -6(PC)
 (2)  005066  001375                            BNE     .-4
 (2)  005070  005367  177756                    DEC     -22(PC)
 (2)  005074  001367                            BNE     .-20
1229  005076                                            LET DELCNT(R2) := DELCNT(R2) - #1
 (7)  005076  005362  003016                    DEC     DELCNT(R2)
1230  005102                                            IF DELCNT(R2) EQ #0 THEN
 (6)  005102  005762  003016                    TST     DELCNT(R2)
 (10) 005106  001011                            BNE     50042$
```

```
1231  005110                                                    LET ERRCOD := #TIMOUT
 (4)  005110  012737  000002  002302          MOV     #TIMOUT,ERRCOD
1232  005116                                                    LET STATUS(R2) := STATUS(R2) CLR.BY #ACTIVE
 (7)  005116  042762  020000  002456          BIC     #ACTIVE,STATUS(R2)
1233  005124                                                    LET STATUS(R2) := STATUS(R2) SET.BY #ERROR
 (7)  005124  052762  100000  002456          BIS     #ERROR,STATUS(R2)
1234  005132                                                ENDIF
 (4)  005132                               50042$:
1235  005132                                                ENDDO
 (4)  005132  000736                               BR      50040$
 (3)  005134                               50041$:
1236  005134                                            ENDIF
 (4)  005134                               50037$:
1237  005134                                        ENDIF
 (4)  005134                               50035$:
1238  005134                                        ENDIF
 (4)  005134                               50034$:
1239  005134                                        IF ERRCOD NE #0 THEN
 (6)  005134  005737  002302                       TST     ERRCOD
 (10) 005140  001403                               BEQ     50043$
1240                                       ;
1241                                       ; REPORT THE ERROR
1242                                       ;
1243  005142  004777  175710                                   JSR     PC,@ERRSVC
1244  005146                                                ELSE
 (4)  005146  000425                               BR      50044$
 (3)  005150                               50043$:
1245                                       ;
1246                                       ; Q UP THE MESSAGE AND ENABLE INTERRUPTS
1247                                       ; THE I/O DRIVER WILL PICK UP FROM HERE.
1248                                       ;
1249  005150                                                    LET CURADD(R2) := BUFADD        ; BYTE ADDRESS
 (4)  005150  013762  002314  002516          MOV     BUFADD,CURADD(R2)
1250  005156                                                    LET MSGADR(R2) := BUFADD        ; MESSAGE ADDRESS
 (4)  005156  013762  002314  002656          MOV     BUFADD,MSGADR(R2)
1251  005164                                                    LET CURCNT(R2) := BUFCNT        ; OUTPUT COUNT
 (4)  005164  013762  002316  002716          MOV     BUFCNT,CURCNT(R2)
1252  005172                                                    LET MSGCNT(R2) := BUFCNT        ; BYTE COUNT
 (4)  005172  013762  002316  002556          MOV     BUFCNT,MSGCNT(R2)
1253  005200                                                    LET REPCNT(R2) := BUFREP        ; PRINT COUNT
 (4)  005200  013762  002320  002616          MOV     BUFREP,REPCNT(R2)
1254  005206                                                    IF CURCNT(R2) GT #0 THEN
 (6)  005206  005762  002716                       TST     CURCNT(R2)
 (10) 005212  003403                               BLE     50045$
1255  005214                                                        LET @LPCSR(R2) := #100  ; ENABLE INTERRUPTS
 (4)  005214  012772  000100  002322          MOV     #100,@LPCSR(R2)
1256  005222                                                    ENDIF
 (4)  005222                               50045$:
1257  005222                                            ENDIF
 (4)  005222                               50044$:
1258  005222                                        ENDIF
 (4)  005222                               50032$:
1259                                       ;
1260                                       ; CLEAR OUT ANY TIMEOUT COUNT
1261                                       ;
1262  005222                                            LET DELCNT(R2) := #0
```

```
   (4)  005222  005062  003016                        CLR     DELCNT(R2)
  1263                                               ;
  1264                                               ; SELECT THE NEXT UNIT AND DECRIMENT THE LINECOUNT
  1265                                               ;
  1266  005226                                                 LET R3 := R3 - #1
   (7)  005226  005303                                       DEC     R3
  1267  005230                                                 LET L$LUN := L$LUN + #1
   (7)  005230  005237  002074                               INC     L$LUN
  1268  005234  000137  004722                               JMP     CTLLOP
  1269  005240                                     CTLEND:
  1270  005240                                               POP  R3,R2
   (2)  005240  012603                                       MOV     (SP)+,R3
   (3)  005242  012602                                       MOV     (SP)+,R2
  1271  005244  000207                                       RTS     PC
  1272
  1273                                               ;++++
  1274                                               ; SUBROUTINE    QUIET
  1275                                               ;
  1276                                               ; THIS SUBROUTINE WILL EFFECTIVLY DELAY UNTIL ALL QUEUED OUTPUT
  1277                                               ; IS FINISHED. THE DELAY IS ACCOMPLISHED BY QUEUEING A NULL
  1278                                               ; MESSAGE TO ALL LINES.
  1279                                               ;-----
  1280
  1281  005246                                     QUIET:  OUTPUT  #0,#0           ; NULL MESSAGE OUTPUT
  1282  005310  000240                                     NOP
  1283  005312  000207                                     RTS     PC
  1284
```

```
1286                                    ;=============================================================
1287                                    ; DROPIT        FUNCTIONAL DESCRIPTION :
1288                                    ;
1289                                    ; THIS SUBROUTINE IS USED TO DROP A BAD PRINTER FROM THE TEST
1290                                    ; DISABLE ANY INTERRUPTS FROM THE PRINTER, AND NOTIFY THE
1291                                    ; OPERATOR THAT THE PRINTER WAS DROPPED.
1292                                    ;=============================================================
1293
1294  005314                            DROPIT: LET STATUS(R2) := STATUS(R2) SET.BY #DROPED
 (7)  005314  052762  040000  002456            BIS     #DROPED,STATUS(R2)
1295  005322                                    LET CURCNT(R2) := #-1
 (4)  005322  012762  177777  002716            MOV     #-1,CURCNT(R2)
1296  005330                                    LET @LPCSR(R2) := #0
 (4)  005330  005072  002322                    CLR     @LPCSR(R2)
1297  005334                                    PRINTF #LPDROP, L$LUN
 (8)  005334  013746  002074                    MOV     L$LUN,-(SP)
 (7)  005340  012746  004002                    MOV     #LPDROP,-(SP)
 (6)  005344  012746  000002                    MOV     #2,-(SP)
 (3)  005350  010600                            MOV     SP,R0
 (4)  005352  104417                            TRAP    C$PNTF
 (4)  005354  062706  000006                    ADD     #6,SP
1298  005360                                    LET ERRTBL(R2) := #0
 (4)  005360  005062  003060                    CLR     ERRTBL(R2)
1299  005364                                    LET UUT := UUT - #1
 (7)  005364  005337  002306                    DEC     UUT
1300  005370                                    IF UUT EQ #0 THEN
 (6)  005370  005737  002306                    TST     UUT
 (10) 005374  001011                            BNE     50046$
1301  005376                                    FRINTF #UUTEQ0
 (7)  005376  012746  003732                    MOV     #UUTEQ0,-(SP)
 (6)  005402  012746  000001                    MOV     #1,-(SP)
 (3)  005406  010600                            MOV     SP,R0
 (4)  005410  104417                            TRAP    C$PNTF
 (4)  005412  062706  000004                    ADD     #4,SP
1302  005416                                        DOCLN       ; NOTHING TO TEST
 (3)  005416  104444                            TRAP    C$DCLN
1303  005420                                    ENDIF
 (4)  005420                            50046$:
1304  005420  000207                            RTS     PC
1305
1306
1307
1308                                    ;=============================================================
1309                                    ; FAKE          FUNCTIONAL DESCRIPTION:
1310                                    ;
1311                                    ; THIS SUBROUTINE IS REQUIRED TO INSURE PROPER PASS COUNT REPORTS
1312                                    ; IN A MULTI UNIT MODE OF OPERATION.
1313                                    ;=============================================================
1314
1315  005422                            FAKE:   LET L$LUN := #0
 (4)  005422  005037  002074                    CLR     L$LUN
1316  005426                                    WHILE L$LUN LT L$UNIT DO
 (4)  005426                            50047$:
 (6)  005426  023737  002074  002012            CMP     L$LUN,L$UNIT
 (10) 005434  002007                            BGE     50050$
1317  005436                                        GPHARD      L$LUN, R3
```

```
   (3)  005436  013700  002074          MOV     L$LUN,R0
   (3)  005442  104442                  TRAP    C$GPHRD
   (3)  005444  010003                  MOV     R0,R3
  1318  005446                              LET L$LUN := L$LUN + #1
   (7)  005446  005237  002074          INC     L$LUN
  1319  005452                          ENDDO
   (4)  005452  000765                  BR      50047$
   (3)  005454                  50050$:
  1320  005454  000207                  RTS     PC
  1321
  1322
  1323  005456                          ENDMOD
```

```
                                        .SBTTL  INITIALIZATION SECTION
                                        ;++
                                        ;THE INITIALIZE ROUTINE IS EXECUTED AT THE BEGINNING OF EACH SUB-PASS AND IS
                                        ;PRIMARILY USED FOR REQUESTING P-TABLE PARAMETERS. INFORMATION REQUESTED FROM
                                        ;THE OPERATOR INCLUDE THE NUMBER OF UNITS UNDER TEST, DEVICE ADDRESSES, VECTORS,
                                        ;AND CLOCK TYPE.
                                        ;--
005456                                  BGNMOD
005456                                  BGNINIT
005456                                  L$INIT::
                                        ;RESET EXTERNAL BUS IF START EVENT FLAG IS SET
                                        ;OR POWER FAIL RESTART
005456                                          READEF  #EF.START               ;TEST START EF INDICATOR
005456  012700  000040                          MOV     #EF.START,R0
005462  104447                                  TRAP    C$REFG
005464                                          BCOMPLETE 1$                    ;BRANCH IF FROM START UP
005464  103466                                  BCS     1$
005466                                          READEF  #EF.RESTART             ;NOW THE RESTARTFLAG
005466  012700  000037                          MOV     #EF.RESTART,R0
005472  104447                                  TRAP    C$REFG
005474                                          BCOMPLETE 1$                    ;IF EITHER START OR POWER FAIL RESTART
005474  103462                                  BCS     1$
                                                                                ;DO A BUS RESET
005476  004737  005422                          JSR     PC,FAKE                 ; UPDATE PASS COUNT
005502                                          SETPRI  #PR100                  ; PRIORITY ZERO
005502  012700  000000                          MOV     #PR100,R0
005506  104441                                  TRAP    C$SPRI
005510                                          LET OUTBUF :B= #14
005510  112737  000014  003124                  MOVB    #14,OUTBUF
005516                                          OUTPUT #OUTBUF,#1
005560                                          DECR WORK1 FROM #6 TO #1 BY #1
005560  012737  000006  003122                  MOV     #6,WORK1
005566  000402                                  BR      50051$
005570                                  50052$:
005570  005337  003122                          DEC     WORK1
005574                                  50051$:
005574  023727  003122  000001                  CMP     WORK1,#1
005602  002415                                  BLT     50053$
005604                                          DELAY 250
005604  012727  000250                          MOV     #250,(PC)+
005610  000000                                  .WORD   0
005612  013727  002116                          MOV     L$DLY,(PC)+
005616  000000                                  .WORD   0
005620  005367  177772                          DEC     -6(PC)
005624  001375                                  BNE     .-4
005626  005367  177756                          DEC     -22(PC)
005632  001367                                  BNE     .-20
005634                                          ENDDEC
005634  000755                                  BR      50052$
005636                                  50053$:
005636                                          EXIT INIT                       ; ELSE EXIT INIT CODE
005636  104432                                  TRAP    C$EXIT
005640  001304                                  .WORD   L10004-.
                                        ;
                                        ;POWER UP RESTART OR START COMMAND ISSUED
                                        ;
```

```
1352  005642                          1$:     BRESET                          ;RESET THE BUS
 (3)  005642  104433                         TRAP    C$RESET
1353  005644                                IF L$UNIT GT #16. THEN
 (6)  005644  023727  002012  000020        CMP     L$UNIT,#16.
 (10) 005652  003420                        BLE     50054$
1354  005654                                  PRINTF #NRGT16
 (7)  005654  012746  006506                MOV     #NRGT16,-(SP)
 (6)  005660  012746  000001                MOV     #1,-(SP)
 (3)  005664  010600                        MOV     SP,R0
 (4)  005666  104417                        TRAP    C$PNTF
 (4)  005670  062706  000004                ADD     #4,SP
1355  005674                                  PRINTF #NRGT17
 (7)  005674  012746  006571                MOV     #NRGT17,-(SP)
 (6)  005700  012746  000001                MOV     #1,-(SP)
 (3)  005704  010600                        MOV     SP,R0
 (4)  005706  104417                        TRAP    C$PNTF
 (4)  005710  062706  000004                ADD     #4,SP
1356  005714                                ENDIF
 (4)  005714                          50054$:
1357  005714                                MANUAL                          ; CHECK FOR UNATTENDED MODE
 (3)  005714  104450                        TRAP    C$MANI
1358  005716                                BNCOMPLETE      2$              ; IF UNATTENDED BYPASS MANUAL INSTRUCTIONS
 (2)  005716  103024                        BCC     2$
1359
1360  005720                                PRINTF #RESET1
 (7)  005720  012746  006701                MOV     #RESET1,-(SP)
 (6)  005724  012746  000001                MOV     #1,-(SP)
 (3)  005730  010600                        MOV     SP,R0
 (4)  005732  104417                        TRAP    C$PNTF
 (4)  005734  062706  000004                ADD     #4,SP
1361                                    ;
1362                                    ;WAIT FOR A "CR" BEFORE GOING ON
1363                                    ;
1364  005740                                LET FLAG := #0
 (4)  005740  005037  002240                CLR     FLAG
1365  005744                                LET ERRCOD := #0
 (4)  005744  005037  002302                CLR     ERRCOD
1366  005750                                LET UUT := #0
 (4)  005750  005037  002306                CLR     UUT
1367  005754                          100$:
1368  005754                                GMANIL  READY,FLAG,100000,YES
 (3)  005754  104443                        TRAP    C$GMAN
 (3)  005756  000404                        BR      10000$
 (4)  005760  002240                        .WORD   FLAG
 (5)  005762  000130                        .WORD   T$CODE
 (5)  005764  006752                        .WORD   READY
 (5)  005766  100000                        .WORD   100000
 (3)  005770                          10000$:
1369                                    ;
1370                                    ;REQUEST P-TABLE FOR PRINTERS UNDER TEST
1371                                    ;
1372  005770                          2$:     LET R1 := L$UNIT - #1           ;MAXIMUM NUMBER OF UNITS
 (5)  005770  013701  002012                MOV     L$UNIT,R1
 (7)  005774  005301                        DEC     R1
1373  005776                                INCR L$LUN FROM #0 TO R1 BY #1
 (5)  005776  005037  002074                CLR     L$LUN
```

```
   (7)  006002  000402                          BR      50055$
   (6)  006004                          50056$:
   (10) 006004  005237  002074                  INC     L$LUN
   (7)  006010                          50055$:
   (7)  006010  023701  002074                  CMP     L$LUN,R1
   (9)  006014  003073                          BGT     50057$
  1374  006016                                  GPHARD L$LUN,R3             ;REQUEST P-TABLE ADDRESS
   (3)  006016  013700  002074                  MOV     L$LUN,R0
   (3)  006022  104442                          TRAP    C$GPHRD
   (3)  006024  010003                          MOV     R0,R3
  1375  006026                                  BNCOMPLETE 3$               ;BRANCH IF DEVICE NOT PRESENT
   (2)  006026  103062                          BCC     3$
  1376  006030                                  LET R2 := L$LUN SHIFT 1
   (5)  006030  013702  002074                  MOV     L$LUN,R2
   (8)  006034  006302                          ASL     R2
  1377  006036                                  LET ERRTBL(R2) := #0
   (4)  006036  005062  003060                  CLR     ERRTBL(R2)
  1378  006042                                  LET CURCNT(R2) := #-1
   (4)  006042  012762  177777  002716          MOV     #-1,CURCNT(R2)
  1379  006050                                  LET DELCNT(R2) := #0
   (4)  006050  005062  003016                  CLR     DELCNT(R2)
  1380  006054                                  LET REPCNT(R2) := #0
   (4)  006054  005062  002616                  CLR     REPCNT(R2)
  1381                                  ;
  1382                                  ;LOAD CSR ADDRESS INTO TABLE
  1383                                  ;
  1384  006060                                  LET LPCSR(R2) := (R3)+      ;SET UP CSR ADDRESS FOR DEVICE
   (4)  006060  012362  002322                  MOV     (R3)+,LPCSR(R2)
  1385  006064                                  LET LPBUF(R2) := LPCSR(R2) + #2
   (5)  006064  016262  002322  002416          MOV     LPCSR(R2),LPBUF(R2)
   (7)  006072  062762  000002  002416          ADD     #2,LPBUF(R2)
  1386                                  ;
  1387                                  ;SET UP VECTOR ADDRESS INTO GIVEN TABLE
  1388                                  ;
  1389  006100                                      LET LPVEC(R2) := (R3)+
   (4)  006100  012362  002362                  MOV     (R3)+,LPVEC(R2)
  1390                                  ;
  1391                                  ;SET UP DEVICE INTERRUPT VECTOR INFORMATION
  1392                                  ;
  1393  006104                                  LET WORK := R2 SHIFT 3
   (5)  006104  010237  003120                  MOV     R2,WORK
   (8)  006110  006337  003120                  ASL     WORK
   (8)  006114  006337  003120                  ASL     WORK
   (8)  006120  006337  003120                  ASL     WORK
  1394  006124                                  LET WORK := WORK + #INT00
   (7)  006124  062737  021460  003120          ADD     #INT00,WORK
  1395  006132                                  LET LPINTR(R2) := WORK
   (4)  006132  013762  003120  002756          MOV     WORK,LPINTR(R2)
  1396  006140                                  SETVEC LPVEC(R2), LPINTR(R2), #PRI04
   (7)  006140  012746  000200                  MOV     #PRI04,-(SP)
   (6)  006144  016246  002756                  MOV     LPINTR(R2),-(SP)
   (5)  006150  016246  002362                  MOV     LPVEC(R2),-(SP)
   (4)  006154  012746  000003                  MOV     #3,-(SP)
   (3)  006160  104437                          TRAP    C$SVEC
   (2)  006162  062706  000010                  ADD     #10,SP
  1397                                  ;
```

```
 1398                                            ; ADD ONE TO UNIT UNDER TEST COUNT
 1399                                            ;
 1400  006166                                            LET UUT := UUT + #1
  (7)  006166  005237  002306                            INC     UUT
 1401  006172  000403                                        BR  4$
 1402                                            ;
 1403                                            ;INDICATE L$LUN NOT AVAILABLE FOR TESTING
 1404                                            ;
 1405  006174                                    3$:         LET STATUS(R2) := STATJS(R2) SET.BY #DROPED
  (7)  006174  052762  040000  002456                    BIS     #DROPED,STATUS(R2)
 1406  006202                                    4$:     ENDINC                          ;GO BACK AND DO IT AGAIN
  (5)  006202  000700                                    BR      50056$
  (4)  006204                                    50057$:
 1407                                            ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 1408                                            ; SETUP TO HANDLE CLOCK INTERRUPTS
 1409                                            ; IF AN L-CLOCK IS ON THE SYSTEM THEN SETUP A NOOP INTERRUPT
 1410                                            ; HANDLER BECAUSE LSI SYSTEMS MAY HAVE THE CLOCK ENABLED AT ALL TIMES.
 1411  006204                                            LET CLKTYP := #1        ; DEFAULT FOR NO CLOCK ON SYSTEM
  (4)  006204  012737  000001  002266                    MOV     #1,CLKTYP
 1412  006212                                            CLOCK   L,R4            ; TEST FOR L-CLOCK
  (3)  006212  012700  000114                            MOV     #'L,R0
  (3)  006216  104462                                    TRAP    C$CLCK
  (3)  006220  010004                                    MOV     R0,R4
 1413  006222                                            IFCOND CS THEN          ; WE HAVE AN L-CLOCK
  (7)  006222  103031                                    BCC     50060$
 1414  006224                                                LET CLKTYP := #2
  (4)  006224  012737  000002  002266                    MOV     #2,CLKTYP
 1415  006232                                                LET CLOCKP := R4
  (4)  006232  010437  002270                            MOV     R4,CLOCKP
 1416  006236                                                LET CLKCSR := @CLOCKP
  (4)  006236  017737  174026  002272                    MOV     @CLOCKP,CLKCSR
 1417  006244                                                LET @CLKCSR := #00  ; TRY TO DISABLE INTERRUPTS
  (4)  006244  012777  000000  174020                    MOV     #00,@CLKCSR
 1418                                                        ; SETUP THE NOOP HANDLER
 1419  006252                                                LET CLKVEC := 4(R4)
  (4)  006252  016437  000004  002276                    MOV     4(R4),CLKVEC
 1420  006260                                                SETVEC  CLKVEC,#IGNORE,#PRI06
  (7)  006260  012746  000300                            MOV     #PRI06,-(SP)
  (6)  006264  012746  007146                            MOV     #IGNORE,-(SP)
  (5)  006270  013746  002276                            MOV     CLKVEC,-(SP)
  (4)  006274  012746  000003                            MOV     #3,-(SP)
  (3)  006300  104437                                    TRAP    C$SVEC
  (2)  006302  062706  000010                            ADD     #10,SP
 1421  006306                                            ENDIF
  (4)  006306                                    50060$:
 1422                                            ; IF A P-CLOCK IS ON THE SYSTEM UPGRADE CLOCK TYPE TO 3
 1423  006306                                    CK1:    CLOCK   P,R4
  (3)  006306  012700  000120                            MOV     #'P,R0
  (3)  006312  104462                                    TRAP    C$CLCK
  (3)  006314  010004                                    MOV     R0,R4
 1424  006316                                            IFCOND CS THEN          ; WE HAVE A P-CLOCK
  (7)  006316  103016                                    BCC     50061$
 1425  006320                                                LET CLKTYP := #3
  (4)  006320  012737  000003  002266                    MOV     #3,CLKTYP
 1426  006326                                                LET CLOCKP := R4
  (4)  006326  010437  002270                            MOV     R4,CLOCKP
```

```
 1427  006332                                                    LET CLKCSR := @CLOCKP
  (4)  006332  017737  173732  002272            MOV        @CLOCKP,CLKCSR
 1428  006340                                                    LET CLKVEC := 4(R4)
  (4)  006340  016437  000004  002276            MOV        4(R4),CLKVEC
 1429                                                            ; TRY TO DISABLE THE P-CLOCK
 1430  006346                                                    LET @CLKCSR := #00
  (4)  006346  012777  000000  173716            MOV        #00,@CLKCSR
 1431  006354                                            ENDIF
  (4)  006354                                    50061$:
 1432                                            ; IF NO CLOCKS ON THE SYSTEM NOTIFY THE OPERATOR
 1433  006354                                            IF CLKTYP EQ #1 THEN
  (6)  006354  023727  002266  000001            CMP        CLKTYP,#1
 (10)  006362  001020                            BNE        50062$
 1434  006364                                                    PRINTF #NOCLCK
  (7)  006364  012746  007007                    MOV        #NOCLCK,-(SP)
  (6)  006370  012746  000001                    MOV        #1,-(SP)
  (3)  006374  010600                            MOV        SP,R0
  (4)  006376  104417                            TRAP       C$PNTF
  (4)  006400  062706  000004                    ADD        #4,SP
 1435  006404                                                    PRINTF #NOTIM
  (7)  006404  012746  007051                    MOV        #NOTIM,-(SP)
  (6)  006410  012746  000001                    MOV        #1,-(SP)
  (3)  006414  010600                            MOV        SP,R0
  (4)  006416  104417                            TRAP       C$PNTF
  (4)  006420  062706  000004                    ADD        #4,SP
 1436  006424                                            ENDIF
  (4)  006424                                    50062$:
 1437  006424                                            SETPRI #PR100
  (3)  006424  012700  000000                    MOV        #PR100,R0
  (3)  006430  104441                            TRAP       C$SPRI
 1438  006432                                    LET OUTBUF :B= #14
  (4)  006432  112737  000014  003124            MOVB       #14,OUTBUF
 1439  006440                                    OUTPUT #OUTBUF,#1
 1440  006502                                    EXIT INIT
  (3)  006502  104432                            TRAP       C$EXIT
  (3)  006504  000440                            .WORD      L10004-.
 1441                                            .NLIST BEX
 1442
 1443  006506  047045  040445  052516    NRGT16: .ASCIZ  /%N%ANUMBER OF LINE PRINTERS UNDER TEST EXCEEDS 16./
 1444  006571     045  022516  047501    NRGT17: .ASCIZ  /%N%AONLY 16 WILL BE TESTED./
 1445  006625     045  022516  051101    MRESET: .ASCIZ  /%N%ARESET PRINTER(S),  AND PLACE ON LINE.%N/
 1446  006701     045  022516  051101    RESET1: .ASCIZ /%N%ARESET PRINTER(S) AND PLACE ON LINE%N/
 1447
 1448  006752  042504  051120  051505    READY:  .ASCIZ  /DEPRESS "RETURN" WHEN READY./
 1449  007007     045  022516  044101    NOCLCK: .ASCIZ  /%N%AHARDWARE CLOCK NOT AVAILABLE./
 1450  007051     045  022516  040501    NOTIM:  .ASCIZ  /%N%AAUTO PRINTING SPEED MEASUREMENT CANNOT BE PERFORMED./
 1451                                    .EVEN
 1452  007142  000000                    PLOC:   .WORD   0
 1453
 1454                                    .LIST BEX
 1455  007144                            ENDINIT
  (3)  007144                            L10004:
  (3)  007144  104411                            TRAP       C$INIT
 1456
 1457                                    ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
 1458                                    ; IGNORE          AN INTERRUPT CATCHER FOR THE L-CLOCK
```

```
                                        ;               THAT IGNORES THE INTERRUPT.
                                        ;               USED FOR SYSTEMS WHERE CLOCK CANNOT BE TURNED OFF.
                                        ;::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

       007146                           IGNORE:                 ; NOOP
       007146  000002                           RTI



                                        ;------------------------------------------------------------------
                                        ;
                                        ; RESVEC                FUNCTIONAL DESCRIPTION
                                        ;
                                        ;       THIS SUBROUTINE WILL SETUP ALL UNITS VECTOR AREAS
                                        ;       TO THE 'NORMAL' INTERRUPT ROUTINES STARTING AT INT00.
                                        ;------------------------------------------------------------------

       007150                           RESVEC::                PUSH    R3,R4
  (2)  007150  010346                           MOV     R3,-(SP)
  (3)  007152  010446                           MOV     R4,-(SP)
       007154                                           LET  R4 := #0
  (4)  007154  005004                           CLR     R4
       007156                                           LET R3 := L$UNIT
  (4)  007156  013703  002012                   MOV     L$UNIT,R3
       007162                                           WHILE R3 GT #0 DO
  (4)  007162                           50063$:
  (6)  007162  005703                           TST     R3
 (10)  007164  003417                           BLE     50064$
       007166                                                   SETVEC LPVEC(R4), LPINTR(R4), #PRI04
  (7)  007166  012746  000200                   MOV     #PRI04,-(SP)
  (6)  007172  016446  002756                   MOV     LPINTR(R4),-(SP)
  (5)  007176  016446  002362                   MOV     LPVEC(R4),-(SP)
  (4)  007202  012746  000003                   MOV     #3,-(SP)
  (3)  007206  104437                           TRAP    C$SVEC
  (2)  007210  062706  000010                   ADD     #10,SP
       007214                                                   LET R4 := R4 + #2
  (7)  007214  062704  000002                   ADD     #2,R4
       007220                                                   LET R3 := R3 - #1
  (7)  007220  005303                           DEC     R3
       007222                                           ENDDO
  (4)  007222  000757                           BR      50063$
  (3)  007224                           50064$:
       007224                                           POP R4,R3
  (2)  007224  012604                           MOV     (SP)+,R4
  (3)  007226  012603                           MOV     (SP)+,R3
       007230  000207                                   RTS     PC
```

```
1488                                    .SBTTL  CLEANUP CODING SECTION
1489  007232                            STARS
 (2)                                    ;;**********.**********.**********.**********.**********.**********.**********.**********
1490                                    ;++
1491                                    ;THE PURPOSE OF THE CLEANUP SECTION IS TO CLEANUP ALL PRINTERS UNDER TEST
1492                                    ;AND RETEST ANY UNITS WHICH HAVE BEEN DROPPED FROM  TESTING TO INSURE THAT
1493                                    ;THEY HAVE NOT COME BACK ON LINE. IF THE DEVICE HAS COME BACK ON LINE
1494                                    ;TESTING WILL BE RESTARTED ON THE DEVICE. THIS INSURES THAT
1495                                    ;IN THE EVENT A PAPER OUT OCCURRED AND THE OPERATOR HAS PUT ADDITIONAL PAPER
1496                                    ;INTO THE UNIT UNDER TEST, THE INITIALIZATION SEQUENCE DOES NOT
1497                                    ;HAVE TO BE DONE AGAIN IN ORDER TO GET THE DEVICE ACTIVE.
1498                                    ;--
1499  007232                            STARS
 (2)                                    ;;**********.**********.**********.**********.**********.**********.**********.**********
1500  007232                            BGNCLN
 (3)  007232                            L$CLEAN::
1501  007232                                    SETPRI #PRI07
 (3)  007232  012700  000340                      MOV     #PRI07,R0
 (3)  007236  104441                              TRAP    C$SPRI
1502  007240                                        BRESET
 (3)  007240  104433                              TRAP    C$RESET
1503
1504  007242                            CLEAN:  LET R1 .= L$UNIT - #1   ;NUMBER OF UNITS-1
 (5)  007242  013701  002012                      MOV     L$UNIT,R1
 (7)  007246  005301                              DEC     R1
1505  007250                                    INCR L$LUN FROM #0 TO R1 BY #1
 (5)  007250  005037  002074                      CLR     L$LUN
 (7)  007254  000402                              BR      50065$
 (6)  007256                            50066$:
 (10) 007256  005237  002074                      INC     L$LUN
 (7)  007262                            50065$:
 (7)  007262  023701  002074                      CMP     L$LUN,R1
 (9)  007266  003020                              BGT     50067$
1506                                    ; DISABLE ALL INTERRUPTS, SELECT ALL LINES
1507                                    ; ZERO ALL ERROR COUNTS
1508  007270                                        LET R2 := L$LUN SHIFT 1
 (5)  007270  013702  002074                      MOV     L$LUN,R2
 (8)  007274  006302                              ASL     R2
1509                                            ; CLEAR ALL BITS IN STATUS EXCEPT DEVICE TYPE
1510  007276                                    LET STATUS(R2) := STATUS(R2) CLR.BY #ERROR!DROPED!ACTIVE!LOBYTE
 (7)  007276  042762  160377  002456          BIC     #ERROR!DROPED!ACTIVE!LOBYTE,STATUS(R2)
1511  007304                                    LET CURCNT(R2) := #-1
 (4)  007304  012762  177777  002716          MOV     #-1,CURCNT(R2)
1512  007312                                    LET ERRTBL(R2) := #0
 (4)  007312  005062  003060                      CLR     ERRTBL(R2)
1513  007316                                    LET DELCNT(R2) := #0
 (4)  007316  005062  003016                      CLR     DELCNT(R2)
1514  007322                                    LET REPCNT(R2) := #0
 (4)  007322  005062  002616                      CLR     REPCNT(R2)
1515  007326                                    ENDINC
 (5)  007326  000753                              BR      50066$
 (4)  007330                            50067$:
1516  007330  004737  007150                      JSR     PC,RESVEC               ; RESET THE VECTORS
1517  007334                                    IF CLKTYP EQ #3 THEN
 (6)  007334  023727  002266  000003              CMP     CLKTYP,#3
 (10) 007342  001006                              BNE     50070$
```

```
1518  007344                                    CLRVEC @CLKVEC
 (3)  007344  017700  172726                    MOV     @CLKVEC,R0
 (3)  007350  104436                            TRAP    C$CVEC
1519  007352                                    LET @CLKCSR := #00
 (4)  007352  012777  000000  172712            MOV     #00,@CLKCSR
1520  007360                                    ENDIF
 (4)  007360                              50070$:
1521  007360                                    IF CLKTYP EQ #2 THEN
 (6)  007360  023727  002266  000002            CMP     CLKTYP,#2
 (10) 007366  001013                            BNE     50071$
1522  007370                                    SETVEC CLKVEC,#IGNORE,#PRI06
 (7)  007370  012746  000300                    MOV     #PRI06,-(SP)
 (6)  007374  012746  007146                    MOV     #IGNORE,-(SP)
 (5)  007400  013746  002276                    MOV     CLKVEC,-(SP)
 (4)  007404  012746  000003                    MOV     #3,-(SP)
 (3)  007410  104437                            TRAP    C$SVEC
 (2)  007412  062706  000010                    ADD     #10,SP
1523  007416                                    ENDIF
 (4)  007416                              50071$:
1524  007416                                    SETPRI #PRI00
 (3)  007416  012700  000000                    MOV     #PRI00,R0
 (3)  007422  104441                            TRAP    C$SPRI
1525  007424                              ENDCLN
 (3)  007424                              L10005:
 (3)  007424  104412                            TRAP    C$CLEAN
1526
1527  007426                              ENDMOD
1528                                      .SBTTL  INTERFACE LOGIC
1529
1530  007426                              BGNMOD
1531                                      ;++
1532                                      ;THIS TEST VERIFIES THE OPERATION OF THE INTERFACE LOGIC. TESTS ARE
1533                                      ;PERFORMED FOR PRINTER ERROR, PRINTER READY, AND CLEARING PRINTER READY
1534                                      ;BY LOADING A CHARACTER INTO THE OUTPUT BUFFER. ALSO IT IS VERIFIED
1535                                      ;THAT THE PRINTER WILL NOT INTERRUPT IF IT IS AT THE SAME PRIORITY LEVEL
1536                                      ;AS THE PROCESSOR, BUT WILL INTERRUPT IF THE PROCESSOR IS AT A LOWER
1537                                      ;PRIORITY LEVEL. THE PRINTER IS AT PRIORITY LEVEL 4.
1538                                      ;
1539                                      ;
1540                                      ;--
1541  007426                              BGNTST  1
 (3)  007426                              T1::
1542  007426                              LET R1 := L$UNIT - #1                   ;MAX NUMBER OF UNITS ON SYSTEM
 (5)  007426  013701  002012                    MOV     L$UNIT,R1
 (7)  007432  005301                            DEC     R1
1543                                      ;
1544                                      ;HARD CODED INCREMEMNT LOOP
1545                                      ;INCR LUNIT FROM #0 TO R1 BY #1         ;START LOOP
1546                                      ;
1547  007434  005037  002260                    CLR     LUNIT                   ;UNIT TO 0
1548  007440  000402                            BR      T1C                     ;DO COMPARE
1549  007442                              T1A:
1550  007442  005237  002260                    INC     LUNIT                   ;UPDATE UNIT NUMBER
1551  007446                              T1C:
1552  007446  023701  002260                    CMP     LUNIT,R1                ;DO COMPARISON OF UNIT NUMBER
1553  007452  003402                            BLE     1$                      ;ONTO NEXT UNIT
```

```
1554  007454  000137  010220                        JMP     T1B                     ;EXIT LOOP
1555  007460                                1$:
1556  007460                                        LET R2 := LUNIT SHIFT 1
 (5)  007460  013702  002260                        MOV     LUNIT,R2
 (8)  007464  006302                                ASL     R2
1557  007466                                    IF #BIT15 SETIN @LPCSR(R2) THEN
 (6)  007466  032772  100000  002322                BIT     #BIT15,@LPCSR(R2)
(10)  007474  001416                                BEQ     50072$
1558  007476                                        LET STATUS(R2) := STATUS(R2) SET.BY #ERROR
 (7)  007476  052762  100000  002456                BIS     #ERROR,STATUS(R2)
1559  007504                                        LET ERRTBL(R2) := ERRTBL(R2) + #1
 (7)  007504  005262  003060                        INC     ERRTBL(R2)
1560  007510                                        LET L$LUN := LUNIT
 (4)  007510  013737  002260  002074                MOV     LUNIT,L$LUN
1561  007516                                        ERRHRD 1,CSRERR         ;ERROR BIT WAS SET. SAY SO
 (4)  007516  104456                                TRAP    C$ERHRD
 (5)  007520  000001                                .WORD   1
 (5)  007522  003352                                .WORD   CSRERR
 (5)  007524  000000                                .WORD   0
1562  007526                                        LET @LPCSR(R2) := #0
 (4)  007526  005072  002322                        CLR     @LPCSR(R2)
1563  007532                                    ENDIF
 (4)  007532                            50072$:
1564                                    ;TIME DELAY
1565                                            ; IF NOT READY ALLOW 3 SECONDS TO COME UP
1566  007532                                    IF #BIT7 NOTSETIN @LPCSR(R2) THEN
 (6)  007532  032772  000200  002322                BIT     #BIT7,@LPCSR(R2)
(10)  007540  001027                                BNE     50073$
1567  007542                                        DECR WORK1 FROM #12. TO #1 BY #1
 (5)  007542  012737  000014  003122                MOV     #12.,WORK1
 (7)  007550  000402                                BR      50074$
 (6)  007552                            50075$:
(10)  007552  005337  003122                        DEC     WORK1
 (7)  007556                            50074$:
 (7)  007556  023727  003122  000001                CMP     WORK1,#1
 (9)  007564  002415                                BLT     50076$
1568  007566                                        DELAY 250
 (2)  007566  012727  000250                        MOV     #250.,(PC)+
 (2)  007572  000000                                .WORD   0
 (2)  007574  013727  002116                        MOV     L$DLY,(PC)+
 (2)  007600  000000                                .WORD   0
 (2)  007602  005367  177772                        DEC     -6(PC)
 (2)  007606  001375                                BNE     .-4
 (2)  007610  005367  177756                        DEC     -22(PC)
 (2)  007614  001367                                BNE     .-20
1569  007616                                        ENDDEC
 (5)  007616  000755                                BR      50075$
 (4)  007620                            50076$:
1570  007620                                    ENDIF
 (4)  007620                            50073$:
1571                                    ;
1572                                    ;NOW TEST FOR PRINTER READY
1573                                    ;
1574  007620                                IF #BIT07 NOTSETIN @LPCSR(R2) THEN          ;TEST FOR THE READY BIT
 (6)  007620  032772  000200  002322                BIT     #BIT07,@LPCSR(R2)
(10)  007626  001014                                BNE     50077$
```

```
1575  007630                                    LET STATUS(R2) := STATUS(R2) SET.BY #ERROR
 (7)  007630  052762  100000  002456            BIS     #ERROR,STATUS(R2)
1576  007636                                    LET L$LUN := LUNIT
 (4)  007636  013737  002260  002074            MOV     LUNIT,L$LUN
1577  007644                                    LET ERRTBL(R2) := ERRTBL(R2) + #1
 (7)  007644  005262  003060                     INC     ERRTBL(R2)
1578  007650                                    ERRHRD 2,RDYERR          ;REPORT AN ERROR
 (4)  007650  104456                            TRAP    C$ERHRD
 (5)  007652  000002                            .WORD   2
 (5)  007654  003370                            .WORD   RDYERR
 (5)  007656  000000                            .WORD   0
1579  007660                                ENDIF
 (4)  007660                            50077$:
1580                                    ;
1581                                    ;INSURE LOADING CHARACTER CAUSES PRINTER READY TO GO AWAY
1582                                    ;
1583  007660                                LET @LPBUF(R2) := #12
 (4)  007660  012772  000012  002416            MOV     #12,@LPBUF(R2)
1584  007666                                IF #BIT07 SETIN @LPCSR(R2) THEN
 (6)  007666  032772  000200  002322            BIT     #BIT07,@LPCSR(R2)
 (10) 007674  001416                            BEQ     50100$
1585  007676                                    LET STATUS(R2) := STATUS(R2) SET.BY #ERROR
 (7)  007676  052762  100000  002456            BIS     #ERROR,STATUS(R2)
1586  007704                                    LET ERRTBL(R2) := ERRTBL(R2) + #1
 (7)  007704  005262  003060                     INC     ERRTBL(R2)
1587  007710                                    LET L$LUN := LUNIT
 (4)  007710  013737  002260  002074            MOV     LUNIT,L$LUN
1588  007716                                    ERRHRD 3,ERR11           ;REPORT AN ERROR
 (4)  007716  104456                            TRAP    C$ERHRD
 (5)  007720  000003                            .WORD   3
 (5)  007722  010504                            .WORD   ERR11
 (5)  007724  000000                            .WORD   0
1589  007726                                    LET @LPCSR(R2) := #0
 (4)  007726  005072  002322                     CLR     @LPCSR(R2)
1590  007732                                ENDIF
 (4)  007732                            50100$:
1591                                    ;
1592                                    ;VERIFY THAT THE PRINTER WILL NOT INTERRUPT IF IT IS AT A PRIORITY LEVEL
1593                                    ;THE SAME AS THE CPU
1594                                    ;
1595  007732                                SETPRI #PRI04                        ;CPU TO PRIORITY 4
 (3)  007732  012700  000200                    MOV     #PRI04,R0
 (3)  007736  104441                            TRAP    C$SPRI
1596  007740                                SETVEC LPVEC(R2),#INTERR,#PRI04      ;LP VECTOR SET UP
 (7)  007740  012746  000200                    MOV     #PRI04,-(SP)
 (6)  007744  012746  010366                    MOV     #INTERR,-(SP)
 (5)  007750  016246  002362                    MOV     LPVEC(R2),-(SP)
 (4)  007754  012746  000003                    MOV     #3,-(SP)
 (3)  007760  104437                            TRAP    C$SVEC
 (2)  007762  062706  000010                    ADD     #10,SP
1597  007766                                    LET @LPCSR(R2) := @LPCSR(R2) SET.BY #100     ;INTERRUPT ENABLE
 (7)  007766  052772  000100  002322            BIS     #100,@LPCSR(R2)
1598  007774                                    DECR WORK1 FROM #12 TO #1 BY #1
 (5)  007774  012737  000012  003122            MOV     #12,WORK1
 (7)  010002  000402                            BR      50101$
 (6)  010004                            50102$:
```

```
   (10) 010004  005337  003122                        DEC     WORK1
    (7) 010010                                >0101$:
    (7) 010010  023727  003122  000001                CMP     WORK1,#1
    (9) 010016  002415                                BLT     50103$
  1599 010020                                         DELAY   250.                ; ALLOW 3 SEC FOR DELAY
    (2) 010020  012727  000372                        MOV     #250.,(PC)+
    (2) 010024  000000                                .WORD   0
    (2) 010026  013727  002116                        MOV     L$DLY,(PC)+
    (2) 010032  000000                                .WORD   0
    (2) 010034  005367  177772                        DEC     -6(PC)
    (2) 010040  001375                                BNE     .-4
    (2) 010042  005367  177756                        DEC     -22(PC)
    (2) 010046  001367                                BNE     .-20
  1600 010050                                     ENDDEC
    (5) 010050  000755                                BR      50102$
    (4) 010052                                50103$:
  1601                                        ;
  1602                                        ;NOW TEST THAT THE PRINTER WILL INTERRUPT IF THE CPU PRIORITY IS LOWER THAN
  1603                                        ;THE PRINTER PRIORITY
  1604                                        ;
  1605 010052                                     LET @LPCSR(R2) := @LPCSR(R2) CLR.BY #100    ;CLEAR INTERRUPT ENABLE
    (7) 010052  042772  000100  002322                BIC     #100,@LPCSR(R2)
  1606 010060                                     SETPRI #PRI03                           ;CPU TO PRIORITY 3
    (3) 010060  012700  000140                        MOV     #PRI03,R0
    (3) 010064  104441                                TRAP    C$SPRI
  1607 010066                                     SETVEC LPVEC(R2),#INTHDL,#PRI04
    (7) 010066  012746  000200                        MOV     #PRI04,-(SP)
    (6) 010072  012746  010416                        MOV     #INTHDL,-(SP)
    (5) 010076  016246  002362                        MOV     LPVEC(R2),-(SP)
    (4) 010102  012746  000003                        MOV     #3,-(SP)
    (3) 010106  104437                                TRAP    C$SVEC
    (2) 010110  062706  000010                        ADD     #10,SP
  1608 010114                                     LET @LPCSR(R2) := @LPCSR(R2) SET.BY #100    ;INTERRUPT ENABLE
    (7) 010114  052772  000100  002322                BIS     #100,@LPCSR(R2)
  1609 010122                                         DELAY 30                        ; ALLOW 3 SEC DELAY
    (2) 010122  012727  000030                        MOV     #30,(PC)+
    (2) 010126  000000                                .WORD   0
    (2) 010130  013727  002116                        MOV     L$DLY,(PC)+
    (2) 010134  000000                                .WORD   0
    (2) 010136  005367  177772                        DEC     -6(PC)
    (2) 010142  001375                                BNE     .-4
    (2) 010144  005367  177756                        DEC     -22(PC)
    (2) 010150  001367                                BNE     .-20
  1610 010152                                         LET ERRTBL(R2) := ERRTBL(R2) + #1
    (7) 010152  005262  003060                        INC     ERRTBL(R2)
  1611 010156                                         LET L$LUN := LUNIT
    (4) 010156  013737  002260  002074                MOV     LUNIT,L$LUN
  161< 010164                                     ERRHRD 4,ERR13
    (4) 010164  104456                                TRAP    C$ERHRD
    (5) 010166  000004                                .WORD   4
    (5) 010170  010643                                .WORD   ERR13
    (5) 010172  000000                                .WORD   0
  1613 010174                                 END2:   LET @LPCSR(R2) := #00               ; CLEAR THE LPCSR
    (4) 010174  012772  000000  002322                MOV     #00,@LPCSR(R2)
  1614 010202                                         LET STATUS(R2) := STATUS(R2) CLR.BY #ERROR DROPED!ACTIVE
    (7) 010202  042762  160000  002456                BIC     #ERROR!DROPED!ACTIVE,STATUS(R2)
```

```
1615  010210                                  LET DELCNT(R2) := #0
 (4)  010210  005062  003016                  CLR     DELCNT(R2)
1616                                    ;
1617                                    ;END OF HARD CODED INCREMENT LOOP
1618                                    ;ENDINC
1619                                    ;
1620  010214  000137  007442                  JMP     T1A                     ;UPDATE UNIT #
1621  010220  004737  007150            T1B:  JSR     PC,RESVEC       ; RESET STANDARD VECTORS
1622  010224                                  SETPRI #PRI00
 (3)  010224  012700  000000                  MOV     #PRI00,R0
 (3)  010230  104441                          TRAP    C$SPRI
1623  010232                                  OUTPUT #INTFAC,#47.
1627  010274                                  LET OUTBUF :B= #14
 (4)  010274  112737  000014  003124          MOVB    #14,OUTBUF
1628  010302                                  OUTPUT #OUTBUF,#1
1629  010344  004737  005246                  JSR PC,QUIET
1630  010350                                  WHILE #BIT7 NOTSETIN @LPCSR(R2) DO      ;WAIT FOR READY
 (4)  010350                            50104$:
 (6)  010350  032772  000200  002322          BIT     #BIT7,@LPCSR(R2)
 (10) 010356  001001                          BNE     50105$
1631  010360                                  ENDDO
 (4)  010360  000773                          BR      50104$
 (3)  010362                            50105$:
1632  010362                            EXIT TST                                ;EXIT THE TEST
 (3)  010362  104432                          TRAP    C$EXIT
 (3)  010364  000334                          .WORD   L10006-.
1633                                    ;
1634                                    ;INTERRUPT HANDLER TO SERVICE FAULTY INTERRUPT FROM LP INTERFACE.
1635                                    ;THIS ROUTINE IS ENTERED ONLY WHEN THE LP INTERRUPTS AT THE SAME LEVEL AS
1636                                    ;THE CPU AND IS CONSIDERED AN ERROR.
1637                                    ;
1638  010366                            BGNSRV
1639  010366                            INTERR: LET ERRTBL(R2) := ERRTBL(R2) + #1
 (7)  010366  005262  003060                  INC     ERRTBL(R2)
1640  010372                                  LET L$LUN :- LUNIT
 (4)  010372  013737  002260  002074          MOV     LUNIT,L$LUN
1641  010400                                  ERRHRD 5,ERR12
 (4)  010400  104456                          TRAP    C$ERHRD
 (5)  010402  000005                          .WORD   5
 (5)  010404  010560                          .WORD   ERR12
 (5)  010406  000000                          .WORD   0
1642  010410                            LET (SP) := #END2
 (4)  010410  012716  010174                  MOV     #END2,(SP)
1643  010414                            ENDSRV
 (3)  010414                            L10007:
 (2)  010414  000002                          RTI
1644                                    ;
1645                                    ;INTERRUPT HANDLER FOR EXPECTED INTERRUPT
1646                                    ;
1647  010416                            BGNSRV
1648                                    ;
1649  010416                            INTHDL: LET (SP) := #END2
 (4)  010416  012716  010174                  MOV     #END2,(SP)
1650  010422                            ENDSRV
 (3)  010422                            L10010:
 (2)  010422  000002                          RTI
```

```
1651                                    ;
1652                                    .NLIST BEX
1653  010424  047111  042524  043122    INTFAC: .ASCIZ  /INTERFACE LOGIC TEST 1   ----   TEST COMPLETE/<12><12>
1654                                    ;
1655                                    ;ERROR MESSAGES ASSOCIATED WITH THIS TEST
1656                                    ;
1657  010504  047514  042101  047111    ERR11:  .ASCIZ  /LOADING PRINTER BUFFER DOES NOT CLEAR READY/
1658  010560  051120  047111  042524    ERR12:  .ASCIZ  /PRINTER INTERRUPTED AT SAME LEVEL AS THE PROCESSOR/
1659  010643     120  044522  052116    ERR13:  .ASCIZ  /PRINTER DID NOT INTERRUPT AT CPU PRIORITY 3/
1660          010720                    .EVEN
1661  010720                            ENDTST
 (3)  010720                            L10006:
 (3)  010720  104401                            TRAP    C$ETST
1662                                    .LIST BEX
1663  010722                            ENDMOD
1664
```

```
                                        .SBTTL DATA TRANSFER PATHS

                                        BGNMOD
                                        ;++
                                        ;THIS TEST CHECKS THE DATA TRANSFER
                                        ;PATHS FROM THE PROCESSOR INTERFACE
                                        ;TO THE PRINTER OUTPUT. AN ALTERNATING
                                        ;PATTERN OF ONES AND ZEROES CORRESPONDING
                                        ;TO AN ALTERNATING STRING OF "*" AND
                                        ;"U" CHARACTERS ARE TRANSMITTED ON THE
                                        ;FULL 132 COLUMNS.  AFTER 16 LINES OF
                                        ;THIS PATTERN, THE OUTPUT PATTERN IS
                                        ;SWITCHED TO AN ALTERNATING PATTERN
                                        ;OF "?" AND "@" CHARACTERS FOR ANOTHER
                                        ;16 LINES.
                                        ;--

                                        BGNTST  2
 (3)  010722                            T2::

                                        ;PRINT TEST IDENTIFICATION
                                        OUTPUT #DATPTH,#29.
                                        ;PRINT ALTERNATING STRINGS OF CHARACTERS
                                        INCR PATTERN FROM #1 TO #2 BY #1
 (5)  010764  012737  000001  011352            MOV     #1,PATTERN
 (7)  010772  000402                            BR      50106$
 (6)  010774                            50107$:
 (10) 010774  005237  011352                    INC     PATTERN
 (7)  011000                            50106$:
 (7)  011000  023727  011352  000002            CMP     PATTERN,#2
 (9)  011006  003111                            BGT     50110$
                                            IF PATTERN EQ #1 THEN
 (6)  011010  023727  011352  000001            CMP     PATTERN,#1
 (10) 011016  001004                            BNE     50111$
                                                    LET CHAR :B= #'U
 (4)  011020  112737  000125  011312            MOVB    #'U,CHAR
                                                ELSE
 (4)  011026  000403                            BR      50112$
 (3)  011030                            50111$:
                                                    LET CHAR :B= #'?
 (4)  011030  112737  000077  011312            MOVB    #'?,CHAR
                                            ENDIF
 (4)  011036                            50112$:
                                                LET R4 := #OUTBUF
 (4)  011036  012704  003124                    MOV     #OUTBUF,R4
                                                INCR CCNT FROM #1 TO #66. BY #1
 (5)  011042  012737  000001  002250            MOV     #1,CCNT
 (7)  011050  000402                            BR      50113$
 (6)  011052                            50114$:
 (10) 011052  005237  002250                    INC     CCNT
 (7)  011056                            50113$:
 (7)  011056  023727  002250  000102            CMP     CCNT,#66.
 (9)  011064  003017                            BGT     50115$
                                                    LET (R4)+ :B= CHAR
 (4)  011066  113724  011312                    MOVB    CHAR,(R4)+
      011072  105137  011312                        COMB CHAR
```

```
1701  011076                                        LET CHAR :B= CHAR CLR.BY #200
 (7)  011076  142737  000200  011312          BICB    #200,CHAR
1702  011104                                        LET (R4)+ :B= CHAR
 (4)  011104  113724  011312                MOVB    CHAR,(R4)+
1703  011110  105137  011312                    COMB CHAR
1704  011114                                        LET CHAR :B= CHAR CLR.BY #200
 (7)  011114  142737  000200  011312          BICB    #200,CHAR
1705  011122                                    ENDINC
 (5)  011122  000753                          BR      50114$
 (4)  011124                              50115$:
1706  011124                                    LET (R4)+ :B= #15
 (4)  011124  112724  000015                  MOVB    #15,(R4)+
1707  011130                                    LET (R4) :B= #12
 (4)  011130  112714  000012                  MOVB    #12,(R4)
1708  011134                                    INCR LINCNT FROM #1 TO #16. BY #1
 (5)  011134  012737  000001  002242          MOV     #1,LINCNT
 (7)  011142  000402                          BR      50116$
 (6)  011144                              50117$:
 (10) 011144  005237  002242                  INC     LINCNT
 (7)  011150                              50116$:
 (7)  011150  023727  002242  000020          CMP     LINCNT,#16.
 (9)  011156  003024                          BGT     50120$
1709  011160                                        OUTPUT #OUTBUF, #134.
1710  011222  004737  005246                       JSR PC,     QUIET
1711  011226                                    ENDINC
 (5)  011226  000746                          BR      50117$
 (4)  011230                              50120$:
1712  011230                              ENDINC
 (5)  011230  000661                          BR      50107$
 (4)  011232                              50110$:
1713  011232                              LET OUTBUF :B= #14
 (4)  011232  112737  000014  003124              MOVB    #14,OUTBUF
1714  011240                              OUTPUT  #OUTBUF, #1
1715  011302  004737  005246              JSR PC,QUIET
1716  011306                              EXIT TST
 (3)  011306  104432                              TRAP    C$EXIT
 (3)  011310  000044                              .WORD   L10011-.
1717                                      .NLIST BEX
1718  011312  000000                      CHAR:   .WORD 0
1719  011314  040504  040524  052040    DATPTH: .ASCIZ /DATA TRANSFER PATHS TEST 2/ <12><12><12>
1720
1721                                      .EVEN
1722  011352  000000                      PATTERN:        .WORD 0
1723                                                      .EVEN
1724                                      .EVEN
1725                                      .LIST BEX
1726
1727  011354                              ENDTST
 (3)  011354                              L10011:
 (3)  011354  104401                              TRAP    C$ETST
1728
1729  011356                              ENDMOD
```

```
 1731                                   .SBTTL PRINTABLE CHARACTERS
 1732  011356                           BGNMOD
 1733                                   ;++
 1734                                   ; THIS TEST WILL PRINT A FULL LINE OF EACH UPPER AND LOWER CASE PRINTABLE CHARACTER
 1735                                   ; IN THE 7 BIT RANGE
 1736                                   ;--
 1737
 1738  011356                           BGNTST  3
  (3)  011356                           T3::
 1742  011356                                   OUTPUT #PRTCHR, #30.            ; PRINT TEST ID
 1743                                   ;
 1744                                   ; PRINT ALL CHARACTERS ON ALL UNITS
 1745                                   ;
 1746          000001                   $BRJMP=1
 1747  011420                                   INCR WORK FROM #40 TO #177 BY #1
  (5)  011420  012737  000040  003120          MOV     #40,WORK
  (7)  011426  000402                          BR      50122$
  (6)  011430                           50121$:
  (8)  011430  005237  003120                  INC     WORK
  (6)  011434                           50122$:
  (7)  011434  023727  003120  000177          CMP     WORK,#177
  (9)  011442  003402                          BLE     50123$
  (7)  011444  000137  011566                  JMP     50124$
  (6)  011450                           50123$:
 1748  011450                                       LET R4 := #OUTBUF
  (4)  011450  012704  003124                  MOV     #OUTBUF,R4
 1749  011454                                       INCR COUNT FROM #1 TO #132. BY #1
  (5)  011454  012737  000001  002246          MOV     #1,COUNT
  (7)  011462  000402                          BR      50126$
  (6)  011464                           50125$:
  (8)  011464  005237  002246                  INC     COUNT
  (6)  011470                           50126$:
  (7)  011470  023727  002246  000204          CMP     COUNT,#132.
  (9)  011476  003402                          BLE     50127$
  (7)  011500  000137  011512                  JMP     50130$
  (6)  011504                           50127$:
 1750  011504                                       LET (R4)+ :B= WORK
  (4)  011504  113724  003120                  MOVB    WORK,(R4)+
 1751  011510                                   ENDINC
  (4)  011510  000765                          BR      50125$
  (4)  011512                           50130$:
 1752  011512                                       LET (R4)+ :B= #LF
  (4)  011512  112724  000012                  MOVB    #LF,(R4)+
 1753  011516                                       OUTPUT #OUTBUF,#133.
 1754  011560  004737  005246                      JSR PC,QUIET
 1755  011564                                   ENDINC
  (4)  011564  000721                          BR      50121$
  (4)  011566                           50124$:
 1756  011566                                   OUTPUT #DONE,#14.                       ; TEST DONE MESSAGE
 1757  011630                                   LET OUTBUF :B= #14
  (4)  011630  112737  000014  003124          MOVB    #14,OUTBUF
 1758  011636                                   OUTPUT #OUTBUF,#1                       ; EXECUTE TOF
 1759  011700  004737  005246                  JSR PC, QUIET
 1760          177777                   $BRJMP=-1
 1761  011704                           EXIT TST
  (3)  011704  104432                          TRAP    C$EXIT
```

```
  (3)  011706  000110                          .WORD   L10012-.
1762
1763  011710  051120  047111  040524  PRTCHR: .ASCIZ /PRINTABLE CHARACTERS TEST 3/ <12><12><12>
1764  011747     124  051505  020124  DONE:   .ASCII  /TEST COMPLETE/<12>
1765  011765     033     120     061  PORSQ:  .BYTE   33,120,61,73,62,73,61,61,171,124,151,164        ; SEQS TO ASSIGN AND SEI
1766  012001     141     156     061          .BYTE   141,156,61,60,55,120,33,134,33,133,61,61,155
1767                                           .EVEN
1768
1769  012016                               ENDTST
  (3) 012016                               L10012:
  (3) 012016  104401                           TRAP    C$ETST
1770                                     .LIST BEX
1771  012020                               ENDMOD
```

```
                                        .SBTTL  NON-PRINTABLE CHARACTERS
                                
012020                                  BGNMOD
                                        ;++
                                        ;THIS TEST CHECKS FOR DETECTION OF ALL NON-PRINTABLE CHARACTERS
                                        ;EXCEPT FOR HORIZONTAL TAB, LINE FEED, VERTICAL TAB, FORM
                                        ;FEED, CARRIAGE RETURN, AND ESCAPE WHICH
                                        ;WOULD BE INTERPRETED AS VALID CONTROL CHARACTERS BY THE LN01. EACH
                                        ;CHARACTER WILL APPEAR ON THE PRINTER OUTPUT IN THE FORM OF ITS OCTAL
                                        ;CODE ACCOMPANIED WITH ITS MNEMONIC.
                                        ;122 OF THE TESTED CODE ARE THEN SENT FOLLOWED BY AN "@" CHAPACTER.
                                        ;IF THE CONTROL CODE HAS TAKEN UP A SPACE IN THE BUFFER THE "@" CHARACTER
                                        ;WILL APPEAR AT THE RIGHT MARGIN OF THE PAGE. IF THE CONTROL CODE HAS NOT
                                        ;TAKEN UP SPACE IN THE BUFFER THE "@" WILL APPEAR IMMEDIATELY TO THE RIGHT
                                        ;OF THE MNEMONIC FOR THE CONTROL CODE.
                                        ;
                                        ;       "NOTE"
                                        ;               IF THIS TEST IS ASSEMBLED AS PART OF VERSION 1 LN01 DIAGNOSTIC
                                        ;               IT WILL ALSO SEND ALL 8 BIT CONTROL CODES.  THE ASSEMBLER
                                        ;               WILL SEE THE CONDITIONAL STATEMENT: ".IF DF VERS.1",  AND, IF
                                        ;               VERS.1 IS DEFINED IN SKEL 2 THEN IT WILL ASSEMBLE THE CODE
                                        ;               FOR THE 8 BIT CONTROL CODES AS WELL AS THE NORMAL 7 BIT CONTROL
                                        ;               CODES.
                                        ;--
                                
012020                                  BGNTST  4
012020                                  T4::
                                        ;INDICATE TEST CURRENTLY BEING DONE
012020                                          OUTPUT #NONCHR,#70.
012062                                          LET R4 := #NONBUF
012062  012704  012461                          MOV     #NONBUF,R4
012066                                          LET WORK1 := #27.
012066  012737  000033  003122                  MOV     #27.,WORK1
                                        ;
                                        ; DO ONE LINE FOR EACH TABLE ENTRY
                                        ;
012074                                          LET COUNT := #0
012074  005037  002246                          CLR     COUNT
012100                                          INCR LINCNT FROM #1 TO WORK1 BY #1
012100  012737  000001  002242                  MOV     #1,LINCNT
012106  000402                                  BR      50131$
012110                                  50132$:
012110  005237  002242                          INC     LINCNT
012114                                  50131$:
012114  023737  002242  003122                  CMP     LINCNT,WORK1
012122  003063                                  BGT     50133$
012124                                              LET R3 := #OUTBUF
012124  012703  003124                          MOV     #OUTBUF,R3
                                        ;
                                        ; MOVE CODE AND MNEMONIC TO PRINT BUFFER
                                        ;
                                                INCR WORK FROM #1 TO #8. BY #1
012130  012737  000001  003120                  MOV     #1,WORK
012136  000402                                  BR      50134$
012140                                  50135$:
```

```
   (10) 012140  005237  003120                    INC     WORK
    (7) 012144                          50134$:
    (7) 012144  023727  003120  000010            CMP     WORK,#8.
    (9) 012152  003002                            BGT     50136$
  1814  012154                                    LET (R3)+ :B= (R4)+
    (4) 012154  112423                            MOVB    (R4)+,(R3)+
  1815  012156                                        ENDINC
    (5) 012156  000770                            BR      50135$
    (4) 012160                          50136$:
  1816
  1817                                  ;
  1818                                  ;  PUT 120 BYTES OF CODE INTO PRINT BUFFER
  1819                                  ;
  1820  012160                                        INCR WORK FROM #1 TO #122. BY #1
    (5) 012160  012737  000001  003120              MOV     #1,WORK
    (7) 012166  000402                              BR      50137$
    (6) 012170                          50140$:
   (10) 012170  005237  003120                      INC     WORK
    (7) 012174                          50137$:
    (7) 012174  023727  003120  000172              CMP     WORK,#122.
    (9) 012202  003002                              BGT     50141$
  1821  012204                                      LET (R3)+ :B= (R4)
    (4) 012204  111423                              MOVB    (R4),(R3)+
  1822  012206                                          ENDINC
    (5) 012206  000770                              BR      50140$
    (4) 012210                          50141$:
  1823
  1824                                  ;
  1825                                  ; FOLLOWED BY AN "@" CHARACTER AND A LF
  1826                                  ;
  1827  012210                                          LET (R3)+ :B= #100          ;"@"
    (4) 012210  112723  000100                    MOVB    #100,(R3)+
  1828  012214                                          LET (R3)+ :B= #12           ;LF
    (4) 012214  112723  000012                    MOVB    #12,(R3)+
  1829                                  ;
  1830                                  ; PRINT LINE OF OCTAL CODE, MNEMONIC, 120 BYTES(NONPRINTABLE CODE), AND "@"
  1831                                  ;
  1832  012220                                          OUTPUT  #OUTBUF,#132.
  1833  012262  004737  005246                         JSR PC,     QUIET
  1834  012266                                        LET R4 := R4 + #1
    (7) 012266  005204                              INC     R4
  1835  012270                                        ENDINC
    (5) 012270  000707                              BR      50132$
    (4) 012272                          50133$:
  1872  012272                                      LET OUTBUF :B= #14
    (4) 012272  112737  000014  003124              MOVB    #14,OUTBUF
  1873  012300                                      OUTPUT #OUTBUF,#1
  1874  012342  004737  005246                     JSR PC, QUIET
  1875  012346                                      EXIT    TST                     ;AND EXIT TEST
    (3) 012346  104432                              TRAP    C$EXIT
    (3) 012350  000476                              .WORD   L10013-.
  1876                                  ;
  1877                                  ;CHARACTER BUFFER AND TEST HEADER MESSAGE
  1878                                  ;
  1879                                  .NLIST BEX
  1880  012352  047516  026516  051120  NONCHR: .ASCII  /NON-PRINTABLE CHARACTERS TEST 4/<12>
```

```
1881  012412  020101  052506  046114          .ASCIZ  /A FULL LINE OF EACH CODE WILL BE SENT/<12>
1882
1883  012461     040  030060  020060  NONBUF: .ASCII  / 000 NUL/<0>
1884  012472  030040  030460  051440          .ASCII  / 001 SOH/<1>
1885  012503     040  030060  020062          .ASCII  / 002 STX/<2>
1886  012514  030040  031460  042440          .ASCII  / 003 ETX/<3>
1887  012525     040  030060  020064          .ASCII  / 004 EOT/<4>
1888  012536  030040  032460  042440          .ASCII  / 005 ENQ/<5>
1889  012547     040  030060  020066          .ASCII  / 006 ACK/<6>
1890  012560  030040  033460  041040          .ASCII  / 007 BEL/<7>
1891  012571     040  030460  020060          .ASCII  / 010 BS /<10>
1892  012602  030040  033061  051440          .ASCII  / 016 SO /<16>
1893  012613     040  030460  020067          .ASCII  / 017 SI /<17>
1894  012624  030040  030062  042040          .ASCII  / 020 DLE/<20>
1895  012635     040  031060  020061          .ASCII  / 021 XON/<21>
1896  012646  030040  031062  042040          .ASCII  / 022 DC2/<22>
1897  012657     040  031060  020063          .ASCII  / 023 XOF/<23>
1898  012670  030040  032062  042040          .ASCII  / 024 DC4/<24>
1899  012701     040  031060  020065          .ASCII  / 025 NAK/<25>
1900  012712  030040  033062  051440          .ASCII  / 026 SYN/<26>
1901  012723     040  031060  020067          .ASCII  / 027 ETB/<27>
1902  012734  030040  030063  041440          .ASCII  / 030 CAN/<30>
1903  012745     040  031460  020061          .ASCII  / 031 EM /<31>
1904  012756  030040  031063  051440          .ASCII  / 032 SUB/<32>
1905  012767     040  031460  020064          .ASCII  / 034 FS /<34>
1906  013000  030040  032463  043440          .ASCII  / 035 GS /<35>
1907  013011     040  031460  020066          .ASCII  / 036 RS /<36>
1908  013022  030040  033463  052440          .ASCII  / 037 US /<37>
1909  013033     040  033461  020067          .ASCII  / 177 DEL/<177>
1944                                          .EVEN
1945  013044  000000                  NUM:    .WORD 0
1946
1947                                  .LIST BEX
1948  013046                          ENDTST
 (3)  013046                          L10013:
 (3)  013046  104401                          TRAP    C$ETST
1949
1950  013050                          ENDMOD
```

```
                                .SBTTL PRINT CONTROL

                                BGNMOD
                                ;++
                                ;THIS TEST CHECKS THE PRINT CONTROL BY SENDING MORE THAN 132 CHARACTERS
                                ;BEFORE SENDING A CARRIAGE RETURN AND LINE FEED. ALL CHARACTERS IN EXCESS
                                ;OF 132 CHARACTERS SHOULD BE DISREGARDED.
                                ;
                                ;THREE LINES ARE PRINTED PER ITERATION, THESE LINES WILL IDENTIFY THE
                                ;COLUMN NUMBERS ACROSS THE PAGE. EXAMPLE :
                                ;          0         0         0.....................          :
                                ;          1         2         3.....................          3
                                ;12345678901234567890123456789 0.....................          012
                                ;
                                ;NOTICE THAT THE PRINTOUT SHOULD IDENTIFY 132 COLUMNS ACROSS THE PAGE.
                                ;
                                ;THIS OUTPUT IS REPEATED 13 TIMES.
                                ;--
013050                          BGNTST  5
 (3) 013050                     T5::
013050                          LET R1 := L$UNIT - #1
 (5) 013050  013701  002012             MOV     L$UNIT,R1
 (7) 013054  005301             DEC     R1
000001                          $BRJMP=1
013056                          INCR LUNIT FROM #0 TO R1 BY #1
 (5) 013056  005037  002260             CLR     LUNIT
 (7) 013062  000402             BR      50143$
 (6) 013064                     50142$:
 (8) 013064  005237  002260             INC     LUNIT
 (6) 013070                     50143$:
 (7) 013070  023701  002260             CMP     LUNIT,R1
 (9) 013074  003402             BLE     50144$
 (7) 013076  000137  013734             JMP     50145$
 (6) 013102                     50144$:
013102                          LET R2 := LUNIT SHIFT 1
 (5) 013102  013702  002260             MOV     LUNIT,R2
 (8) 013106  006302             ASL     R2
013110                          OUTPUI #PRTCTL,#56.,,LUNIT
013152                          LET COUNT := #13.
 (4) 013152  012737  000015  002246     MOV     #13.,COUNT
013160                          1$:
013160                          LET R5 := #TABLE1
 (4) 013160  012705  014210             MOV     #TABLE1,R5
013164                          WHILE (R5) NE #0 DO
 (4) 013164                     50146$:
 (6) 013164  005715             TST     (R5)
 (8) 013166  001002             BNE     .+6
 (9) 013170  000137  013236             JMP     50147$
013174                              OUTPUI (R5)+,#10.,,LUNIT
013234                          ENDDO
 (3) 013234  000753             BR      50146$
 (3) 013236                     50147$:
013236                          LET OUTBUF :B= #12
 (4) 013236  112737  000012  003124     MOVB    #12,OUTBUF
013244                          OUTPUI #OUTBUF,#1,,LUNIT

```

```
1988  013306                                                    LET R5 := #TABLE2
 (4)  013306  012705  014244                        MOV       #TABLE2,R5
1989  013312                                                    WHILE (R5) NE #0 DO
 (4)  013312                                50150$:
 (6)  013312  005715                                TST       (R5)
 (8)  013314  001002                                BNE       .+6
 (9)  013316  000137  013364                        JMP       50151$
1990  013322                                                        OUTPUI (R5)+,#10.,,LUNIT
1991  013362                                                    ENDDO
 (3)  013362  000753                                BR        50150$
 (3)  013364                                50151$:
1992  013364                                                    OUTPUI #OUTBUF,#1,,LUNIT
1993
1994  013426                                                    DECR LINCNT FROM #14. TO #1 BY #1
 (5)  013426  012737  000016  002242                MOV       #14.,LINCNT
 (7)  013434  000402                                BR        50153$
 (6)  013436                                50152$:
 (8)  013436  005337  002242                        DEC       LINCNT
 (6)  013442                                50153$:
 (7)  013442  023727  002242  000001                CMP       LINCNT,#1
 (9)  013450  002002                                BGE       50154$
 (7)  013452  000137  013522                        JMP       50155$
 (6)  013456                                50154$:
1995  013456                                                        OUTPUI #X11,#10.,,LUNIT
1996  013520                                                    ENDDECR
 (4)  013520  000746                                BR        50152$
 (4)  013522                                50155$:
1997  013522                                                    OUTPUI #OUTBUF,#1,,LUNIT
1998  013564                                                    OUTPUI #OUTBUF,#1,,LUNIT
1999  013626                                                    LET COUNT := COUNT - #1
 (7)  013626  005337  002246                        DEC       COUNT
2000  013632                                                    IF COUNT GT #0 THEN
 (6)  013632  005737  002246                        TST       COUNT
 (8)  013636  003002                                BGT       .+6
 (9)  013640  000137  013650                        JMP       50156$
2001  013644  000137  013160                                      JMP 1$
2002  013650                                                    ENDIF
 (4)  013650                                50156$:
2003  013650  004737  005246                                  JSR PC, QUIET
2004  013654                                                    LET OUTBUF :B= #14
 (4)  013654  112737  000014  003124                MOVB      #14,OUTBUF
2005  013662                                                    OUTPUI  #OUTBUF,#1,,LUNIT
2006  013724  004737  005246                                  JSR PC,QUIET
2007  013730                                ENDINCR
 (4)  013730  000137  013064                        JMP       50142$
 (4)  013734                                50145$:
2008       177777                          $BRJMP=-1
2009  013734                                EXIT TST
 (3)  013734  104432                                TRAP      C$EXIT
 (3)  013736  000342                                .WORD     L10014-.
2010                                       .NLIST BEX
2011  013740  051120  047111  020124       PRTCTL: .ASCII /PRINT CONTROL TEST  5/ <12>
2012  013766  044123  052517  042114               .ASCIZ  /SHOULD SHOW 132 COLUMNS PRINTED/<12><12><15>
2013
2014  014031     040  020040  020040       X0:     .ASCII  /         0/
2015  014043     040  020040  020040       X1:     .ASCII  /         1/
```

```
 2016  014055      040  020040  020040  X2:     .ASCII  /          2/
 2017  014067      040  020040  020040  X3:     .ASCII  /          3/
 2018  014101      040  020040  020040  X4:     .ASCII  /          4/
 2019  014113      040  020040  020040  X5:     .ASCII  /          5/
 2020  014125      040  020040  020040  X6:     .ASCII  /          6/
 2021  014137      040  020040  020040  X7:     .ASCII  /          7/
 2022  014151      040  020040  020040  X8:     .ASCII  /          8/
 2023  014163      040  020040  020040  X9:     .ASCII  /          9/
 2024
 2025  014175      061  031462  032464   11:    .ASCII  /1234567890/
 2026
 2027          014210                   .EVEN
 2028  014210  014031  014031  014031  TABLE1: .WORD   X0,X0,X0,X0,X0,X0,X0,X0,X0,X1,X1,X1,X1,0
 2029  014244  014043  014055  014067  TABLE2: .WORD   X1,X2,X3,X4,X5,X6,X7,X8,X9,X0,X1,X2,X3,0
 2030                                           .EVEN
 2031
 2032                                   .LIST BEX
 2033  014300                           ENDTST
   (3) 014300                           L10014:
   (3) 014300  104401                           TRAP    C$ETST
 2034  014302                           ENDMOD
```

```
                                        .SBTTL  MULTIPLE LINE ADVANCE

                                        BGNMOD
                                        ;++
                                        ;THIS TEST CHECKS THE MULTIPLE LINE ADVANCE OF THE LN01. A LINE OF
                                        ;NUMBERS IS SENT AND THEN A NUMBER OF LINE FEEDS ARE SENT. THUS THE
                                        ;NUMBER PRINTED WILL INDICATE THE NUMBER OF BLANK LINES FOLLOWING THAT
                                        ;LINE. THE NUMBER OF LINES IS VARIED BETWEEN 2 AND 7 AND A LINE OF
                                        ;ALL 0'S WILL INDICATE THE END OF THE TEST  SEQUENCE.
                                        ;--


014302                                  BGNTST  6
(3)  014302                             T6::

                                        ;PRINT TEST IDENTIFICATION

014302                                  OUTPUT #MULINE,#86.

014344                                  LET STACHR := #TABSTR           ;OUTPUT CHARACTERS
(4)  014344  012737  014622  014620             MOV     #TABSTR,STACHR

014352                                  REPEAT
(3)  014352                             50157$:
014352                                      LET LINCNT :B= @STACHR              ;GET A CHARACTER TO OUTPUT
(4)  014352  117737  000242  002242            MOVB    @STACHR,LINCNT
014360                                      LET LINCNT := LINCNT AND #7 ;MAKE THE ASCII TO OCTAL
(7)  014360  013746  002242                     MOV     LINCNT,-(SP)
(7)  014364  042716  000007                     BIC     #7,(SP)
(7)  014370  042637  002242                     BIC     (SP)+,LINCNT
014374                                      LET R3 := #OUTBUF               ;SET UP OUTPUT BUFFER
(4)  014374  012703  003124                     MOV     #OUTBUF,R3
014400                                      INCR CCNT FROM #1 TO #132. BY #1
(5)  014400  012737  000001  002250            MOV     #1,CCNT
(7)  014406  000402                             BR      50160$
(6)  014410                             50161$:
(10) 014410  005237  002250                     INC     CCNT
(7)  014414                             50160$:
(7)  014414  023727  002250  000204             CMP     CCNT,#132.
(9)  014422  003003                             BGT     50162$
014424                                          LET (R3)+ :B= @STACHR           ;PUT CHARACTER  IN OUTPUT BUFFER
(4)  014424  117723  000170                     MOVB    @STACHR,(R3)+
014430                                      ENDINC
(5)  014430  000767                             BR      50161$
(4)  014432                             50162$:
014432                                      LET R4 := #0
(4)  014432  005004                             CLR     R4
014434                                      WHILE R4 NE LINCNT DO
(4)  014434                             50163$:
(6)  014434  020437  002242                     CMP     R4,LINCNT
(10) 014440  001404                             BEQ     50164$
014442                                          LET (R3)+ :B= #12               ;FILL WITH LINE  EEDS
(4)  014442  112723  000012                     MOVB    #12,(R3)+
014446                                          LET R4 := R4 + #1
(7)  014446  005204                             INC     R4
014450                                      ENDDO
```

```
  (4)  014450  000771                        BR      50163$
  (3)  014452                          50164$:
2071
2072
2073
2074  014452                              LET R4 := LINCNT + #132.          ;NUMBER OF CHARACTERS TO OUTPUT
  (5)  014452  013704  002242                MOV     LINCNT,R4
  (7)  014456  062704  000204                ADD     #132.,R4
2075  014462                              LET STACHR := STACHR + #1         ; UPDATE CHARACTER COUNT
  (7)  014462  005237  014620                INC     STACHR
2076  014466                              OUTPUT #OUTBUF,R4                 ;OUTPUT THE LINE
2077  014526  004737  005246                JSR PC,     QUIET
2078
2079  014532                              UNTIL LINCNT EQ #0
  (3)  014532  005737  002242                TST     LINCNT
  (7)  014536  001305                        BNE     50157$
2080  014540                              LET OUTBUF :B= #14
  (4)  014540  112737  000014  003124        MOVB    #14,OUTBUF
2081  014546                              OUTPUT #OUTBUF,#1
2082  014610  004737  005246              JSR PC,QUIET
2083  014614                              EXIT TST
  (3)  014614  104432                        TRAP    C$EXIT
  (3)  014616  000150                        .WORD   L10015-.
2084
2085
2086  014620  000000                      STACHR: .WORD 0
2087                                      .NLIST BEX
2088  014622  033462  033062  033463      TABSTR: .ASCIZ  /272637463540/
2089  014637     115  046125  044524      MULINE: .ASCII  /MULTIPLE LINE ADVANCE TEST  6/<12>
2090  014675     116  046525  042502              .ASCIZ  /NUMBERS PRINTED REPRESENT # LINES TO NEXT LINE PRINTED/<12><12>
2091
2092
2093
2094                                      .EVEN
2095                                      .LIST BEX
2096
2097  014766                              ENDTST
  (3)  014766                          L10015:
  (3)  014766  104401                        TRAP    C$ETST
2098  014770                              ENDMOD
```

```
2100                                    .SBTTL DVSTRIKE
2101                                    ;MODULE DVSTR1.P11
2102  014770                            BGNMOD
2103                                    ;++
2104                                    ;THIS TEST WILL VERIFY CORRECT OPERATION OF THE PRINTER WHILE OPERATING
2105                                    ;JUST WITHIN OVERSTRIKE, LINE BUFFER AND PAGE BUFFER LIMITS.
2106                                    ;A TOTAL OF 4 OVERSTRIKES ON ONE LINE IS ALLOWED AND IS TESTED.
2107                                    ;WE ARE LIMITED WHEN SENDING CHARACTERS TO THE LINE BUFFER.  IF YOU USE THE DEFAULT
2108                                    ;FONT AND PRINT SEQUENTIAL LINES, THE LIMIT IS 148 TOTAL CHARACTERS PER LINE COUNTING
2109                                    ;ALL CHARACTERS THAT ARE ACTUALLY TRANSMITTED FROM THE LP11 HOST INTERFACE.
2110                                    ;THIS LIMIT IS TESTED AS WELL.
2111                                    ;THE PAGE BUFFER LIMIT IS 10,000 CHARACTERS AND THIS TEST OPERATES "JUST"
2112                                    ;WITHIN THAT LIMIT, USING OVER 9,900 CHARACTERS.
2113                                    ;
2114                                    ;       NOTE:
2115                                    ;               THIS TEST IN A SOMEWHAT MODIFIED FORM IS BEING USED ALSO FOR
2116                                    ;               THE VAX VERSION OF THE LN01 EXTENDED DIAGNOSTIC.
2117                                    ;
2118                                    ;       IN THE PDP-11 FORM (WHICH IS THIS MODULE) THE PAGE BUFFER ENDS UP CONTAINING
2119                                    ;       A TOTAL OF 9,939 CHARACTERS. THIS INCLUDES THE EVER PRESENT 7 CHARACTERS FOR
2120                                    ;       EACH LINE THAT THE PRINTER USES.
2121                                    ;
2122                                    ;       IN THE VAX FORM (WHICH IS USED IN THE VAX PRINTER DIAGNOSTIC) THE PAGE BUFFER
2123                                    ;       ENDS UP CONTAINING A TOTAL OF 9,949 CHARACTERS. THIS INCLUDES 7 THE EVER PRESENT
2124                                    ;       CHARACTERS FOR EACH LINE THAT THE PRINTER USES.
2125                                    ;
2126                                    ;       NOTE:
2127                                    ;       ONE THING NOT COUNTED HERE IS THE POSSIBILITY OF THE PRINTER USING ONE EXTRA
2128                                    ;       SPACE PER LINE FEED IN ITS BUFFER WHEN IN "LINE FEED-NEWLINE MODE".
2129                                    ;       THE WORST CASE IF THIS SHOULD HAPPEN IS THAT 66 EXTRA CHARACTERS WOULD END
2130                                    ;       UP ADDED TO THE PREVIOUS PAGE BUFFER TOTAL. I DON'T BELIEVE THAT EVEN THIS
2131                                    ;       CASE WOULD CAUSE A FAILURE. HOWEVER, THE POSSIBILITY EXISTS BECAUSE THE PAGE BUF
2132                                    ;       WOULD SLIGHTLY EXCEED ITS LIMIT ON THE PDP11 VERSION AS WELL AS THE VAX VERSION.
2133                                    ;--
2134
2135  014770                            BGNTST 7
 (3)  014770                            T7::
2136
2137  014770                                    LET R4 := #OUTBUF                     ; ADDRESS OF BUFFER
 (4)  014770  012704  003124                      MOV     #OUTBUF,R4
2138  014774                                    LET (R4)+ :B= #116                    ; "N"
 (4)  014774  112724  000116                      MOVB    #116,(R4)+
2139  015000                                    LET (R4)+ :B= #132                    ; "Z"
 (4)  015000  112724  000132                      MOVB    #132,(R4)+
2140  015004                                    LET (R4)+ :B= #116                    ; "N"
 (4)  015004  112724  000116                      MOVB    #116,(R4)+
2141  015010                                    LET (R4)+ :B= #CR                     ; CARRIAGE RET.
 (4)  015010  112724  000015                      MOVB    #CR,(R4)+
2142  015014                                    LET (R4)+ :B= #LF                     ; LINE FEED
 (4)  015014  112724  000012                      MOVB    #LF,(R4)+
2143  015020                                    LET (R4)+ :B= #FF                     ; FORM FEED
 (4)  015020  112724  000014                      MOVB    #FF,(R4)+
2144  015024                                    LET (R4)+ :B= #40                     ; SPACE
 (4)  015024  112724  000040                      MOVB    #40,(R4)+
2148  015030                                    OUTPUT #DVSTR,#15.                    ; TEST ID WITHOUT LINE FEED
2149  015072                                    OUTPUT #OUTBUF+6,#1,,#90.             ; DO 90 SPACES TO TAKE UP LINE BUFFER SP
```

```
2150  015134                                  OUTPUT #OUTBUF+4,#1                     ; DO LINE FEED
2151  015176                                  OUTPUT #ZN1,#3                          ;1ST OVERPRINT LINE
2152  015240                                  OUTPUT #OUTBUF+6,#1,,#90.               ; 90 SPACES
2153  015302                                  OUTPUT #OUTBUF+4,#1                     ; LINE FEED
2154  015344                                  OUTPUT #ZN2,#7.                         ;2ND      "       "
2155  015406                                  OUTPUT #OUTBUF+6,#1,,#90.               ; DO 90 SPACES TO TAKE UP LINE BUFFER SP
2156  015450                                  OUTPUT #OUTBUF+4,#1                     ; DO LINE FEED
2157  015512                                  OUTPUT #ZN3,#12.                        ;3RD      "       "
2158  015554                                  OUTPUT #OUTBUF+6,#1,,#90.               ; DO 90 SPACES TO TAKE UP LINE BUFFER SP
2159  015616                                  OUTPUT #OUTBUF+4,#1                     ; DO LINE FEED
2160  015660                                  OUTPUT #ZN4,#18.                        ;4TH      "       "
2161  015722                                  OUTPUT #OUTBUF+6,#1,,#90.               ; DO 90 SPACES TO TAKE UP LINE BUFFER SP
2162  015764                                  OUTPUT #OUTBUF+4,#1                     ; DO LINE FEED
2163  016026  004737  005246                  JSR PC, QUIET
2164  016032                                  INCR COUNT FROM #1 TO #61. BY #1        ; 61 LINES OF OVERPRINTING
 (5)  016032  012737  000001  002246          MOV     #1,COUNT
 (7)  016040  000402                          BR      50165$
 (6)  016042                          50166$:
 (10) 016042  005237  002246                  INC     COUNT
 (7)  016046                          50165$:
 (7)  016046  023727  002246  000075          CMP     COUNT,#61.
 (9)  016054  003135                          BGT     50167$
2165  016056                                          OUTPUT #OUTBUF,#2,,#36.         ; LINE OF NZ
2166  016120                                          OUTPUT #OUTBUF+3,#1             ; CARRIAGE RET. (NO LF)
2167  016162                                          OUTPUT #OUTBUF+1,#2,,#36.       ; LINE OF ZN
2168  016224                                          IF COUNT EQ #61. THEN           ; IS THIS THE LAST LINE?
 (6)  016224  023727  002246  000075          CMP     COUNT,#61.
 (10) 016232  001022                          BNE     50170$
2169  016234                                                  OUTPUT #OUTBUF+3,#1     ; CARRIAGE RETURN ONLY ON LAST LINE
2170  016276                                          ELSE
 (4)  016276  000421                          BR      50171$
 (3)  016300                          50170$:
2171  016300                                                  OUTPUT #OUTBUF+3,#2     ; CR AND LF ON EVERY OTHER LINE
2172  016342                                          ENDIF
 (4)  016342                          50171$:
2173  016342  004737  005246                        JSR PC,QUIET
2174  016346                                  ENDINC                                  ; WHEN DONE FALL THROUGH
 (5)  016346  000635                          BR      50166$
 (4)  016350                          50167$:
2175  016350                                  OUTPUT #OUTBUF+5,#1                     ; DO FORM FEED
2176  016412  004737  005246                  JSR PC,QUIET
2177  016416                          EXIT TST
 (3)  016416  104432                          TRAP    C$EXIT
 (3)  016420  000100                          .WORD   L10016-.
2178                                    .NLIST BEX
2179  016422  000000                    ZNUM:   .WORD 0
2180  016424  000000                    SPANUM: .WORD 0
2181  016426  000000                    NEWSPA: .WORD 0
2182  016430  053104  052123  044522    DVSTR:  .ASCIZ  /DVSTRIKE TEST 7/
2183  016450  006532     116            ZN1:    .ASCII  /Z/<15>/N/
2184  016453     132  006532  006516     ZN2:    .ASCII  /ZZ/<15>/N/<15><40>/N/
2185  016462  055132  006532  006516     ZN3:    .ASCII  /ZZZ/<15>/N/<15><40>/N/<15><40><40>/N/
2186  016476  055132  055132  047015     ZN4:    .ASCII  /ZZZZ/<15>/N/<15><40>/N/<15><40><40>/N/<15><40><40><40>/N/
2187                                    .EVEN
2188  016520                          ENDTST
 (3)  016520                          L10016:
```

```
  (3)  016520  104401                        TRAP    C$ETST
 2189
 2190  016522                               .LIST BEX
 2191                                       ENDMOD
 2192
 2193  016522                               BGNMOD
 2194                                       ;++
 2195                                       ;THIS TEST CHECKS THE OPERATION OF THE
 2196                                       ;PRINTER READY INTERLOCK SWITCHES.
 2197                                       ;MANUAL INTERVENTION IS USED TO
 2198                                       ;OPEN THE INTERLOCKS TO PRODUCE FAULTS
 2199                                       ;IN THE PRINTER AFTER WHICH THE RESULTANT ERROR
 2200                                       ;INDICATION IS VERIFIED.
 2201                                       ;--
 2202
 2203  016522                               BGNTST  8.
  (3)  016522                               T8::
 2204                                       ;DETERMINE IF MANUAL INTERVENTION IS ALLOWED
 2205  016522                                       MANUAL
  (3)  016522  104450                         TRAP    C$MANI
 2206  016524                               BCOMPLETE 11$
  (2)  016524  103402                         BCS     11$
 2207  016526                               EXIT TST
  (3)  016526  104432                         TRAP    C$EXIT
  (3)  016530  002726                         .WORD   L10017-.
 2208                                       ;EXIT TEST IF MANUAL INTERVENTION TESTS ARE NOT SPECIFIED
 2209  016532                               11$:    IF INHINT EQ #0 THEN
  (6)  016532  005737  002234                 TST     INHINT
 (10)  016536  001002                         BNE     50172$
 2210  016540                               EXIT TST
  (3)  016540  104432                         TRAP    C$EXIT
  (3)  016542  002714                         .WORD   L10017-.
 2211  016544                                ENDIF
  (4)  016544                           50172$:
 2212  016544                               LET FLAG := #0
  (4)  016544  005037  002240                 CLR     FLAG
 2213  016550                               LET R1 := L$UNIT - #1
  (5)  016550  013701  002012                 MOV     L$UNIT,R1
  (7)  016554  005301                         DEC     R1
 2214
 2215                                       ;CHECK FOR ERROR IN EACH PRINTER UNDER TEST
 2216  016556                                   INCR LUNIT FROM #0 TO R1 BY #1
  (5)  016556  005037  002260                 CLR     LUNIT
  (7)  016562  000402                         BR      50173$
  (6)  016564                           50174$:
 (10)  016564  005237  002260                 INC     LUNIT
  (7)  016570                           50173$:
  (7)  016570  023701  002260                 CMP     LUNIT,R1
  (9)  016574  003020                         BGT     50175$
 2217  016576                               LET R2 := LUNIT SHIFT 1
  (5)  016576  013702  002260                 MOV     LUNIT,R2
  (8)  016602  006302                         ASL     R2
 2218  016604                               IF #BIT15 SETIN @LPCSR(R2) THEN
  (6)  016604  032772  100000  002322         BIT     #BIT15,@LPCSR(R2)
 (10)  016612  001410                         BEQ     50176$
 2219  016614                                       LET ERRTBL(R2) := ERRTBL(R2) + #1
```

```
  (7)  016614  005262  003060                    INC     ERRTBL(R2)
2220   016620                                    ERRHRD 6, CSRERR
  (4)  016620  104456                            TRAP    C$ERHRD
  (5)  016622  000006                            .WORD   6
  (5)  016624  003352                            .WORD   CSRERR
  (5)  016626  000000                            .WORD   0
2221   016630                                    LET @LPCSR(R2) := #0
  (4)  016630  005072  002322                    CLR     @LPCSR(R2)
2222   016634                                    ENDIF
  (4)  016634                              50176$:
2223   016634                                  ENDINC
  (5)  016634  000753                            BR      50174$
  (4)  016636                              50175$:
2224                                       :CHECK FOR READY IN EACH PRINTER UNDER TEST
2225   016636                                  INCR LUNIT FROM #0 TO R1 BY #1
  (5)  016636  005037  002260                    CLR     LUNIT
  (7)  016642  000402                            BR      50177$
  (6)  016644                              50200$:
  (10) 016644  005237  002260                    INC     LUNIT
  (7)  016650                              50177$:
  (7)  016650  023701  002260                    CMP     LUNIT,R1
  (9)  016654  003021                            BGT     50201$
2226   016656                                    LET R2 := LUNIT SHIFT 1
  (5)  016656  013702  002260                    MOV     LUNIT,R2
  (8)  016662  006302                            ASL     R2
2227   016664                                    LET L$LUN := LUNIT
  (4)  016664  013737  002260  002074            MOV     LUNIT,L$LUN
2228   016672                                    IF #BIT07 NOTSETIN @LPCSR(R2) THEN
  (6)  016672  032772  000200  002322            BIT     #BIT07,@LPCSR(R2)
  (10) 016700  001006                            BNE     50202$
2229   016702                                    LET ERRTBL(R2) := ERRTBL(R2) + #1
  (7)  016702  005262  003060                    INC     ERRTBL(R2)
2230   016706                                    ERRHRD 7, RDYERR
  (4)  016706  104456                            TRAP    C$ERHRD
  (5)  016710  000007                            .WORD   7
  (5)  016712  003370                            .WORD   RDYERR
  (5)  016714  000000                            .WORD   0
2231   016716                                    ENDIF
  (4)  016716                              50202$:
2232   016716                                  ENDINC
  (5)  016716  000752                            BR      50200$
  (4)  016720                              50201$:
2233                                       ;
2234                                       ; PRINT TEST NAME
2235                                       ;
2236   016720                                  OUTPUT #INTLK,#29.
2237                                       ;VERIFY OPERATION OF PAPER OUT INTERLOCK SWITCH
2238                                       ;HARD CODED INCREMENT LOOP
2239                                       ;
2240   016762                                    LET ERRFLG := #0
  (4)  016762  005037  002304                    CLR     ERRFLG
2241   016766  005037  002260              CLR LUNIT
2242   016772  000405                      BR  1$
2243   016774                             2$:
2244   016774  005237  002260                INC LUNIT
2245   017000                              LET R2 := LUNIT SHIFT 1
```

```
   (5)  017000  013702  002260                    MOV     LUNIT,R2
   (8)  017004  006302                            ASL     R2
 2246   017006                                1$:
 2247   017006  023701  002260                        CMP LUNIT,R1
 2248   017012  003402                            BLE 3$
 2249   017014  000137  017546                    JMP 4$
 2250   017020                                3$:
 2251   017020                                    LET FLAG := #0
   (4)  017020  005037  002240                    CLR     FLAG
 2252   017024                                    PRINTF #PAPRSW
   (7)  017024  012746  020400                    MOV     #PAPRSW,-(SP)
   (6)  017030  012746  000001                    MOV     #1,-(SP)
   (3)  017034  010600                            MOV     SP,R0
   (4)  017036  104417                            TRAP    C$PNTF
   (4)  017040  062706  000004                    ADD     #4,SP
 2253   017044                                    PRINTF #PAPSW1,LUNIT
   (8)  017044  013746  002260                    MOV     LUNIT,-(SP)
   (7)  017050  012746  020455                    MOV     #PAPSW1,-(SP)
   (6)  017054  012746  000002                    MOV     #2,-(SP)
   (3)  017060  010600                            MOV     SP,R0
   (4)  017062  104417                            TRAP    C$PNTF
   (4)  017064  062706  000006                    ADD     #6,SP
 2254   017070                                    PRINTF #PAPSW2
   (7)  017070  012746  020535                    MOV     #PAPSW2,-(SP)
   (6)  017074  012746  000001                    MOV     #1,-(SP)
   (3)  017100  010600                            MOV     SP,R0
   (4)  017102  104417                            TRAP    C$PNTF
   (4)  017104  062706  000004                    ADD     #4,SP
 2255   017110                                    GMANIL READY, FLAG, 100000, YES
   (3)  017110  104443                            TRAP    C$GMAN
   (3)  017112  000404                            BR      10000$
   (4)  017114  002240                            .WORD   FLAG
   (5)  017116  000130                            .WORD   T$CODE
   (5)  017120  006752                            .WORD   READY
   (5)  017122  100000                            .WORD   100000
   (3)  017124                           10000$:
 2256   017124                                    LET LINCNT := #2        ; LINE COUNT WILL ALLOW FOR 3 PAGES OF PAPER
   (4)  017124  012737  000002  002242            MOV     #2,LINCNT
 2257   017132                                    LET ERRFLG := #0
   (4)  017132  005037  002304                    CLR     ERRFLG
 2258   017136                                    REPEAT
   (3)  017136                           50203$:
 2259   017136                                        OUTPUT #PAPTST,#15.,#5$,LUNIT
 2260   017200                                        LET LINCNT := LINCNT + #1
   (7)  017200  005237  002242                INC     LINCNT
 2261   017204                                        IF LINCNT EQ #65. OR LINCNT EQ #130. OR LINCNT EQ #195. THEN
   (6)  017204  023727  002242  000101            CMP     LINCNT,#65.
   (8)  017212  001410                            BEQ     50204$
   (6)  017214  023727  002242  000202            CMP     LINCNT,#130.
   (8)  017222  001404                            BEQ     50204$
   (6)  017224  023727  002242  000303            CMP     LINCNT,#195.
  (10)  017232  001024                            BNE     50205$
   (6)  017234                           50204$:
 2262   017234                                        LET OUTBUF := #14                       ; FORM FEED
   (4)  017234  012737  000014  003124            MOV     #14,OUTBUF
 2263   017242                                        OUTPUT #OUTBUF,#1,5$,LUNIT              ; OUTPUT THE FORM FEED
```

```
2264  017304                                        ENDIF
 (4)  017304                                50205$:
2265  017304                                        UNTIL LINCNT EQ #260. OR ERRFLG NE #0          ; UNTIL FOUR PAGES PRINTED OR IN
 (4)  017304  023727  002242  000404               CMP     LINCNT,#260.
 (6)  017312  001403                               BEQ     50206$
 (4)  017314  005737  002304                       TST     ERRFLG
 (8)  017320  001706                               BEQ     50203$
 (4)  017322                                50206$:
2266  017322                                        IF ERRFLG EQ #0 THEN
 (6)  017322  005737  002304                       TST     ERRFLG
 (10) 017326  001035                               BNE     50207$
2267  017330                                          LET OUTBUF := #14                            ; FORM FEED
 (4)  017330  012737  000014  003124               MOV     #14,OUTBUF
2268  017336                                          OUTPUT #OUTBUF,#1,5$,LUNIT                   ; OUTPUT THE FF
2269  017400                                          ERRHRD 8,PAPSWI
 (4)  017400  104456                               TRAP    C$ERHRD
 (5)  017402  000010                               .WORD   8
 (5)  017404  003412                               .WORD   PAPSWI
 (5)  017406  000000                               .WORD   0
2270  017410                                          LET ERRTBL(R2) := ERRTBL(R2) + #1
 (7)  017410  005262  003060                       INC     ERRTBL(R2)
2271  017414                                          INLINE      <JMP    11002$>
 (2)  017414  000137  017426                       JMP     11002$
2272  017420                                        ELSE
 (4)  017420  000402                               BR      50210$
 (3)  017422                                50207$:
2273  017422                                          LET ERRFLG := #0
 (4)  017422  005037  002304                       CLR     ERRFLG
2274  017426                                        ENDIF
 (4)  017426                                50210$:
2275  017426                                11002$: PRINTF #PAPRDY,LUNIT
 (8)  017426  013746  002260                       MOV     LUNIT,-(SP)
 (7)  017432  012746  020577                       MOV     #PAPRDY,-(SP)
 (6)  017436  012746  000002                       MOV     #2,-(SP)
 (3)  017442  010600                               MOV     SP,R0
 (4)  017444  104417                               TRAP    C$PNTF
 (4)  017446  062706  000006                       ADD     #6,SP
2276  017452                                        LET FLAG := #0
 (4)  017452  005037  002240                       CLR     FLAG
2277  017456                                        GMANIL READY,FLAG,100000,YES
 (3)  017456  104443                               TRAP    C$GMAN
 (3)  017460  000404                               BR      10001$
 (4)  017462  002240                               .WORD   FLAG
 (5)  017464  000130                               .WORD   T$CODE
 (5)  017466  006752                               .WORD   READY
 (5)  017470  100000                               .WORD   100000
 (3)  017472                                10001$:
2278  017472                                        LET R2 := LUNIT SHIFT 1
 (5)  017472  013702  002260                       MOV     LUNIT,R2
 (8)  017476  006302                               ASL     R2
2279  017500                                        LET @LPCSR(R2) := #0                         ; RESET THE LP CSR
 (4)  017500  005072  002322                       CLR     @LPCSR(R2)
2280  017504  000137  016774                       JMP 2$
2281                                       ;EXPECTED ERROR HANDLER.
2282                                       ;JUST SET EXPECTED ERROR INDICATOR.
2283                                       ;
```

```
2284  017510                                  5$:     LET ERRFLG := #1
 (4)  017510  012737  000001  002304                 MOV     #1,ERRFLG
2285  017516                                          LET ERRCOD := #0
 (4)  017516  005037  002302                         CLR     ERRCOD
2286  017522                                          LET STATUS(R2) := STATUS(R2) CLR.BY #ERROR!ACTIVE
 (7)  017522  042762  120000  002456                 BIC     #ERROR!ACTIVE,STATUS(R2)
2287  017530                                          LET CURCNT(R2) := #0    ; CLEAN UP THE DRIVER PARAMETERS
 (4)  017530  005062  002716                         CLR     CURCNT(R2)
2288  017534                                          LET CURADD(R2) := #0
 (4)  017534  005062  002516                         CLR     CURADD(R2)
2289  017540                                          LET REPCNT(R2) := #0
 (4)  017540  005062  002616                         CLR     REPCNT(R2)
2290  017544  000207                                 RTS PC                          ;AND RETURN
2291                                          ;VERIFY OPERATION OF PAPER TRAY HANDLE INTERLOCK SWITCH.
2292  017546                                  4$:    INCR LUNIT FROM #0 TO R1 BY #1
 (5)  017546  005037  002260                         CLR     LUNIT
 (7)  017552  000402                                 BR      50211$
 (6)  017554                                 50212$:
 (10) 017554  005237  002260                         INC     LUNIT
 (7)  017560                                 50211$:
 (7)  017560  023701  002260                         CMP     LUNIT,R1
 (9)  017564  003117                                 BGT     50213$
2293  017566                                          LET R2 := LUNIT SHIFT 1
 (5)  017566  013702  002260                         MOV     LUNIT,R2
 (8)  017572  006302                                 ASL     R2
2294  017574                                          LET L$LUN := LUNIT
 (4)  017574  013737  002260  002074                 MOV     LUNIT,L$LUN
2295  017602                                          LET FLAG := #0
 (4)  017602  005037  002240                         CLR     FLAG
2296  017606                                          PRINTF #HANRSW
 (7)  017606  012746  020706                         MOV     #HANRSW,-(SP)
 (6)  017612  012746  000001                         MOV     #1,-(SP)
 (3)  017616  010600                                 MOV     SP,R0
 (4)  017620  104417                                 TRAP    C$PNTF
 (4)  017622  062706  000004                         ADD     #4,SP
2297  017626                                          PRINTF #HANSW1,LUNIT
 (8)  017626  013746  002260                         MOV     LUNIT,-(SP)
 (7)  017632  012746  020770                         MOV     #HANSW1,-(SP)
 (6)  017636  012746  000002                         MOV     #2,-(SP)
 (3)  017642  010600                                 MOV     SP,R0
 (4)  017644  104417                                 TRAP    C$PNTF
 (4)  017646  062706  000006                         ADD     #6,SP
2298  017652                                          PRINTF #HANSW2
 (7)  017652  012746  021055                         MOV     #HANSW2,-(SP)
 (6)  017656  012746  000001                         MOV     #1,-(SP)
 (3)  017662  010600                                 MOV     SP,R0
 (4)  017664  104417                                 TRAP    C$PNTF
 (4)  017666  062706  000004                         ADD     #4,SP
2299  017672                                          GMANIL READY, FLAG, 100000, YES
 (3)  017672  104443                                 TRAP    C$GMAN
 (3)  017674  000404                                 BR      10002$
 (4)  017676  002240                                 .WORD   FLAG
 (5)  017700  000130                                 .WORD   T$CODE
 (5)  017702  006752                                 .WORD   READY
 (5)  017704  100000                                 .WORD   100000
 (3)  017706                                 10002$:
```

```
2300  017706                                    IF #BIT15 SETIN @LPCSR(R2) THEN
 (6)  017706  032772  100000  002322           BIT     #BIT15,@LPCSR(R2)
 (10) 017714  001431                           BEQ     50214$
2301  017716                                    PRINTF #HANRDY
 (7)  017716  012746  021114                   MOV     #HANRDY,-(SP)
 (6)  017722  012746  000001                   MOV     #1,-(SP)
 (3)  017726  010600                           MOV     SP,R0
 (4)  017730  104417                           TRAP    C$PNTF
 (4)  017732  062706  000004                   ADD     #4,SP
2302  017736                                    PRINTF #HANRD1,LUNIT
 (8)  017736  013746  002260                   MOV     LUNIT,-(SP)
 (7)  017742  012746  021201                   MOV     #HANRD1,-(SP)
 (6)  017746  012746  000002                   MOV     #2,-(SP)
 (3)  017752  010600                           MOV     SP,R0
 (4)  017754  104417                           TRAP    C$PNTF
 (4)  017756  062706  000006                   ADD     #6,SP
2303  017762                                    GMANIL READY, FLAG, 100000, YES
 (3)  017762  104443                           TRAP    C$GMAN
 (3)  017764  000404                           BR      10003$
 (4)  017766  002240                           .WORD   FLAG
 (5)  017770  000130                           .WORD   T$CODE
 (5)  017772  006752                           .WORD   READY
 (5)  017774  100000                           .WORD   100000
 (3)  017776                          10003$:
2304  017776                                    ELSE
 (4)  017776  000411                           BR      50215$
 (3)  020000                          50214$:
2305  020000                                    LET ERRTBL(R2) :- ERRTBL(R2) + #1
 (7)  020000  005262  003060                   INC     ERRTBL(R2)
2306  020004                                    LET L$LUN := LUNIT
 (4)  020004  013737  002260  002074           MOV     LUNIT,L$LUN
2307  020012                                    ERRHRD 9,HANSWI
 (4)  020012  104456                           TRAP    C$ERHRD
 (5)  020014  000011                           .WORD   9
 (5)  020016  003455                           .WORD   HANSWI
 (5)  020020  000000                           .WORD   0
2308  020022                                    ENDIF
 (4)  020022                          50215$:
2309  020022                              ENDINC
 (5)  020022  000654                           BR      50212$
 (4)  020024                          50213$:
2310                                  ;VERIFY OPERATION OF FRONT DOOR INTERLOCK SWITCH.
2311  020024                              INCR LUNIT FROM #0 TO R1 BY #1
 (5)  020024  005037  002260                   CLR     LUNIT
 (7)  020030  000402                           BR      50216$
 (6)  020032                          50217$:
 (10) 020032  005237  002260                   INC     LUNIT
 (7)  020036                          50216$:
 (7)  020036  023701  002260                   CMP     LUNIT,R1
 (9)  020042  003107                           BGT     50220$
2312  020044                                    LET R2 := LUNIT SHIFT 1
 (5)  020044  013702  002260                   MOV     LUNIT,R2
 (8)  020050  006302                           ASL     R2
2313  020052                                    LET FLAG := #0
 (4)  020052  005037  002240                   CLR     FLAG
2314  020056                                    PRINTF #DOORSW,LUNIT
```

```
   (8)  020056  013746  002260                    MOV     LUNIT,-(SP)
   (7)  020062  012746  021247                    MOV     #DOORSW,-(SP)
   (6)  020066  012746  000002                    MOV     #2,-(SP)
   (3)  020072  010600                            MOV     SP,R0
   (4)  020074  104417                            TRAP    C$PNTF
   (4)  020076  062706  000006                    ADD     #6,SP
2315    020102                                    PRINTF #DOOSW1
   (7)  020102  012746  021324                    MOV     #DOOSW1,-(SP)
   (6)  020106  012746  000001                    MOV     #1,-(SP)
   (3)  020112  010600                            MOV     SP,R0
   (4)  020114  104417                            TRAP    C$PNTF
   (4)  020116  062706  000004                    ADD     #4,SP
2316    020122                                    GMANIL READY, FLAG, 100000, YES
   (3)  020122  104443                            TRAP    C$GMAN
   (3)  020124  000404                            BR      10004$
   (4)  020126  002240                            .WORD   FLAG
   (5)  020130  000130                            .WORD   T$CODE
   (5)  020132  006752                            .WORD   READY
   (5)  020134  100000                            .WORD   100000
   (3)  020136                          10004$:
2317    020136                                    IF #BIT15 SETIN @LPCSR(R2) THEN
   (6)  020136  032772  100000  002322            BIT     #BIT15,@LPCSR(R2)
   (10) 020144  001431                            BEQ     50221$
2318    020146                                    PRINTF #DOORDY,LUNIT
   (8)  020146  013746  002260                    MOV     LUNIT,-(SP)
   (7)  020152  012746  021354                    MOV     #DOORDY,-(SP)
   (6)  020156  012746  000002                    MOV     #2,-(SP)
   (3)  020162  010600                            MOV     SP,R0
   (4)  020164  104417                            TRAP    C$PNTF
   (4)  020166  062706  000006                    ADD     #6,SP
2319    020172                                    PRINTF #DOORD1
   (7)  020172  012746  021423                    MOV     #DOORD1,-(SP)
   (6)  020176  012746  000001                    MOV     #1,-(SP)
   (3)  020202  010600                            MOV     SP,R0
   (4)  020204  104417                            TRAP    C$PNTF
   (4)  020206  062706  000004                    ADD     #4,SP
2320    020212                                    GMANIL READY, FLAG, 100000, YES
   (3)  020212  104443                            TRAP    C$GMAN
   (3)  020214  000404                            BR      10005$
   (4)  020216  002240                            .WORD   FLAG
   (5)  020220  000130                            .WORD   T$CODE
   (5)  020222  006752                            .WORD   READY
   (5)  020224  100000                            .WORD   100000
   (3)  020226                          10005$:
2321    020226                                    ELSE
   (4)  020226  000411                            BR      50222$
   (3)  020230                          50221$:
2322    020230                                    LET ERRTBL(R2) := ERRTBL(R2) + #1
   (7)  020230  005262  003060                    INC     ERRTBL(R2)
2323    020234                                    LET L$LUN := LUNIT
   (4)  020234  013737  002260  002074            MOV     LUNIT,L$LUN
2324    020242                                    ERRHRD 10, DOOSWI
   (4)  020242  104456                            TRAP    C$ERHRD
   (5)  020244  000012                            .WORD   10
   (5)  020246  003530                            .WORD   DOOSWI
   (5)  020250  000000                            .WORD   0
```

```
2325  020252                                  ENDIF
 (4)  020252                            50222$:
2326  020252                                  LET @LPCSR(R2) := #00
 (4)  020252  012772  000000  002322         MOV     #00,@LPCSR(R2)
2327  020260                              ENDINC
 (5)  020260  000664                         BR      50217$
 (4)  020262                            50220$:
2328  020262                              LET OUTBUF := #14
 (4)  020262  012737  000014  003124         MOV     #14,OUTBUF
2329  020270                              OUTPUT #OUTBUF,#1
2330  020332  004737  005246               JSR PC,QUIET
2331  020336                              EXIT TST
 (3)  020336  104432                         TRAP    C$EXIT
 (3)  020340  001116                         .WORD   L10017-.
2332
2333                                    .NLIST BEX
2334
2335  020342  042522  042101  020131   INTLK:  .ASCIZ  /READY LINE INTERLOCK TEST 9/  2><12>
2336  020400  047045  040445  042522   PAPRSW: .ASCIZ  /%N%AREMOVE ALL PAPER FROM BOTH PAPER TRAYS%N/
2337  020455     045  053501  052111   PAPSW1: .ASCIZ  /%AWITH EXCEPTION OF ONE PER TRAY ON LUNIT %D2%N/
2338  020535     045  052101  020117   PAPSW2: .ASCIZ  /%ATO CHECK PAPER OUT INTERLOCK.%N/
2339  020577     045  022516  051101   PAPRDY: .ASCIZ  /%N%ARESTORE PAPER, CLEAR, PLACE LUNIT %D2%A ON LINE.%N/
2340  020666  040520  042520  020122   PAPTST: .ASCIZ  /PAPER OUT TEST/<12>
2341  020706  047045  040445  052524   HANRSW: .ASCIZ  /%N%ATURN PAPER TRAY HANDLE COUNTER CLOCKWISE TO%N/
2342  020770  040445  047510  044522   HANSW1: .ASCIZ  /%AHORIZONTAL POSITION UNTIL IT STOPS, ON LUNIT %D2%N/
2343  021055     045  052101  020117   HANSW2: .ASCIZ  /%ATO CHECK INTERLOCK SWITCH.%N/
2344  021114  047045  040445  042522   HANRDY: .ASCIZ  /%N%ARETURN PAPER TRAY HANDLE TO VERTICAL POSITION,%N/
2345  021201     045  041501  042514   HANRD1: .ASCIZ  /%ACLEAR, PLACE LUNIT %D2%A ON LINE.%N/
2346  021247     045  022516  047501   DOORSW: .ASCIZ  /%N%AOPEN FRONT DOOR ON LUNIT %D2%A TO CHECK /
2347  021324  047045  040445  047111   DOOSW1: .ASCIZ  /%N%AINTERLOCK SWITCH.%N/
2348  021354  047045  040445  046103   DOORDY: .ASCIZ  /%N%ACLOSE FRONT DOOR ON LUNIT %D2%A,%N/
2349  021423     045  041501  042514   DOORD1: .ASCIZ  /%ACLEAR, PLACE ON LINE.%N/
2350        021456                      .EVEN
2351
2352                                    .LIST BEX
2353  021456                            ENDTST
 (3)  021456                            L10017:
 (3)  021456  104401                         TRAP    C$ETST
2354
2355  021460                            ENDMOD
2356
2357                                    .SBTTL  INTERRUPT SERVICE ROUTINES
2358  021460                            BGNSRV
2359                                    ;
2360                                    ;++
2361                                    ;INTERRUPT VECTORS ARE ESTABLISHED DURING INITIALIZATION
2362                                    ;POINTING TO THE BASIC ROUTINES WHICH
2363                                    ;SET UP THE UNIT NUMBER CAUSING THE INTERRUPTS.
2364                                    ;LINE NUMBER IS RETURNED IN R2
2365                                    ;
2366                                    ;--
2367        000000                      X=0
2368  021460  000020                    INT00:  .REPT 16.
2369                                            SETPRI #PR104
2370                                            PUSH    R2
2371                                            LET R2 :- #X
```

```
                                                        INLINE <JMP     IODRV>
                                        X=X+2
                                                        .ENDR
 2372
 2373
 2374
  (4)  021460  012700  000200                           MOV     #PRI04,R0
  (4)  021464  104441                                    TRAP    C$SPRI
  (3)  021466  010246                                    MOV     R2,-(SP)
  (5)  021470  012702  000000                           MOV     #X,R2
  (3)  021474  000137  004464                           JMP     IODRV
  (4)  021500  012700  000200                           MOV     #PRI04,R0
  (4)  021504  104441                                    TRAP    C$SPRI
  (3)  021506  010246                                    MOV     R2,-(SP)
  (5)  021510  012702  000002                           MOV     #X,R2
  (3)  021514  000137  004464                           JMP     IODRV
  (4)  021520  012700  000200                           MOV     #PRI04,R0
  (4)  021524  104441                                    TRAP    C$SPRI
  (3)  021526  010246                                    MOV     R2,-(SP)
  (5)  021530  012702  000004                           MOV     #X,R2
  (3)  021534  000137  004464                           JMP     IODRV
  (4)  021540  012700  000200                           MOV     #PRI04,R0
  (4)  021544  104441                                    TRAP    C$SPRI
  (3)  021546  010246                                    MOV     R2,-(SP)
  (5)  021550  012702  000006                           MOV     #X,R2
  (3)  021554  000137  004464                           JMP     IODRV
  (4)  021560  012700  000200                           MOV     #PRI04,R0
  (4)  021564  104441                                    TRAP    C$SPRI
  (3)  021566  010246                                    MOV     R2,-(SP)
  (5)  021570  012702  000010                           MOV     #X,R2
  (3)  021574  000137  004464                           JMP     IODRV
  (4)  021600  012700  000200                           MOV     #PRI04,R0
  (4)  021604  104441                                    TRAP    C$SPRI
  (3)  021606  010246                                    MOV     R2,-(SP)
  (5)  021610  012702  000012                           MOV     #X,R2
  (3)  021614  000137  004464                           JMP     IODRV
  (4)  021620  012700  000200                           MOV     #PRI04,R0
  (4)  021624  104441                                    TRAP    C$SPRI
  (3)  021626  010246                                    MOV     R2,-(SP)
  (5)  021630  012702  000014                           MOV     #X,R2
  (3)  021634  000137  004464                           JMP     IODRV
  (4)  021640  012700  000200                           MOV     #PRI04,R0
  (4)  021644  104441                                    TRAP    C$SPRI
  (3)  021646  010246                                    MOV     R2,-(SP)
  (5)  021650  012702  000016                           MOV     #X,R2
  (3)  021654  000137  004464                           JMP     IODRV
  (4)  021660  012700  000200                           MOV     #PRI04,R0
  (4)  021664  104441                                    TRAP    C$SPRI
  (3)  021666  010246                                    MOV     R2,-(SP)
  (5)  021670  012702  000020                           MOV     #X,R2
  (3)  021674  000137  004464                           JMP     IODRV
  (4)  021700  012700  000200                           MOV     #PRI04,R0
  (4)  021704  104441                                    TRAP    C$SPRI
  (3)  021706  010246                                    MOV     R2,-(SP)
  (5)  021710  012702  000022                           MOV     #X,R2
  (3)  021714  000137  004464                           JMP     IODRV
  (4)  021720  012700  000200                           MOV     #PRI04,R0
  (4)  021724  104441                                    TRAP    C$SPRI
  (3)  021726  010246                                    MOV     R2,-(SP)
```

```
   (5)  021730  012702  000024                MOV     #X,R2
   (3)  021734  000137  004464                JMP     IODRV
   (4)  021740  012700  000200                MOV     #PRI04,R0
   (4)  021744  104441                        TRAP    C$SPRI
   (3)  021746  010246                        MOV     R2,-(SP)
   (5)  021750  012702  000026                MOV     #X,R2
   (3)  021754  000137  004464                JMP     IODRV
   (4)  021760  012700  000200                MOV     #PRI04,R0
   (4)  021764  104441                        TRAP    C$SPRI
   (3)  021766  010246                        MOV     R2,-(SP)
   (5)  021770  012702  000030                MOV     #X,R2
   (3)  021774  000137  004464                JMP     IODRV
   (4)  022000  012700  000200                MOV     #PRI04,R0
   (4)  022004  104441                        TRAP    C$SPRI
   (3)  022006  010246                        MOV     R2,-(SP)
   (5)  022010  012702  000032                MOV     #X,R2
   (3)  022014  000137  004464                JMP     IODRV
   (4)  022020  012700  000200                MOV     #PRI04,R0
   (4)  022024  104441                        TRAP    C$SPRI
   (3)  022026  010246                        MOV     R2,-(SP)
   (5)  022030  012702  000034                MOV     #X,R2
   (3)  022034  000137  004464                JMP     IODRV
   (4)  022040  012700  000200                MOV     #PRI04,R0
   (4)  022044  104441                        TRAP    C$SPRI
   (3)  022046  010246                        MOV     R2,-(SP)
   (5)  022050  012702  000036                MOV     #X,R2
   (3)  022054  000137  004464                JMP     IODRV
  2375
```

```
                                        .SBTTL CLOCK SERVICE ROUTINE
                                        ;++
                                        ;UPDATES THE COUNTER AT A RATE OF 16.67 MILLISECONDS PER TICK
                                        ;AND UPDATES A SECOND COUNTER WHEN THE FIRST OVERFLOWS.
                                        ;--

022060                                  BGNSRV
022060                                  CLKTCK: SETPRI #PRIO6
  (3)  022060  012700  000300                   MOV     #PRIO6,R0
  (3)  022064  104441                           TRAP    C$SPRI
022066                                          IF TICK EQ #0 THEN
  (6)  022066  005737  022134                   TST     TICK
  (10) 022072  001005                           BNE     50223$
022074                                              LET TICK := #60.            ;60 TICKS PER SECOND
  (4)  022074  012737  000074  022134           MOV     #60.,TICK
022102                                              LET TIME := TIME + #1
  (7)  022102  005237  022132                   INC     TIME
022106                                          ENDIF
  (4)  022106                           50223$:
022106                                          LET TICK := TICK - #1           ;BACK UP SECOND TIMER
  (7)  022106  005337  022134                   DEC     TICK
022112                                          IF CLKTYP EQ #2 THEN
  (6)  022112  023727  002266  000002           CMP     CLKTYP,#2
  (10) 022120  001003                           BNE     50224$
022122                                              LET @CLKCSR := #100
  (4)  022122  012777  000100  160142           MOV     #100,@CLKCSR
022130                                          ENDIF
  (4)  022130                           50224$:

022130                                  ENDSRV                              ;AND EXIT
  (3)  022130                           L10021:
  (2)  022130  000002                           RTI

022132  000000                          TIME:   .WORD   0
022134  000000                          TICK:   .WORD   0
                                        .SBTTL  HARDWARE PARAMETER SECTION
022136                                  BGNMOD

                                        ;++
                                        ;THIS SECTION INCLUDES THE QUESTIONS WHICH REQUEST THE OPERATOR TO
                                        ;FURNISH THE HARDWARE INFORMATION NECESSARY TO BUILD THE HARDWARE
                                        ;P-TABLES.
                                        ;
                                        ;--
022136                                  BGNHRD
  (3)  022136  000010                           .WORD L10022-L$HARD/2
  (3)  022140                           L$HARD::

022140                                          GPRMA   GETADR,0,0,160000,177516,YES
  (4)  022140  000031                           .WORD   T$CODE
  (4)  022142  022160                           .WORD   GETADR
  (4)  022144  160000                           .WORD   T$LOLIM
  (4)  022146  177516                           .WORD   T$HILIM
022150                                          GPRMA   GETVEC,2,0,110,770,YES
  (4)  022150  001031                           .WORD   T$CODE
  (4)  022152  022175                           .WORD   GETVEC
```

```
  (4)  022154  000110                        .WORD   T$LOLIM
  (4)  022156  000770                        .WORD   T$HILIM
2411   022160                              ENDHRD
  (2)                                      .EVEN
  (3)  022160                              L10022:
2412                                       .NLIST BEX
2413   022160  050114  030461  040440      GETADR: .ASCIZ  /LP11 ADDRESS/
2414   022175     111  052116  051105      GETVEC: .ASCIZ  /INTERRUPT VECTOR/
2415                                       .LIST BEX
2416                                       .EVEN
2417                                       .SBTTL  SOFTWARE PARAMETER SECTION
2418                                       ;
2419                                       ;++
2420                                       ;THIS SECTION INCLUDES THE QUESTIONS WHICH REQUEST THE OPERATOR TO FURNISH
2421                                       ;THE SOFTWARE INFORMATION NECESSARY TO BUILD THE SOFTWARE P-TABLES.
2422                                       ;--
2423                                       ;
2424   022216                              BGNSFT
  (3)  022216  000010                        .WORD L10023-L$SOFT/2
  (3)  022220                              L$SOFT::
2425   022220                                      GPRML   MGTINT,0,1,YES
  (4)  022220  000130                        .WORD   T$CODE
  (4)  022222  022240                        .WORD   MGTINT
  (4)  022224  000001                        .WORD   1
2426   022226                                      GPRMD   GETMAX,2,D,377,1,255.,YES
  (4)  022226  001052                        .WORD   T$CODE
  (4)  022230  022276                        .WORD   GETMAX
  (4)  022232  000377                        .WORD   377
  (4)  022234  000001                        .WORD   T$LOLIM
  (4)  022236  000377                        .WORD   T$HILIM
2427   022240                              ENDSFT
  (2)                                      .EVEN
  (3)  022240                              L10023:
2428                                       .NLIST BEX
2429   022240  052522  020116  040515      MGTINT: .ASCIZ  /RUN MANUAL INTERVENTION TESTS/
2430   022276  052501  047524  051104      GETMAX: .ASCIZ  /AUTODROP ERROR COUNT/
2431                                       .LIST BEX
2432           022324                      .EVEN
2433                                       ;
2434
2435   022324  000020                      PATCH:  .BLKW   20
2436   022364                              LASTAD
  (2)                                      .EVEN
  (4)  022364  000000                        .WORD   0
  (4)  022366  000000                        .WORD   0
  (3)  022370                              L$LAST::
2437   022370                              ENDMOD
2438           000001                      .END
```

```
ACTIVE= 020000          855#  1135  1147  1152  1220  1226  1232  1510  1614  2286
ADR   = 000020 G        816#
ASSEMB= 000010          659   661
BIN2DA  004252         1069#
BIT0  = 000001 G        816#
BIT00 = 000001 G        816#
BIT01 = 000002 G        816#
BIT02 = 000004 G        816#
BIT03 = 000010 G        816#
BIT04 = 000020 G        816#
BIT05 = 000040 G        816#
BIT06 = 000100 G        816#
BIT07 = 000200 G        816#  1574  1584  2228
BIT08 = 000400 G        816#
BIT09 = 001000 G        816#
BIT1  = 000002 G        816#
BIT10 = 002000 G        816#
BIT11 = 004000 G        816#
BIT12 = 010000 G        816#
BIT13 = 020000 G        816#   855
BIT14 = 040000 G        816#   854
BIT15 = 100000 G        816#   853  1125  1212  1557  2218  2300  2317
BIT2  = 000004 G        816#
BIT3  = 000010 G        816#
BIT4  = 000020 G        816#
BIT5  = 000040 G        816#
BIT6  = 000100 G        816#
BIT7  = 000200 G        816#  1566  1630
BIT8  = 000400 G        816#
BIT9  = 001000 G        816#
BOE   = 000400 G        816#
BUFADD  002314          899#  1249  1250  1281* 1344* 1439* 1623* 1628* 1689* 1709* 1714* 1742* 1753*
                       1756* 1758* 1801* 1832* 1873* 1978* 1983* 1986* 1990* 1992* 1995* 1997* 1998*
                       2005* 2055* 2076* 2081* 2148* 2149* 2150* 2151* 2152* 2153* 2154* 2155* 2156*
                       2157* 2158* 2159* 2160* 2161* 2162* 2165* 2166* 2167* 2169* 2171* 2175* 2236*
                       2259* 2263* 2268* 2329*
BUFCNT  002316          901#  1251  1252  1281* 1344* 1439* 1623* 1628* 1689* 1709* 1714* 1742* 1753*
                       1756* 1758* 1801* 1832* 1873* 1978* 1983* 1986* 1990* 1992* 1995* 1997* 1998*
                       2005* 2055* 2076* 2081* 2148* 2149* 2150* 2151* 2152* 2153* 2154* 2155* 2156*
                       2157* 2158* 2159* 2160* 2161* 2162* 2165* 2166* 2167* 2169* 2171* 2175* 2236*
                       2259* 2263* 2268* 2329*
BUFREP  002320          903#  1253  1281* 1344* 1439* 1623* 1628* 1689* 1709* 1714* 1742* 1753* 1756*
                       1758* 1801* 1832* 1873* 1978* 1983* 1986* 1990* 1992* 1995* 1997* 1998* 2005*
                       2055* 2076* 2081* 2148* 2149* 2150* 2151* 2152* 2153* 2154* 2155* 2156* 2157*
                       2158* 2159* 2160* 2161* 2162* 2165* 2166* 2167* 2169* 2171* 2175* 2236* 2259*
                       2263* 2268* 2329*
CCNT    002250          868#  1698* 2063*
CHAR    011312         1693* 1695* 1699  1700* 1701* 1702  1703* 1704* 1718#
CHRGEN  002254          870#
CK1     006306         1423#
CLEAN   007242         1504#
CLKCSR  002272          883#  1416* 1417* 1427* 1430* 1519* 2391*
CLKENA  002300          886#
CLKSET  002274          884#
CLKTCK  022060         2384#
CLKTYP  002266          878#  1411* 1414* 1425* 1433  1517  1521  2390
```

```
CLKVEC  002276            885#   1419*  1420   1428*  1518   1522
CLOCKP  002270            882#   1415*  1416   1426*  1427
COUNT   002246            867#   1749*  1807*  1979*  1999*  2000   2164*  2168
COUNTD  004460           1073*   1078*  1081   1082*  1083   1090*  1105#
CR    = 000015 G          823#   2141
CSRERR  003352            971#   1561   2220
CTLEND  005240           1198    1269#
CTLLOP  004722           1196#   1268
CURADD  002516            921#   1130   1131*  1142*  1144   1145*  1249*  2288*
CURCNT  002716            933#   1129   1136*  1143*  1146*  1219   1251*  1254   1295*  1378*  1511*  2287*
C$AU  = 000052            659#
C$AUTO= 000061            659#    731
C$BRK = 000022            659#
C$BSEG= 000004            659#
C$BSUB= 000002            659#
C$CEFG= 000045            659#
C$CLCK= 000062            659#   1412   1423
C$CLEA= 000012            659#   1525
C$CLOS= 000035            659#
C$CLP1= 000006            659#
C$CVEC= 000036            659#   1518
C$DCLN= 000044            659#   1302
C$DODU= 000051            659#
C$DRP1= 000024            659#
C$DU  = 000053            659#
C$EDIT= 000003            659#    680
C$ERDF= 000055            659#
C$ERHR= 000056            659#   1035   1040   1046   1561   1578   1588   1612   1641   2220   2230   2269   2307
                         2324
C$ERRO= 000060            659#
C$ERSF= 000054            659#
C$ERSO= 000057            659#
C$ESCA= 000010            659#
C$ESEG= 000005            659#
C$ESUB= 000003            659#
C$ETST= 000001            659#   1661   1727   1769   1948   2033   2097   2188   2353
C$EXIT= 000032            659#   1348   1440   1632   1716   1761   1875   2009   2083   2177   2207   2210   2331
C$GETB= 000026            659#
C$GETW= 000027            659#
C$GMAN= 000043            659#   1368   2255   2277   2299   2303   2316   2320
C$GPHR= 000042            659#   1317   1374
C$GPLO= 000030            659#
C$GPRI= 000040            659#
C$INIT= 000011            659#   1455
C$INLP= 000020            659#
C$MANI= 000050            659#   1357   2205
C$MEM = 000031            659#
C$MSG = 000023            659#
C$OPEN= 000034            659#
C$PNTB= 000014            659#
C$PNTF= 000017            659#   1297   1301   1354   1355   1360   1434   1435   2252   2253   2254   2275   2296
                         2297    2298   2301   2302   2314   2315   2318   2319
C$PNTS= 000016            659#
C$PNTX= 000015            659#
C$QIO = 000377            659#
C$RDBU= 000007            659#
```

```
C$REFG= 000047            659#   1336   1338
C$RESE= 000033            659#   1352   1502
C$REVI= 000003            659#    680
C$RFLA= 000021            659#
C$RPT = 000025            659#
C$SEFG= 000046            659#
C$SPRI= 000041            659#   1342   1437   1501   1524   1595   1606   1622   2374   2384
C$SVEC= 000037            659#   1396   1420   1480   1522   1596   1607
C$TPRI= 000013            659#
DATPTH  011314           1689    1719#
DEL   = 000177 G           824#
DELCNT  003016             939#   1227*  1229*  1230   1262*  1379*  1513*  1615*
DFPTBL  002222 G           716#
DIAGMC= 000000             659     661
DIGITS  004462            1075*   1106#
DONE    011747            1756    1764#
DOORDY  021354            2318    2348#
DOORD1  021423            2319    2349#
DOORSW  021247            2314    2346#
DOOSWI  003530             975#   2324
DOOSW1  021324            2315    2347#
DROPED= 040000             854#   1208   1294   1405   1510   1614
DROPIT  005314            1053    1294#
DVSTR   016430            2148    2182#
EF.CON= 000036 G           816#
EF.NEW= 000035 G           816#
EF.PWR= 000034 G           816#
EF.RES= 000037 G           816#   1338
EF.STA= 000040 G           816#   1336
END2    010174            1613#   1642   1649
ERFLG = 000400            1030#
ERRCOD  002302             887#   1030   1056*  1160*  1204*  1213*  1224*  1231*  1239   1365*  2285*
ERRFLG  002304             889#   2240*  2257*  2265   2266   2273*  2284*
ERROR = 100000             853#   1055   1159   1214   1233   1510   1558   1575   1585   1614   2286
ERRSVC  003056             942#   1161   1243   1281*  1344*  1439*  1623*  1628*  1689*  1709*  1714*  1742*  1753*
                          1756*  1758*  1801*  1832*  1873*  1978*  1983*  1986*  1990*  1992*  1995*  1997*  1998*
                          2005*  2055*  2076*  2081*  2148*  2149*  2150*  2151*  2152*  2153*  2154*  2155*  2156*
                          2157*  2158*  2159*  2160*  2161*  2162*  2165*  2166*  2167*  2169*  2171*  2175*  2236*
                          2259*  2263*  2268*  2329*
ERRTBL  003060 G           943#   1033*  1038*  1044*  1052   1298*  1377*  1512*  1559*  1577*  1586*  1610*  1639*
                          2219*  2229*  2270*  2305*  2322*
ERR11   010504            1588    1657#
ERR12   010560            1641    1658#
ERR13   010643            1612    1659#
EVL   = 000004 G           816#
E$END = 002100             659#
E$LOAD= 000035             659#    680
FAKE    005422            1315#   1341
FF    = 000014 G           822#   2143
FLAG    002240             864#   1364*  1368   2212*  2251*  2255   2276*  2277   2295*  2299   2303   2313*  2316
                          2320
FLAGDA  004456            1072*   1081   1084*  1093   1104#
F$AU  = 000015             659#
F$AUTO= 000020             659#    727    731
F$BGN = 000040             659#    677    685    727    802    806   1323   1332   1333   1348   1440   1500   1527
                          1530    1541   1632   1638   1647   1661   1663   1668   1683   1716   1727   1729   1732
```

```
                    1738    1761    1769    1771    1775    1798    1875    1948    1950    1954    1970    2009    2033
                    2034    2038    2048    2083    2097    2098    2102    2135    2177    2188    2190    2193    2203
                    2207    2210    2331    2353    2355    2358    2383    2399    2407    2424    2437
F$CLEA= 000007       659#    1500    1525
F$DU  = 000016       659#
F$END = 000041       659#     677     731     802     806    1323    1332    1348    1440    1455    1525    1527    1530
                    1541    1632    1643    1650    1661    1663    1668    1683    1716    1727    1729    1732    1738
                    1761    1769    1771    1775    1798    1875    1948    1950    1954    1970    2009    2033    2034
                    2038    2048    2083    2097    2098    2102    2135    2177    2188    2190    2193    2203    2207
                    2210    2331    2353    2355    2394    2399    2411    2427    2437
F$HARD= 000004       659#    2407    2411
F$HW  = 000013       659#     716     723
F$INIT= 000006       659#    1333    1455
F$JMP = 000050       659#    1348    1440    1632    1716    1761    1875    2009    2083    2177    2207    2210    2331
F$MOD = 000000       659#     677     802     806    1323    1332    1527    1530    1663    1668    1729    1732    1771
                    1775    1950    1954    2034    2038    2098    2102    2190    2193    2355    2399    2437
F$MSG = 000011       659#
F$PROT= 000021       659#     685     689
F$PWR = 000017       659#
F$RPT = 000012       659#
F$SEG = 000003       659#
F$SOFT= 000005       659#    2424    2427
F$SRV = 000010       659#    1638    1643    1647    1650    2358    2383    2394
F$SUB = 000002       659#
F$SW  = 000014       659#     740     749
F$TEST= 000001       659#    1541    1661    1683    1727    1738    1769    1798    1948    1970    2033    2048    2097
                    2135    2188    2203    2353
GETADR  022160      2409    2413#
GETMAX  022276      2426    2430#
GETVEC  022175      2410    2414#
G$CNTO= 000200       659#
G$DELM= 000372       659#    1228    1346    1568    1599    1609
G$DISP= 000003       659#
G$EXCP= 000400       659#
G$HILI= 000002       659#
G$LOLI= 000001       659#
G$NO  = 000000       659#
G$OFFS= 000400       659#    1368    2255    2277    2299    2303    2316    2320    2409    2410    2425    2426
G$OFSI= 000376       659#    1368    2255    2277    2299    2303    2316    2320    2409    2410    2425    2426
G$PRMA= 000001       659#    2409    2410
G$PRMD= 000002       659#    2426
G$PRML= 000000       659#    1368    2255    2277    2299    2303    2316    2320    2425
G$RADA= 000140       659#
G$RADB= 000000       659#
G$RADD= 000040       659#    2426
G$RADL= 000120       659#    1368    2255    2277    2299    2303    2316    2320    2425
G$RADO= 000020       659#    2409    2410
G$XFER= 000004       659#
G$YES = 000010       659#    1368    2255    2277    2299    2303    2316    2320    2409    2410    2425    2426
HANRDY  021114      2301    2344#
HANRD1  021201      2302    2345#
HANRSW  020706      2296    2341#
HANSW1  003455       974#    2307
HANSW1  020770      2297    2342#
HANSW2  021055      2298    2343#
HOE   = 100000 G     816#
```

```
IBE    = 010000 G          816#
IDU    = 000040 G          816#
IER    = 020000 G          816#
IGNORE  007146             1420    1463#   1522
INDEX   002310              894#
INHINT  002234              742#   2209
INTERR  010366             1596    1639#
INTER1  003574              976#
INTFAC  010424             1623    1653#
INTHDL  010416             1607    1649#
INTLK   020342             2236    2335#
INTOO   021460             1394    2368#
IOCTRL  004662             1181#   1281    1344    1439    1623    1628    1689    1709    1714    1742    1753    1756    1758
                           1801    1832    1873    1978    1983    1986    1990    1992    1995    1997    1998    2005    2055
                           2076    2081    2148    2149    2150    2151    2152    2153    2154    2155    2156    2157    2158
                           2159    2160    2161    2162    2165    2166    2167    2169    2171    2175    2236    2259    2263
                           2268    2329
IODRV   004464             1125#   2374
ISR    = 000100 G           816#
IXE    = 004000 G           816#
I$AU   = 000041             659#
I$AUTO= 000041              659#    727#    731#
I$CLN  = 000041             659#   1500#   1525#
I$DU   = 000041             659#
I$HRD  = 000041            2407#   2411#
I$INIT= 000041              659#   1333#   1348    1440    1455#
I$MOD  = 000041             659#    677#    802#    806#   1323#   1332#   1527#   1530#   1663#   1668#   1729#   1732#   1771#
                           1775#   1950#   1954#   2034#   2038#   2098#   2102#   2190#   2193#   2355#   2399#   2437#
I$MSG  = 000041             659#
I$PROT= 000040              659#    685#
I$PTAB= 000041              659#
I$PWR  = 000041             659#
I$RPT  = 000041             659#
I$SEG  = 000041             659#   15 1    1683    1738    1798    1970    2048    2135    2203
I$SETU= 000041              659#
I$SFT  = 000041            2424#   2427#
I$SRV  = 000041             659#   1638#   1643#   1647#   1650#   2358#   2383#   2394#
I$SUB  = 000041             659#   1541    1683    1738    1798    1970    2048    2135    2203
I$TST  = 000041             659#   1541#   1632    1661#   1683#   1716    1727#   1738#   1761    1769#   1798#   1875    1948#
                           1970#   2009    2033#   2048#   2083    2097#   2135#   2177    2186#   2203#   2207    2210    2331
                           2353#
J$JMP  = 000167             659#
LF     = 000012 G           821#   1752    2142
LINCNT  002242              865#   1708*   1808*   1994*   2060*   2061*   2067    2074    2079    2256*   2260*   2261    2265
LOBYTE= 000377              856#   1510
LOE    = 040000 G           816#
LOT    = 000010 G           816#
LPBUF   002416              915#   1130*   1144*   1385*   1583*
LPCSR   002322              909#   1057*   1125    1137*   1148*   1153*   1212    1255*   1296*   1384*   1385    1557    1562*
                           1566    1574    1584    1589*   1597*   1605*   1608*   1613*   1630    2218    2221*   2228    2279*
                           2300    2317    2326*
LPDROP  004002              992#   1297
LPERR   004050             1030#   1281    1344    1439    1623    1628    1689    1709    1714    1742    1753    1756    1758
                           1801    1832    1873    1978    1983    1986    1990    1992    1995    1997    1998    2005    2055
                           2076    2081    2148    2149    2150    2151    2152    2153    2154    2155    2156    2157    2158
                           2159    2160    2161    2162    2165    2166    2167    2169    2171    2175    2236    2329
```

```
LPINTR  002756          936#  1395*  1396  1480
LPVEC   002362          912#  1389*  1396  1480  1596  1607
LSTCNT  002244          866#
LUNIT   002260          872#  1547*  1550*  1552  1556  1560  1576  1587  1611  1640  1976*  1977  1978
                       1983   1986   1990  1992  1995  1997  1998  2005  2216*  2217  2225*  2226  2227
                       2241*  2244*  2245  2247  2253  2259  2263  2268  2275  2278  2292*  2293  2294
                       2297   2302   2306  2311*  2312  2314  2318  2323
L$ACP   002110 G        680#
L$APT   002036 G        680#
L$AUT   002070 G        680#
L$AUTO  002226 G        680   727#
L$CCP   002106 G        680#
L$CLEA  007232 G        680   1500#
L$CO    002032 G        680#
L$DEPO  002011 G        680#
L$DESC  002152 G        680   702#
L$DESP  002076 G        680#
L$DEVP  002060 G        680#
L$DISP  002132 G        680   698#
L$DLY   002116 G        680#  1228  1346  1568  1599  1609
L$DTP   002040 G        680#
L$DTYP  002034 G        680#
L$DUT   002072 G        680#
L$DVTY  002212 G        680   703#
L$EF    002052 G        680#
L$ENVI  002044 G        680#
L$ETP   002102 G        680#
L$EXP1  002046 G        680#
L$EXP4  002064 G        680#
L$EXP5  002066 G        680#
L$HARD  022140 G        680   2407#
L$HIME  002120 G        680#
L$HPCP  002016 G        680#
L$HPTP  002022 G        680#
L$HW    002222 G        680   716#
L$ICP   002104 G        680#
L$INIT  005456 G        680   333#
L$LADP  002026 G        680#
L$LAST  022370 G        680   2436#
L$LOAD  002100 G        680#
L$LUN   002074 G        680#  1034*  1039*  1045*  1188*  1191*  1203  1267*  1297  1315*  1316  1317  1318*
                       1373*  1374   1376   1505*  1508   1560*  1576*  1587*  1611*  1640*  2227*  2294*  2306*
                       2323*
L$MREV  002050 G        680#
L$NAME  002000 G        680#
L$PRIO  002042 G        680#
L$PROT  002122 G        680   685#
L$PRT   002112 G        680#
L$REPP  002062 G        680#
L$REV   002010 G        680#
L$SOFT  022220 G        680   2424#
L$SPC   002056 G        680#
L$SPCP  002020 G        680#
L$SPTP  002024 G        680#
L$STA   002030 G        680#
L$SW    002234 G        680   740#
```

```
L$TEST  002114 G          680#
L$TIML  002014 G          680#
L$UNIT  002012 G          680#   1187    1316    1353    1372    1478    1504    1542    1974    2213
L10001  002226            716     723#
L10002  002230            731#
L10003  002240            740     749#
L10004  007144           1348    1440    1455#
L10005  007424           1525#
L10006  010720           1632    1661#
L10007  010414           1643#
L10010  010422           1650#
L10011  011354           1716    1727#
L10012  012016           1761    1769#
L10013  013046           1875    1948#
L10014  014300           2009    2033#
L10015  014766           2083    2097#
L10016  016520           2177    2188#
L10017  021456           2207    2210    2331    2353#
L10021  022130           2394#
L10022  022160           2407    2411#
L10023  022240           2424    2427#
MAXERR  002236            746#   1052
MGTINT  022240           2425    2429#
MRESET  006625           1445#
MSGADR  002656            930#   1142    1250*
MSGCNT  002556            924#   1143    1252*
MULINE  014637           2055    2089#
NEWSPA  016426           2181#
NOCLCK  007007           1434    1449#
NOINTR= 000003            832#   1224
NONBUF  012461           1802    1883#
NONCHR  012352           1801    1880#
NOTIM   007051           1435    1450#
NRGT16  006506           1354    1443#
NRGT17  006571           1355    1444#
NUM     013044           1945#
ONEFIL= 000001              4#      8     599
OUTBUF  003124            958#   1343*   1344    1438*   1439    1627*   1628    1697    1709    1713*   1714    1748    1753
                         1757*   1758    1809    1832    1872*   1873    1985*   1986    1992    1997    1998    2004*   2005
                         2062    2076    2080*   2081    2137    2149    2150    2152    2153    2155    2156    2158    2159
                         2161    2162    2165    2166    2167    2169    2171    2175    2262*   2263    2267*   2268    2328*
                         2329
OUTTIM  003654            978#   1040
O$APTS= 000000            659#    680
O$AU  = 000000            659#    680
O$BGNR= 000000            659#    680
O$BGNS= 000001            659#    678#    680
O$DU  = 000000            659#    680
O$ERRT= 000000            659#    680
O$GNSW= 000001            659#    678#    680
O$POIN= 000001            659#    678#    680
O$SETU= 000000            659#    680    2436
PAPRDY  020577           2275    2339#
PAPRSW  020400           2252    2336#
PAPSWI  003412            973#   2269
PAPSW1  020455           2253    2337#
```

```
PAPSW2  020535            2254    2338#
PAPTST  020666            2259    2340#
PATCH   022324            2435#
PATTER  011352            1691*   1692    1722#
PLOC    007142            1452#
PNT   = 001000 G           816#
PORSQ   011765            1765#
PRI   = 002000 G           816#
PRINTR  002264             876#   1186    1191    1281*   1344*   1439*   1623*   1628*   1689*   1709*   1714*   1742*   1753*
                          1756*   1758*   1801*   1832*   1873*   1978*   1983*   1986*   1990*   1992*   1995*   1997*   1998*
                          2005*   2055*   2076*   2081*   2148*   2149*   2150*   2151*   2152*   2157*   2154*   2155*   2156*
                          2157*   2158*   2159*   2160*   2161*   2162*   2165*   2166*   2167*   2169*   2171*   2175*   2236*
                          2259*   2263*   2268*   2329*
PRI00 = 000000 G           816#   1342    1437    1524    1622
PRI01 = 000040 G           816#
PRI02 = 000100 G           816#
PRI03 = 000140 G           816#   1606
PRI04 = 000200 G           816#   1396    1480    1595    1596    1607    2374
PRI05 = 000240 G           816#
PRI06 = 000300 G           816#   1420    1522    2384
PRI07 = 000340 G           816#   1501
PRICHR  011710            1742    1763#
PRICTL  013740            1978    2011#
PTABAD  002262             875#
QUIET   005246            1281#   1629    1710    1715    1754    1759    1833    1874    2003    2006    2077    2082    2163
                          2173    2176    2330
RDYERR  003370             972#   1578    2230
READY   006752            1368    1448#   2255    2277    2299    2303    2316    2320
REPCNT  002616             927#   1140*   1141    1253*   1380*   1514*   2289*
RESET1  006701            1360    1446#
RESVEC  007150 G          1476#   1516    1621
SFPTBL  002234 G           740#
SPANUM  016424            2180#
STACHR  014620            2057*   2060    2064    2075*   2086#
STATER= 000001             828#   1160    1213
STATUS  002456             918#   1055*   1135*   1147*   1152*   1159*   1208    1214*   1220    1226    1232*   1233*   1294*
                          1405*   1510*   1558*   1575*   1585*   1614*   2286*
STRCNT  002252             869#
SVCGBL= 000000             659#    669#    680     685     698     702     703     716     727     740     1333    1500    2407
                          2424    2436#
SVCINS= 000000             659#    666#    680     698     702     703     716     731     740     1035    1040    1046    1228
                          1297    1301    1302    1317    1336    1337    1338    1339    1342    1346    1348    1352    1354
                          1355    1357    1358    1360    1368    1374    1375    1396    1412    1420    1423    1434    1435
                          1437    1440    1455    1480    1501    1502    1518    1522    1524    1525    1561    1568    1578
                          1588    1595    1596    1599    1606    1607    1609    1612    1622    1632    1641    1643    1650
                          1661    1716    1727    1761    1769    1875    1948    2009    2033    2083    2097    2177    2188
                          2205    2206    2207    2210    2220    2230    2252    2253    2254    2255    2269    2275    2277
                          2296    2297    2298    2299    2301    2302    2303    2307    2314    2315    2316    2318    2319
                          2320    2324    2331    2353    2374    2384    2394    2407    2409    2410    2411    2424    2425
                          2426    2427    2436
SVCSUB= 000000             659#    668#
SVCTAG= 000000             659#    670#    723     731     749    1368    1455    1489    1499    1525    1643    1650    1661
                          1727    1769    1948    2033    2097    2188    2255    2277    2299    2303    2316    2320    2353
                          2394    2411    2427
SVCTST= 000000             659#    667#    1541    1683    1738    1798    1970    2048    2135    2203
SVM     000037            1030#
```

```
SYMD  = 000007   1033#  1034#  1038#  1039#  1044#  1045#  1055#  1075#  1077#  1078#  1082#  1084#  1089#
                 1099#  1131#  1135#  1136#  1137#  1140#  1145#  1146#  1147#  1152#  1159#  1203#  1214#
                 1229#  1232#  1233#  1266#  1267#  1294#  1299#  1318#  1345#  1372#  1373#  1376#  1385#
                 1393#  1394#  1400#  1405#  1481#  1482#  1504#  1505#  1508#  1510#  1542#  1556#  1558#
                 1559#  1567#  1575#  1577#  1585#  1586#  1597#  1598#  1605#  1608#  1610#  1614#  1639#
                 1691#  1698#  1701#  1704#  1708#  1808#  1813#  1820#  1834#  1974#  1977#  1999#  2061#
                 2063#  2069#  2074#  2075#  2164#  2213#  2216#  2217#  2219#  2225#  2226#  2229#  2245#
                 2260#  2270#  2278#  2286#  2292#  2293#  2305#  2311#  2312#  2322#  2387#  2389#
SYMS  = 000007   1033#  1034#  1038#  1039#  1044#  1045#  1055#  1075#  1077#  1078#  1082#  1084#  1089#
                 1099#  1131#  1135#  1136#  1137#  1140#  1145#  1146#  1147#  1152#  1159#  1203#  1214#
                 1229#  1232#  1233#  1266#  1267#  1294#  1299#  1318#  1345#  1372#  1373#  1376#  1385#
                 1393#  1394#  1400#  1405#  1481#  1482#  1504#  1505#  1508#  1510#  1542#  1556#  1558#
                 1559#  1567#  1575#  1577#  1585#  1586#  1597#  1598#  1605#  1608#  1610#  1614#  1639#
                 1691#  1698#  1701#  1704#  1708#  1808#  1813#  1820#  1834#  1974#  1977#  1999#  2061#
                 2063#  2069#  2074#  2075#  2164#  2213#  2216#  2217#  2219#  2225#  2226#  2229#  2245#
                 2260#  2270#  2278#  2286#  2292#  2293#  2305#  2311#  2312#  2322#  2387#  2389#
S$LSYM- 010000    659#   723#   731#   749#  1368#  1455#  1525#  1643#  1650#  1661#  1727#  1769#  1948#
                 2033#  2097#  2188#  2255#  2277#  2299#  2303#  2316#  2320#  2353#  2394#  2411#  2427#
TABLDA  004444   1071   1103#
TABLE1  014210   1981   2028#
TABLE2  014244   1988   2029#
TABSTR  014622   2057   2088#
TICK    022134   2385   2386*  2389*  2397#
TIME    022132   2387*  2396#
TIMOUT= 000002    829#  1231
TXERR   003627    977#  1035
TXNOIN  003701    979#  1046
T$ARGC= 000001    680#  1297#  1301#  1354#  1355#  1360#  1434#  1435#  2252#  2253#  2254#  2275#  2296#
                 2297#  2298#  2301#  2302#  2314#  2315#  2318#  2319#
T$CODE= 001052   1368#  2255#  2277#  2299#  2303#  2316#  2320#  2409#  2410#  2425#  2426#
T$ERRN= 000012    659#  1035#  1040#  1046#  1561#  1578#  1588#  1612#  1641#  2220#  2230#  2269#  2307#
                 2324#
T$EXCP= 000000   2409#  2410#  2426#
T$FLAG= 000040   1348#  1440#  1632#  1716#  1761#  1875#  2009#  2083#  2177#  2207#  2210#  2331#
T$GMAN= 000000    659#
T$HILI= 000377   2409#  2410#  2426#
T$LAST= 000001    659#  2436#
T$LOLI= 000001   2409#  2410#  2426#
T$LSYM= 010000    659#   723    731    749   1455   1525   1643   1650   1661   1727   1769   1948   2033
                 2097   2188   2353   2394   2411   2427
T$LTNO= 000010   2436#
T$NEST= 000000    659#   677#   685#   689#   716#   723#   727#   731#   740#   749#   802#   806#  1323#
                 1332#  1333#  1455#  1500#  1525#  1527#  1530#  1541#  1638#  1643#  1647#  1650#  1661#
                 1663#  1668#  1683#  1727#  1729#  1732#  1738#  1769#  1771#  1775#  1798#  1948#  1950#
                 1954#  1970#  2033#  2034#  2038#  2048#  2097#  2098#  2102#  2135#  2188#  2190#  2193#
                 2203#  2353#  2355#  2358#  2383#  2394#  2399#  2407#  2411#  2424#  2427#  2437#
T$NS0 = 000010    677#   802    806#  1323   1332#  1527   1530#  1663   1668#  1729   1732#  1771   1775#
                 1950   1954#  2034   2038#  2098   2102#  2190   2193#  2355   2358#
T$NS1 = 000000    685#   689    716#   723    727#   731    740#   749   1333#  1455   1500#  1525   1541#
                 1661   1683#  1727   1738#  1769   1798#  1948   1970#  2033   2048#  2097   2135#  2188
                 2203#  2353   2383#  2394   2399#  2437
T$NS2 = 000005   1638#  1643   1647#  1650   2407#  2411   2424#  2427
T$PTNU= 000000    659#
T$SAVL= 177777    659#
T$SEGL= 177777    659#
T$SUBN= 000000    659#  1541#  1683#  1738#  1798#  1970#  2048#  2135#  2203#
```

```
T$TAGL= 177777          659#
T$TAGN= 010024          659#    685#    716#    727#    740#   1333#   1500#   1541#   1638#   1647#   1683#   1738#   1798#
                       1970#   2048#   2135#   2203#   2358#   2383#   2407#   2424#
T$TEMP= 000000          689#    698#    723#    731#    749#    802#   1323#   1348#   1368#   1440#   1455#   1489#   1499#
                       1525#   1527#   1632#   1643#   1650#   1661#   1663#   1716#   1727#   1729#   1761#   1769#   1771#
                       1875#   1948#   1950#   2009#   2033#   2034#   2083#   2097#   2098#   2177#   2188#   2190#   2207#
                       2210#   2255#   2277#   2299#   2303#   2316#   2320#   2331#   2353#   2355#   2394#   2409#   2410#
                       2411#   2425#   2426#   2427#   2437#
T$TEST= 000010          659#   1541#   1683#   1738#   1798#   1970#   2048#   2135#   2203#   2436
T$TSTM= 177777          659#    731    1035    1040    1046    1297    1301    1302    1317    1336    1338    1342    1348
                       1352    1354    1355    1357    1360    1368    1374    1396    1412    1420    1423    1434    1435
                       1437    1440    1455    1480    1501    1502    1518    1522    1524    1525    1561    1578    1588
                       1595    1596    1606    1607    1612    1622    1632    1641    1661    1716    1727    1761    1769
                       1875    1948    2009    2033    2083    2097    2177    2188    2205    2207    2210    2220    2230
                       2252    2253    2254    2255    2269    2275    2277    2296    2297    2298    2299    2301    2302
                       2303    2307    2314    2315    2316    2318    2319    2320    2324    2331    2353    2374    2384
T$TSTS= 000001          659#   1541#   1683#   1738#   1798#   1970#   2048#   2135#   2203#
T$$AUT= 010002          727#    731
T$$.LE= 010005         1500#   1525
T$$HAR= 010022         2407#   2411
T$$HW = 010001          716#    723
T$$INI= 010004         1333#   1348    1440    1455
T$$PRO= 010000          685#
T$$SOF= 010023         2424#   2427
T$$SRV= 010021         1638#   1643    1647#   1650    2358#   2383#   2394
T$$SW = 010003          740#    749
T$$TES= 010017         1541#   1632    1661    1683#   1716    1727    1738#   1761    1769    1798#   1875    1948    1970#
                       2009    2033    2048#   2083    2097    2135#   2177    2188    2203#   2207    2210    2331    2353
T1      007426 G        698    1541#
T1A     007442         1549#   1620
T1B     010220         1554    1621#
T1C     007446         1548    1551#
T2      010722 G        698    1683#
T3      011356 G        698    1738#
T4      012020 G        698    1798#
T5      013050 G        698    1970#
T6      014302 G        698    2048#
T7      014770 G        698    2135#
T8      016522 G        698    2203#
UAM   = 000200 G        816#
UNIT    002256          871#
UUT     002306          890#   1299*   1300    1366*   1400*
UUTEQO  003732          980#   1301
VERS.1= ****** U       1624    1685    1739    1836    1910    1971    2052    2145
VFUCMD  002312          895#
WORK    003120 G        947#   1393*   1394*   1395    1747*   1750    1813*   1820*
WORK1   003122          948#   1345*   1567*   1598*   1803*   1808
X     = 000040         2367#   2374#
X$ALWA= 000000          659#
X$FALS= 000040          659#
X$OFFS= 000400          659#
X$TRUE= 000020          659#
X0      014031         2014#   2028    2029
X1      014043         2015#   2028    2029
X11     014175         1995    2025#
X2      014055         2016#   2029
```

```
X3      014067          2017#   2029
X4      014101          2018#   2029
X5      014113          2019#   2029
X6      014125          2020#   2029
X7      014137          2021#   2029
X8      014151          2022#   2029
X9      014163          2023#   2029
ZNUM    016422          2179#
ZN1     016450          2151    2183#
ZN2     016453          2154    2184#
ZN3     016462          2157    2185#
ZN4     016476          2160    2186#
$BGNLE= 177777           661#
$BRJMP= 177777           661#   1030   1037   1042   1052   1075   1076   1079   1081   1085   1091   1093   1125
                        1129   1138   1141   1149   1156   1186   1189   1197   1208   1212   1215   1219   1220
                        1225   1226   1230   1235   1239   1244   1254   1300   1316   1319   1345   1347   1353
                        1373   1406   1413   1424   1433   1479   1483   1505   1515   1517   1521   1557   1566
                        1567   1569   1574   1584   1598   1600   1630   1631   1691   1692   1694   1698   1705
                        1708   1711   1712   1746#  1747   1749   1751   1755   1760#  1808   1813   1815   1820
                        1822   1835   1975#  1976   1982   1984   1989   1991   1994   1996   2000   2007   2008#
                        2063   2065   2067   2070   2079   2164   2168   2170   2174   2209   2216   2218   2223
                        2225   2228   2232   2261   2265   2266   2272   2292   2300   2304   2309   2311   2317
                        2321   2327   2385   2390
$ERFLG- 000400           661#   1033#  1034#  1038#  1039#  1044#  1045#  1055#  1056#  1057#  1070#  1071#  1072#
                        1073#  1075#  1077#  1078#  1082#  1083#  1084#  1086#  1089#  1090#  1094#  1097#  1099#
                        1130#  1131#  1135#  1136#  1137#  1140#  1142#  1143#  1144#  1145#  1146#  1147#  1148#
                        1152#  1153#  1159#  1160#  1187#  1188#  1190#  1191#  1203#  1204#  1213#  1214#  1224#
                        1227#  1229#  1231#  1232#  1233#  1249#  1250#  1251#  1252#  1253#  1255#  1262#  1266#
                        1267#  1294#  1295#  1296#  1298#  1299#  1315#  1318#  1343#  1345#  1364#  1365#  1366#
                        1372#  1373#  1376#  1377#  1378#  1379#  1380#  1384#  1385#  1389#  1393#  1394#  1395#
                        1400#  1405#  1411#  1414#  1415#  1416#  1417#  1419#  1425#  1426#  1427#  1428#  1430#
                        1438#  1477#  1478#  1481#  1482#  1504#  1505#  1508#  1510#  1511#  1512#  1513#  1514#
                        1519#  1542#  1556#  1558#  1559#  1560#  1562#  1567#  1575#  1576#  1577#  1583#  1585#
                        1586#  1587#  1589#  1597#  1598#  1605#  1608#  1610#  1611#  1613#  1614#  1615#  1627#
                        1639#  1640#  1642#  1649#  1691#  1693#  1695#  1697#  1698#  1699#  1701#  1702#  1704#
                        1706#  1707#  1708#  1713#  1747#  1748#  1749#  1750#  1752#  1757#  1802#  1803#  1807#
                        1808#  1809#  1813#  1814#  1820#  1821#  1827#  1828#  1834#  1872#  1974#  1976#  1977#
                        1979#  1981#  1985#  1988#  1994#  1999#  2004#  2057#  2060#  2061#  2062#  2063#  2064#
                        2066#  2068#  2069#  2074#  2075#  2080#  2137#  2138#  2139#  2140#  2141#  2142#  2143#
                        2144#  2164#  2212#  2213#  2216#  2217#  2219#  2221#  2225#  2226#  2227#  2229#  2240#
                        2245#  2251#  2256#  2257#  2260#  2262#  2267#  2270#  2273#  2276#  2278#  2279#  2284#
                        2285#  2286#  2287#  2288#  2289#  2292#  2293#  2294#  2295#  2305#  2306#  2311#  2312#
                        2313#  2322#  2323#  2326#  2328#  2374#  2386#  2387#  2389#  2391#
$FSAND= 000310           661#   1052   1076   1081   1093   1125   1129   1141   1186   1197   1208   1212   1219
                        1220   1226   1230   1239   1254   1300   1316   1353   1433   1479   1517   1521   1557
                        1566   1574   1584   1630   1692   1982   1989   2000   2067   2168   2209   2218   2228
                        2261   2266   2300   2317   2385   2390
$FSBAD= 000401           661#   1030   1033   1034   1038   1039   1044   1045   1052   1055   1056   1057   1070
                        1071   1072   1073   1075   1076   1077   1078   1081   1082   1083   1084   1086   1089
                        1090   1093   1094   1097   1099   1125   1129   1130   1131   1135   1136   1137   1140
                        1141   1142   1143   1144   1145   1146   1147   1148   1152   1153   1159   1160   1186
                        1187   1188   1190   1191   1197   1203   1204   1208   1212   1213   1214   1219   1220
                        1224   1226   1227   1229   1230   1231   1232   1233   1239   1249   1250   1251   1252
                        1253   1254   1255   1262   1266   1267   1294   1295   1296   1298   1299   1300   1315
                        1316   1318   1343   1345   1353   1364   1365   1366   1372   1373   1376   1377   1378
                        1379   1380   1384   1385   1389   1393   1394   1395   1400   1405   1411   1414   1415
```

```
                    1416  1417  1419  1425  1426  1427  1428  1430  1433  1438  1477  1478  1479
                    1481  1482  1504  1505  1508  1510  1511  1512  1513  1514  1517  1519  1521
                    1542  1556  1557  1558  1559  1560  1562  1566  1567  1574  1575  1576  1577
                    1583  1584  1585  1586  1587  1589  1597  1598  1605  1608  1610  1611  1613
                    1614  1615  1627  1630  1639  1640  1642  1649  1691  1692  1693  1695  1697
                    1698  1699  1701  1702  1704  1706  1707  1708  1713  1747  1748  1749  1750
                    1752  1757  1802  1803  1807  1808  1809  1813  1814  1820  1821  1827  1828
                    1834  1872  1974  1976  1977  1979  1981  1982  1985  1988  1989  1994  1999
                    2000  2004  2057  2060  2061  2062  2063  2064  2066  2067  2068  2069  2074
                    2075  2080  2137  2138  2139  2140  2141  2142  2143  2144  2164  2168  2209
                    2212  2213  2216  2217  2218  2219  2221  2225  2226  2227  2228  2229  2240
                    2245  2251  2256  2257  2260  2261  2262  2266  2267  2270  2273  2276  2278
                    2279  2284  2285  2286  2287  2288  2289  2292  2293  2294  2295  2300  2305
                    2306  2311  2312  2313  2317  2322  2323  2326  2328  2374  2385  2386  2387
                    2389  2390  2391
$F$BLA= 000170        661#
$F$CAS= 000150        661#
$F$DEC= 000220        661#  1075  1091  1345  1347  1567  1569  1598  1600  1994  1996
$F$DO = 000340        661#  1076  1226  1316  1479  1630  1982  1989  2067
$F$FAL= 000405        661#  1030
$F$GOO= 000400        661#  1030  1033  1034  1038  1039  1044  1045  1052  1055  1056  1057  1070
                    1071  1072  1073  1075  1076  1077  1078  1081  1082  1083  1084  1086  1089
                    1090  1093  1094  1097  1099  1125  1129  1130  1131  1135  1136  1137  1140
                    1141  1142  1143  1144  1145  1146  1147  1148  1152  1153  1159  1160  1186
                    1187  1188  1190  1191  1197  1203  1204  1208  1212  1213  1214  1219  1220
                    1224  1226  1227  1229  1230  1231  1232  1233  1239  1249  1250  1251  1252
                    1253  1254  1255  1262  1266  1267  1294  1295  1296  1298  1299  1300  1315
                    1316  1318  1343  1345  1353  1364  1365  1366  1372  1373  1376  1377  1378
                    1379  1380  1384  1385  1389  1393  1394  1395  1400  1405  1411  1413  1414
                    1415  1416  1417  1419  1424  1425  1426  1427  1428  1430  1433  1438  1477
                    1478  1479  1481  1482  1504  1505  1508  1510  1511  1512  1513  1514  1517
                    1519  1521  1542  1556  1557  1558  1559  1560  1562  1566  1567  1574  1575
                    1576  1577  1583  1584  1585  1586  1587  1589  1597  1598  1605  1608  1610
                    1611  1613  1614  1615  1627  1630  1639  1640  1642  1649  1691  1692  1693
                    1695  1697  1698  1699  1701  1702  1704  1706  1707  1708  1713  1747  1748
                    1749  1750  1752  1757  1802  1803  1807  1808  1809  1813  1814  1820  1821
                    1827  1828  1834  1872  1974  1976  1977  1979  1981  1982  1985  1988  1989
                    1994  1999  2000  2004  2057  2060  2061  2062  2063  2064  2066  2067  2068
                    2069  2074  2075  2080  2137  2138  2139  2140  2141  2142  2143  2144  2164
                    2168  2209  2212  2213  2216  2217  2218  2219  2221  2225  2226  2227  2228
                    2229  2240  2245  2251  2256  2257  2260  2261  2262  2266  2267  2270  2273
                    2276  2278  2279  2284  2285  2286  2287  2288  2289  2292  2293  2294  2295
                    2300  2305  2306  2311  2312  2313  2317  2322  2323  2326  2328  2374  2385
                    2386  2387  2389  2390  2391
$F$IF  = 000110        661#  1052  1054  1081  1085  1087  1093  1095  1125  1129  1138  1141  1149
                    1154  1155  1156  1163  1186  1189  1192  1197  1199  1208  1212  1215  1219
                    1220  1225  1230  1234  1236  1237  1238  1239  1244  1254  1256  1257  1258
                    1300  1303  1353  1356  1413  1421  1424  1431  1433  1436  1517  1520  1521
                    1523  1557  1563  1566  1570  1574  1579  1584  1590  1692  1694  1696  2000
                    2002  2168  2170  2172  2209  2211  2218  2222  2228  2231  2261  2264  2266
                    2272  2274  2300  2304  2308  2317  2321  2325  2385  2388  2390  2392
$F$INC= 000210        661#  1373  1406  1505  1515  1691  1698  1705  1708  1711  1712  1747  1749
                    1751  1755  1808  1813  1815  1820  1822  1835  1976  2007  2063  2065  2164
                    2174  2216  2223  2225  2232  2292  2309  2311  2327
$F$LOO= 000200        661#
$F$NAM= 000160        661#
```

```
                    1752    1757    1802    1803    1807    1808    1809    1813    1814    1820    1821    1827    1828
                    1834    1872    1974    1976    1977    1979    1981    1982    1985    1988    1989    1994    1999
                    2000    2002    2004    2057    2060    2061    2062    2063    2064    2066    2067    2068    2069
                    2074    2075    2080    2137    2138    2139    2140    2141    2142    2143    2144    2164    2168
                    2170    2172    2209    2211    2212    2213    2216    2217    2218    2219    2221    2222    2225
                    2226    2227    2228    2229    2231    2240    2245    2251    2256    2257    2260    2261    2262
                    2264    2266    2267    2270    2272    2273    2274    2276    2278    2279    2284    2285    2286
                    2287    2288    2289    2292    2293    2294    2295    2300    2304    2305    2306    2308    2311
                    2312    2313    2317    2321    2322    2323    2325    2326    2328    2374    2385    2386    2387
                    2388    2389    2390    2391    2392
$IFLEV= 177777       661#   1052#   1054#   1081#   1087#   1093#   1095#   1125#   1129#   1141#   1154#   1155#   1163#
                    1186#   1192#   1197#   1199#   1208#   1212#   1219#   1220#   1230#   1234#   1236#   1237#   1238#
                    1239#   1254#   1256#   1257#   1258#   1300#   1303#   1353#   1356#   1413#   1421#   1424#   1431#
                    1433#   1436#   1517#   1520#   1521#   1523#   1557#   1563#   1566#   1570#   1574#   1575#   1584#
                    1590#   1692#   1696#   2000#   2002#   2168#   2172#   2209#   2211#   2218#   2222#   2228#   2231#
                    2261#   2264#   2266#   2274#   2300#   2308#   2317#   2325#   2385#   2388#   2390#   2392#
$ISK0 = 000001       1052#   1054    1081#   1087    1093#   1095    1125#   1163    1186#   1192    1197#   1199    1208#
                    1258    1300#   1303    1353#   1356    1413#   1421    1424#   1431    1433#   1436    1517#   1520
                    1521#   1523    1557#   1563    1566#   1570    1574#   1579    1584#   1590    1692#   1696    2000#
                    2002    2168#   2172    2209#   2211    2218#   2222    2228#   2231    2261#   2264    2266#   2274
                    2300#   2308    2317#   2325    2385#   2388    2390#   2392
$ISK1 = 000001       1129#   1155    1212#   1238    1239#   1257
$ISK2 = 000001       1141#   1154    1219#   1237    1254#   1256
$ISK3 = 000001       1220#   1236
$ISK4 = 000001       1230#   1234
$LO   = 177777       1030#   1032#   1037#   1042#
$LOCTA= 177777        661#    664#   1030    1032    1037    1042    1050    1052    1054    1075    1076    1079    1081
                    1085    1087    1091    1093    1095    1125    1129    1138    1141    1149    1154    1155    1156
                    1163    1186    1189    1192    1197    1199    1208    1212    1215    1219    1220    1225    1226
                    1230    1234    1235    1236    1237    1238    1239    1244    1254    1256    1257    1258    1300
                    1303    1316    1319    1345    1347    1353    1356    1373    1406    1413    1421    1424    1431
                    1433    1436    1479    1483    1505    1515    1517    1520    1521    1523    1557    1563    1566
                    1567    1569    1570    1574    1579    1584    1590    1598    1600    1630    1631    1691    1692
                    1694    1696    1698    1705    1708    1711    1712    1747    1749    1751    1755    1808    1813
                    1815    1820    1822    1835    1976    1982    1984    1989    1991    1994    1996    2000    2002
                    2007    2059    2063    2065    2067    2070    2079    2164    2168    2170    2172    2174    2209
                    2211    2216    2218    2222    2223    2225    2228    2231    2232    2258    2261    2264    2265
                    2266    2272    2274    2293    2300    2304    2308    2309    2311    2317    2321    2325    2327
                    2385    2388    2390    2392
$LSK0 = 000000       1030#   1032    1037    1042
$LSTIN= 000000        661#    662#   1030    1033    1034    1037    1038    1039    1042    1044    1045    1050    1052
                    1055    1056    1057    1069    1070    1071    1072    1073    1075    1076    1077    1078    1079
                    1081    1082    1083    1084    1085    1086    1089    1090    1091    1093    1094    1097    1098
                    1099    1125    1129    1130    1131    1135    1136    1137    1138    1140    1141    1142    1143
                    1144    1145    1146    1147    1148    1149    1152    1153    1156    1159    1160    1164    1181
                    1186    1187    1188    1189    1190    1191    1197    1198    1203    1204    1208    1212    1213
                    1214    1215    1219    1220    1224    1225    1226    1227    1229    1230    1231    1232    1233
                    1235    1239    1244    1249    1250    1251    1252    1253    1254    1255    1262    1266    1267
                    1270    1294    1295    1296    1298    1299    1300    1315    1316    1318    1319    1343    1345
                    1347    1353    1364    1365    1366    1372    1373    1376    1377    1378    1379    1380    1384
                    1385    1389    1393    1394    1395    1400    1405    1406    1411    1413    1414    1415    1416
                    1417    1419    1424    1425    1426    1427    1428    1430    1433    1438    1476    1477    1478
                    1479    1481    1482    1483    1484    1504    1505    1508    1510    1511    1512    1513    1514
                    1515    1517    1519    1521    1542    1556    1557    1558    1559    1560    1562    1566    1567
                    1569    1574    1575    1576    1577    1583    1584    1585    1586    1587    1589    1597    1598
                    1600    1605    1608    1610    1611    1613    1614    1615    1627    1630    1631    1639    1640
```

	1642	1649	1691	1692	1693	1694	1695	1697	1698	1699	1701	1702	1704
	1705	1706	1707	1708	1711	1712	1713	1747	1748	1749	1750	1751	1752
	1755	1757	1802	1803	1807	1808	1809	1813	1814	1815	1820	1821	1822
	1827	1828	1834	1835	1872	1974	1976	1977	1979	1981	1982	1984	1985
	1988	1989	1991	1994	1996	1999	2000	2004	2007	2057	2060	2061	2062
	2063	2064	2065	2066	2067	2068	2069	2070	2074	2075	2079	2080	2137
	2138	2139	2140	2141	2142	2143	2144	2164	2168	2170	2174	2209	2212
	2213	2216	2217	2218	2219	2221	2223	2225	2226	2227	2228	2229	2232
	2240	2245	2251	2256	2257	2260	2261	2262	2265	2266	2267	2270	2271
	2272	2273	2276	2278	2279	2284	2285	2286	2287	2288	2289	2292	2293
	2294	2295	2300	2304	2305	2306	2309	2311	2312	2313	2317	2321	2322
	2323	2326	2327	2328	2374	2385	2386	2387	2389	2390	2391		
$LSTTA= 000000	661#	663#	1030	1032	1037	1042	1050	1054	1075	1076	1079	1081	1085
	1087	1091	1095	1138	1149	1154	1155	1156	1163	1189	1192	1199	1215
	1225	1226	1234	1235	1236	1237	1238	1244	1256	1257	1258	1303	1316
	1319	1345	1347	1356	1373	1406	1421	1431	1436	1479	1483	1505	1515
	1520	1523	1563	1567	1569	1570	1579	1590	1598	1600	1630	1631	1691
	1694	1696	1698	1705	1708	1711	1712	1747	1749	1751	1755	1808	1813
	1815	1820	1822	1835	1976	1982	1984	1989	1991	1994	1996	2002	2007
	2059	2063	2065	2067	2070	2164	2170	2172	2174	2211	2216	2222	2223
	2225	2231	2232	2258	2261	2264	2265	2272	2274	2292	2304	2308	2309
	2311	2321	2325	2327	2388	2392							
$NESTL= 177777	661#	1030#	1032	1037	1042	1050#	1052#	1054#	1075#	1076#	1079#	1081#	1085
	1087#	1091#	1093#	1095#	1125#	1129#	1138	1141#	1149	1154#	1155#	1156	1163#
	1186#	1189	1192#	1197#	1199#	1208#	1212#	1215	1219#	1220#	1225	1226#	1230#
	1234#	1235#	1236#	1237#	1238#	1239#	1244	1254#	1256#	1257#	1258#	1300#	1303#
	1316#	1319#	1345#	1347#	1353#	1356#	1373#	1406#	1413#	1421#	1424#	1431#	1433#
	1436#	1479#	1483#	1505#	1515#	1517#	1520#	1521#	1523#	1557#	1563#	1566#	1567#
	1569#	1570#	1574#	1579#	1584#	1590#	1598#	1600#	1630#	1631#	1691#	1692#	1694
	1696#	1698#	1705#	1708#	1711#	1712#	1747#	1749#	1751#	1755#	1808#	1813#	1815#
	1820#	1822#	1835#	1976#	1982#	1984#	1989#	1991#	1994#	1996#	2000#	2002#	2007#
	2059#	2063#	2065#	2067#	2070#	2079#	2164#	2168#	2170	2172#	2174#	2209#	2211#
	2216#	2218#	2222#	2223#	2225#	2228#	2231#	2232#	2258#	2261#	2264#	2265#	2266#
	2272	2274#	2292#	2300#	2304	2308#	2309#	2311#	2317#	2321	2325#	2327#	2355#
	2388#	2390#	2392#										
$NSK0 = 000110	1030#	1032	1037	1042	1050	1052#	1054	1075#	1091	1093#	1095	1125#	1156
	1163	1186#	1189	1192	1197#	1199	1208#	1258	1300#	1303	1316#	1319	1345#
	1347	1353#	1356	1373#	1406	1413#	1421	1424#	1431	1433#	1436	1479#	1483
	1505#	1515	1517#	1520	1521#	1523	1557#	1563	1566#	1570	1574#	1579	1584#
	1590	1598#	1600	1630#	1631	1691#	1712	1747#	1755	1808#	1835	1976#	2007
	2059#	2079	2164#	2174	2209#	2211	2216#	2223	2225#	2232	2258#	2265	2266#
	2272	2274	2292#	2309	2311#	2327	2385#	2388	2390#	2392			
$NSK1 = 000110	1076#	1079	1081#	1085	1087	1129#	1138	1155	1212#	1215	1238	1239#	1244
	1257	1567#	1569	1692#	1694	1696	1698#	1705	1708#	1711	1749#	1751	1813#
	1815	1820#	1822	1982#	1984	1989#	1991	1994#	1996	2000#	2002	2063#	2065
	2067#	2070	2168#	2170	2172	2218#	2222	2228#	2231	2261#	2264	2300#	2304
	2308	2317#	2321	2325									
$NSK2 = 000110	1141#	1149	1154	1219#	1237	1254#	1256						
$NSK3 = 000110	1220#	1225	1236										
$NSK4 = 000120	1226#	1235											
$NSK5 = 000110	1230#	1234											
$SAVE = 050004	661#	1030#	1050#										
$SAVE2= 050005	1030#												
$SAVLE= 177777	661#	1030#	1050#	1075#	1079#	1235#	1319#	1345#	1373#	1483#	1505#	1567#	1598#
	1631#	1691#	1698#	1708#	1751#	1755#	1808#	1813#	1820#	1984#	1991#	1996#	2007#
	2063#	2070#	2164#	2216#	2225#	2292#	2311#						

```
$SELLE= 000000   661#  1030#  1032   1037   1042
$SSK0 = 050217  1030#  1050   1075#  1079#  1235#  1319#  1345#  1373#  1483#  1505#  1567#  1598#  1631#
                1691#  1698#  1708#  1751#  1755#  1808#  1813#  1820#  1984#  1991#  1996#  2007#  2063#
                2070#  2164#  2216#  2225#  2292#  2311#
$SSK1 = 000402  1030#  1050
$SSK2 = 050005  1030#  1050
$TAGLE= 177777   661#  1030#  1032   1037   1042   1050#  1052#  1054#  1075#  1076#  1079#  1081#  1085#
                1087#  1091#  1093#  1095#  1125#  1129#  1138#  1141#  1149#  1154#  1155#  1156#  1163#
                1186#  1189#  1192#  1197#  1199#  1208#  1212#  1215#  1219#  1220#  1225#  1226#  1230#
                1234#  1235#  1236#  1237#  1238#  1239#  1244#  1254#  1256#  1257#  1258#  1300#  1303#
                1316#  1319#  1345#  1347#  1353#  1356#  1373#  1406#  1413#  1421#  1424#  1431#  1433#
                1436#  1479#  1483#  1505#  1515#  1517#  1520#  1521#  1523#  1557#  1563#  1566#  1567#
                1569#  1570#  1574#  1579#  1584#  1590#  1598#  1600#  1630#  1631#  1691#  1692#  1694#
                1696#  1698#  1705#  1708#  1711#  1712#  1747#  1749#  1751#  1755#  1808#  1813#  1815#
                1820#  1822#  1835#  1976#  1982#  1984#  1989#  1991#  1994#  1996#  2000#  2002#  2007#
                2059#  2063#  2065#  2067#  2070#  2079#  2164#  2168#  2170#  2172#  2174#  2209#  2211#
                2216#  2218#  2222#  2223#  2225#  2228#  2231#  2232#  2258#  2261#  2264#  2265#  2266#
                2272#  2274#  2292#  2300#  2304#  2308#  2309#  2311#  2317#  2321#  2325#  2327#  2385#
                2388#  2390#  2392#
$TAGNU- 050225   661#  1030#  1052#  1075#  1076#  1081#  1085#  1093#  1125#  1129#  1138#  1141#  1149#
                1156#  1186#  1189#  1197#  1208#  1212#  1215#  1219#  1220#  1225#  1226#  1230#  1239#
                1244#  1254#  1300#  1316#  1345#  1353#  1373#  1413#  1424#  1433#  1479#  1505#  1517#
                1521#  1557#  1566#  1567#  1574#  1584#  1598#  1630#  1691#  1692#  1694#  1698#  1708#
                1747#  1749#  1808#  1813#  1820#  1976#  1982#  1989#  1994#  2000#  2059#  2063#  2067#
                2164#  2168#  2170#  2209#  2216#  2218#  2225#  2228#  2258#  2261#  2265#  2266#  2272#
                2292#  2300#  2304#  2311#  2317#  2321#  2385#  2390#
$TEMP - 050224   661#  1030#  1032#  1033#  1034#  1037#  1038#  1039#  1042#  1044#  1045#  1050#  1054#
                1055#  1056#  1057#  1070#  1071#  1072#  1073#  1075#  1077#  1078#  1079#  1082#  1083#
                1084#  1085#  1086#  1087#  1089#  1090#  1091#  1094#  1095#  1097#  1099#  1130#  1131#
                1135#  1136#  1137#  1138#  1140#  1142#  1143#  1144#  1145#  1146#  1147#  1148#  1149#
                1152#  1153#  1154#  1155#  1156#  1159#  1160#  1163#  1187#  1188#  1189#  1190#  1191#
                1192#  1199#  1203#  1204#  1213#  1214#  1215#  1224#  1225#  1227#  1229#  1231#  1232#
                1233#  1234#  1235#  1236#  1237#  1238#  1244#  1249#  1250#  1251#  1252#  1253#  1255#
                1256#  1257#  1258#  1262#  1266#  1267#  1294#  1295#  1296#  1298#  1299#  1303#  1315#
                1318#  1319#  1343#  1345#  1347#  1356#  1364#  1365#  1366#  1372#  1373#  1376#  1377#
                1378#  1379#  1380#  1384#  1385#  1389#  1393#  1394#  1395#  1400#  1405#  1406#  1411#
                1414#  1415#  1416#  1417#  1419#  1421#  1425#  1426#  1427#  1428#  1430#  1431#  1436#
                1438#  1477#  1478#  1481#  1482#  1483#  1504#  1505#  1508#  1510#  1511#  1512#  1513#
                1514#  1515#  1519#  1520#  1523#  1542#  1556#  1558#  1559#  1560#  1562#  1563#  1567#
                1569#  1570#  1575#  1576#  1577#  1579#  1583#  1585#  1586#  1587#  1589#  1590#  1597#
                1598#  1600#  1605#  1608#  1610#  1611#  1613#  1614#  1615#  1627#  1631#  1639#  1640#
                1642#  1649#  1691#  1693#  1694#  1695#  1696#  1697#  1698#  1699#  1701#  1702#  1704#
                1705#  1706#  1707#  1708#  1711#  1712#  1713#  1747#  1748#  1749#  1750#  1751#  1752#
                1755#  1757#  1802#  1803#  1807#  1808#  1809#  1813#  1814#  1815#  1820#  1821#  1822#
                1827#  1828#  1834#  1835#  1872#  1974#  1976#  1977#  1979#  1981#  1984#  1985#  1988#
                1991#  1994#  1996#  1999#  2002#  2004#  2007#  2057#  2060#  2061#  2062#  2063#  2064#
                2065#  2066#  2068#  2069#  2070#  2074#  2075#  2079#  2080#  2137#  2138#  2139#  2140#
                2141#  2142#  2143#  2144#  2164#  2170#  2172#  2174#  2211#  2212#  2213#  2216#  2217#
                2219#  2221#  2222#  2223#  2225#  2226#  2227#  2229#  2231#  2232#  2240#  2245#  2251#
                2256#  2257#  2260#  2262#  2264#  2265#  2267#  2270#  2272#  2273#  2274#  2276#  2278#
                2279#  2284#  2285#  2286#  2287#  2288#  2289#  2292#  2293#  2294#  2295#  2304#  2305#
                2306#  2308#  2309#  2311#  2312#  2313#  2321#  2322#  2323#  2325#  2326#  2327#  2329#
                2374#  2386#  2387#  2388#  2389#  2391#  2392#
$TSK0 = 050224  1030#  1050   1052#  1054   1075#  1091   1093#  1095   1125#  1156#  1163   1186#  1189#
                1192   1197#  1199   1208#  1258   1300#  1303   1316#  1319   1345#  1347   1353#  1356
                1373#  1406   1413#  1421   1424#  1431   1433#  1436   1479#  1483   1505#  1515   1517#
```

```
                    1520    1521#   1523    1557#   1563    1566#   1570    1574#   1579    1584#   1590    1598#   1600
                    1630#   1631    1691#   1712    1747#   1755    1808#   1835    1976#   2007    2059#   2079    2164#
                    2174    2209#   2211    2216#   2223    2225#   2232    2258#   2265    2266#   2272#   2274    2292#
                    2309    2311#   2327    2385#   2388    2390#   2392
$TSK1 = 050217      1030#   1042#   1050    1075#   1091    1129#   1138#   1155    1212#   1215#   1238    1239#   1244#
                    1257    1316#   1319    1345#   1347    1373#   1406    1479#   1483    1505#   1515    1567#   1569
                    1598#   1600    1630#   1631    1691#   1712    1747#   1755    1808#   1835    1976#   2007    2063#
                    2065    2067#   2070    2164#   2174    2216#   2223    2225#   2232    2261#   2264    2292#   2309
                    2311#   2327
$TSK2 = 050222      1030#   1037#   1050    1076#   1079    1081#   1085#   1087    1141#   1149#   1154    1219#   1237
                    1254#   1256    1567#   1569    1692#   1694#   1696    1698#   1705    1708#   1711    1749#   1751
                    1813#   1815    1820#   1822    1982#   1984    1989#   1991    1994#   1996    2000#   2002    2063#
                    2065    2067#   2070    2168#   2170#   2172    2218#   2222    2228#   2231    2300#   2304#   2308
                    2317#   2321#   2325
$TSK3 = 050155      1030#   1032#   1050    1076#   1079    1220#   1225#   1236    1698#   1705    1708#   1711    1749#
                    1751    1813#   1815    1820#   1822    1982#   1984    1989#   1991    1994#   1996
$TSK4 = 050040      1030#   1050#   1226#   1235
$TSK5 = 050041      1030#   1037    1042    1050    1226#   1235
$TSK6 = 050042      1230#   1234
$U    = 000403      1075#   1345#   1373#   1505#   1567#   1598#   1691#   1698#   1708#   1747#   1749#   1808#   1813#
                    1820#   1976#   1994#   2063#   2164#   2216#   2225#   2292#   2311#
$$ARGC= 000000       661#
$$BYTE= 000403       661#   1052#   1076#   1081#   1093#   1125#   1129#   1141#   1186#   1197#   1208#   1212#   1219#
                    1220#   1226#   1230#   1239#   1254#   1300#   1316#   1353#   1433#   1479#   1517#   1521#   1557#
                    1566#   1574#   1584#   1630#   1692#   1982#   1989#   2000#   2067#   2168#   2209#   2218#   2228#
                    2261#   2266#   2300#   2317#   2385#   2390#
$$CASE= 000404       661#   1030#   1032#   1037#   1042#
$$DST = 000037       661#   2061#
$$ELOC= 000402       661#   1052#   1054#   1081#   1083    1087#   1093#   1095#   1125#   1129#   1138    1141#   1149
                    1154#   1155#   1156    1163#   1186#   1189    1192#   1197#   1199#   1208#   1212#   1215    1219#
                    1220#   1225    1230#   1234#   1236#   1237#   1238#   1239#   1244    1254#   1256#   1257#   1258#
                    1300#   1303#   1353#   1356#   1413#   1421#   1424#   1431#   1433#   1436#   1517#   1520#   1521#
                    1523#   1557#   1563#   1566#   1570#   1574#   1579#   1584#   1590#   1692#   1694    1696#   2000#
                    2002#   2168#   2170    2172#   2209#   2211#   2218#   2222#   2228#   2231#   2261#   2264#   2266#
                    2272    2274#   2300#   2304    2308#   2317#   2321    2325#   2385#   2388#   2390#   2392#
$$ERFL= 000000       661#
$$FLAG= 000001       661#   1052#   1054#   1076#   1081#   1087#   1093#   1095#   1125#   1129#   1141#   1154#   1155#
                    1163#   1186#   1192#   1197#   1199#   1208#   1212#   1219#   1220#   1226#   1230#   1234#   1236#
                    1237#   1238#   1239#   1254#   1256#   1257#   1258#   1300#   1303#   1316#   1353#   1356#   1413#
                    1421#   1424#   1431#   1433#   1436#   1479#   1517#   1520#   1521#   1523#   1557#   1563#   1566#
                    1570#   1574#   1579#   1584#   1590#   1630#   1692#   1696#   1982#   1989#   2000#   2002#   2067#
                    2168#   2172#   2209#   2211#   2218#   2222#   2228#   2231#   2261#   2264#   2266#   2274#   2300#
                    2308#   2317#   2325#   2385#   2388#   2390#   2392#
$$FROM= 000000       661#
$$INH = 000403       661#   1030#   1052#   1076#   1081#   1093#   1125#   1129#   1141#   1186#   1197#   1208#   1212#
                    1219#   1220#   1226#   1230#   1239#   1254#   1300#   1316#   1353#   1413#   1424#   1433#   1479#
                    1517#   1521#   1557#   1566#   1574#   1584#   1630#   1692#   1982#   1989#   2000#   2067#   2168#
                    2209#   2218#   2228#   2261#   2265#   2266#   2300#   2317#   2385#   2390#
$$LOC = 022120       661#   1030#   1052#   1076#   1081#   1093#   1125#   1129#   1141#   1186#   1197#   1208#   1212#
                    1219#   1220#   1226#   1230#   1239#   1254#   1300#   1316#   1353#   1413#   1424#   1433#   1479#
                    1517#   1521#   1557#   1566#   1574#   1584#   1630#   1692#   1751#   1755#   1984#   1991#   1996#
                    2007#   2067#   2079#   2168#   2209#   2218#   2228#   2261#   2265#   2266#   2300#   2317#   2385#
                    2390#
$$LOCN= 000000       661#
$$REG = 177777       661#
$$RETU= 000000       661#
```

```
$$RTN1= 000000            661#
$$RTN2= 000000            661#
$$SRC = 000027            661#   2061#
$$TGSV= 050006            661#   1050#
$$TGS1= 050005            661#   1032#   1037#   1042#   1050#
$$TGS2= 000000            661#
$$TO  = 000000            661#
$$$TAG= 050000            661#
.     = 022370            675#    703#    981#   1030    1052    1076    1081    1093    1125    1129    1141    1186    1197
                         1208    1212    1219    1220    1226    1228    1230    1239    1254    1300    1316    1346    1348
                         1353    1413    1424    1433    1440    1479    1517    1521    1557    1566    1568    1574    1584
                         1599    1609    1630    1632    1660#   1692    1716    1751    1755    1761    1875    1982    1984
                         1989    1991    1996    2000    2007    2009    2027#   2067    2079    2083    2168    2177    2207
                         2209    2210    2218    2228    2261    2265    2266    2300    2317    2331    2350#   2385    2390
                         2432#   2435#
```

```
ALLOCA    661#
BCOMPL   1337    1339    2206
BEGIN     661#
BGNAUT    727
BGNCLN   1500
BGNHRD   2407
BGNHW     716
BGNINI   1333
BGNMOD    677     806    1332    1530    1668    1732    1775    1954    2038    2102    2193    2399
BGNPRO    685
BGNSFT   2424
BGNSRV   1638    1647    2358    2383
BGNSW     740
BGNIST   1541    1683    1738    1798    1970    2048    2135    2203
BNCOMP   1358    1375
BRESET   1352    1502
CALL      661#
CASE      661#   1032    1037    1042
CLOCK    1412    1423
CLRVEC   1518
DEALLO    661#
DECLAR    661#
DECR      661#   1075    1345    1567    1598    1994
DECRB     661#
DECRU     661#
DECRUB    661#
DEFAUL    661#
DELAY    1228    1346    1568    1599    1609
DESCRI    702
DEVTYP    703
DISPAT    698
DOCLN    1302
ELSE      661#   1085    1138    1149    1156    1189    1215    1225    1244    1694    2170    2272    2304    2321
END       661#
ENDAUT    731
ENDCLN   1525
ENDDEC    661#   1091    1347    1569    1600    1996
ENDDO     661#   1079    1235    1319    1483    1631    1984    1991    2070
ENDHRD   2411
ENDHW     723
ENDIF     661#   1054    1087    1095    1154    1155    1163    1192    1199    1234    1236    1237    1238    1256    1257
          1258   1303    1356    1421    1431    1436    1520    1523    1563    1570    1579    1590    1696    2002    2172
          2211   2222    2231    2264    2274    2308    2325    2388    2392
ENDINC    661#   1406    1515    1705    1711    1712    1751    1755    1815    1822    1835    2007    2065    2174    2223
          2232   2309    2327
ENDINI   1455
ENDLOO    661#
ENDMOD    802    1323    1527    1663    1729    1771    1950    2034    2098    2190    2355    2437
ENDPRO    689
ENDROU    661#
ENDRTI    661#
ENDRTN    661#
ENDSEL    661#   1050
ENDSFT   2427
ENDSRV   1643    1650    2394
ENDSW     749
```

```
M$DEFA   1368#  2255#  2277#  2299#  2303#  2316#  2320#  2409#  2410#  2425#  2426#
M$ENDE    723#   731#   749#   802#  1323#  1455#  1525#  1527#  1643#  1650#  1661#  1663#  1727#  1729#  1769#
         1771#  1948#  1950#  2033#  2034#  2097#  2098#  2188#  2190#  2353#  2355#  2394#  2411#  2427#  2437#
M$ERRI   1035#  1040#  1046#  1561#  1578#  1588#  1612#  1641#  2220#  2230#  2269#  2307#  2324#
M$EXCP   2409#  2410#  2426#
M$EXIT   1348#  1440#  1632#  1716#  1761#  1875#  2009#  2083#  2177#  2207#  2210#  2331#
M$EXSE   1348#  1440#  1632#  1716#  1761#  1875#  2009#  2083#  2177#  2207#  2210#  2331#
M$EXTJ   1348#  1440#  1632#  1716#  1761#  1875#  2009#  2083#  2177#  2207#  2210#  2331#
M$GEN     680#   685#   698#   702#   703#   716#   723#   727#   731#   740#   749#  1333#  1368#  1455#  1500#
         1525#  1541#  1638#  1643#  1647#  1650#  1661#  1683#  1727#  1738#  1769#  1798#  1948#  1970#  2033#
         2048#  2097#  2135#  2188#  2203#  2255#  2277#  2299#  2303#  2316#  2320#  2353#  2358#  2383#  2394#
         2407#  2411#  2424#  2427#  2436#
M$GENB   1368#  2255#  2277#  2299#  2303#  2316#  2320#
M$GETS    689#   723#   731#   749#   802#  1323#  1455#  1525#  1527#  1643#  1650#  1661#    63#  1727#  1729#
         1769#  1771#  1948#  1950#  2033#  2034#  2097#  2098#  2188#  2190#  2353#  2355#  2394#  2411#  2427#
         2437#
M$GETT   1348#  1440#  1632#  1716#  1761#  1875#  2009#  2083#  2177#  2207#  2210#  2331#
M$GNGB    677#   680#   685#   698#   702#   703#   716#   727#   740#   806#  1332#  1333#  1500#  1530#  1638#
         1647#  1668#  1732#  1775#  1954#  2038#  2102#  2193#  2358#  2383#  2399#  2407#  2424#  2436#
M$GNIN    680#   698#   702#   703#   716#   731#   740#  1035#  1040#  1046#  1228#  1297#  1301#  1302#  1317#
         1336#  1337#  1338#  1339#  1342#  1346#  1348#  1352#  1354#  1355#  1357#  1358#  1360#  1368#  1374#
         1375#  1396#  1412#  1420#  1423#  1434#  1435#  1437#  1440#  1455#  1480#  1501#  1502#  1518#  1522#
         1524#  1525#  1561#  1568#  1578#  1588#  1595#  1596#  1599#  1606#  1607#  1609#  1612#  1622#  1632#
         1641#  1643#  1650#  1661#  1716#  1727#  1761#  1769#  1875#  1948#  2009#  2033#  2083#  2097#  2177#
         2188#  2205#  2206#  2207#  2210#  2220#  2230#  2252#  2253#  2254#  2255#  2269#  2275#  2277#  2296#
         2297#  2298#  2299#  2301#  2302#  2303#  2307#  2314#  2315#  2316#  2318#  2319#  2320#  2324#  2331#
         2353#  2374#  2384#  2394#  2407#  2409#  2410#  2411#  2424#  2425#  2426#  2427#  2436#
M$GNLS   1368#  2255#  2277#  2299#  2303#  2316#  2320#
M$GNTA    723#   731#   749#  1455#  1525#  1643#  1650#  1661#  1727#  1769#  1948#  2033#  2097#  2188#  2353#
         2394#  2411#  2427#
M$GNTE   1541#  1683#  1738#  1798#  1970#  2048#  2135#  2203#
M$HAPT    680#
M$HNAP    680#
M$INCR    677#   685#   716#   727#   731#   740#   806#  1035#  1040#  1046#  1297#  1301#  1302#  1317#  1332#
         1333#  1336#  1338#  1342#  1348#  1352#  1354#  1355#  1357#  1360#  1368#  1374#  1396#  1412#  1420#
         1423#  1434#  1435#  1437#  1440#  1455#  1480#  1500#  1501#  1502#  1518#  1522#  1524#  1525#  1530#
         1541#  1561#  1578#  1588#  1595#  1596#  1606#  1607#  1612#  1622#  1632#  1638#  1641#  1647#  1661#
         1668#  1683#  1716#  1727#  1732#  1738#  1761#  1769#  1775#  1798#  1875#  1948#  1954#  1970#  2009#
         2033#  2038#  2048#  2083#  2097#  2102#  2135#  2177#  2188#  2193#  2203#  2205#  2207#  2210#  2220#
         2230#  2252#  2253#  2254#  2255#  2269#  2275#  2277#  2296#  2297#  2298#  2299#  2301#  2302#  2303#
         2307#  2314#  2315#  2316#  2318#  2319#  2320#  2324#  2331#  2353#  2358#  2374#  2383#  2384#  2399#
         2407#  2424#
M$LDRO   1317#  1336#  1338#  1342#  1374#  1412#  1423#  1437#  1501#  1518#  1524#  1595#  1606#  1622#  2374#
         2384#
M$MCHI    659#
M$MCLO    659#
M$POP     689#   723#   731#   749#   802#  1323#  1455#  1525#  1527#  1643#  1650#  1661#  1663#  1727#  1729#
         1769#  1771#  1948#  1950#  2033#  2034#  2097#  2098#  2188#  2190#  2353#  2355#  2394#  2411#  2427#
         2437#
M$PRIN   1297#  1301#  1354#  1355#  1360#  1434#  1435#  2252#  2253#  2254#  2275#  2296#  2297#  2298#  2301#
         2302#  2314#  2315#  2318#  2319#
M$PUSH    677#   685#   716#   727#   740#   806#  1332#  1333#  1500#  1530#  1541#  1638#  1647#  1668#  1683#
         1732#  1738#  1775#  1798#  1954#  1970#  2038#  2048#  2102#  2135#  2193#  2203#  2358#  2383#  2399#
         2407#  2424#
M$PUT    1297#  1301#  1354#  1355#  1360#  1396#  1420#  1434#  1435#  1480#  1522#  1596#  1607#  2252#  2253#
         2254#  2275#  2296#  2297#  2298#  2301#  2302#  2314#  2315#  2318#  2319#
```

	1517#	1521#	1557#	1566#	1567#	1569#	1574#	1584#	1598#	1600#	1630#	1631#	1691#	1692#	1694#
	1698#	1705#	1708#	1711#	1712#	1747#	1749#	1751#	1755#	1808#	1813#	1815#	1820#	1822#	1835#
	1976#	1982#	1984#	1989#	1991#	1994#	1996#	2000#	2007#	2059#	2063#	2065#	2067#	2070#	2164#
	2168#	2170#	2174#	2209#	2216#	2218#	2223#	2225#	2228#	2232#	2258#	2261#	2265#	2266#	2272#
	2292#	2300#	2304#	2309#	2311#	2317#	2321#	2327#	2385#	2390#					
$AND	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#	1230#
	1239#	1254#	1300#	1316#	1353#	1433#	1479#	1517#	1521#	1557#	1566#	1574#	1584#	1630#	1692#
	1982#	1989#	2000#	2067#	2168#	2209#	2218#	2228#	2261#	2266#	2300#	2317#	2385#	2390#	
$BKANC	1030#	1032#	1037#	1042#	1052#	1075#	1076#	1079#	1081#	1085#	1091#	1093#	1125#	1129#	1138#
	1141#	1149#	1156#	1186#	1189#	1197#	1208#	1212#	1215#	1219#	1220#	1225#	1226#	1230#	1235#
	1239#	1244#	1254#	1300#	1316#	1319#	1345#	1347#	1353#	1373#	1406#	1413#	1424#	1433#	1479#
	1483#	1505#	1515#	1517#	1521#	1557#	1566#	1567#	1569#	1574#	1584#	1598#	1600#	1630#	1631#
	1691#	1692#	1694#	1698#	1705#	1708#	1711#	1712#	1747#	1749#	1751#	1755#	1808#	1813#	1815#
	1820#	1822#	1835#	1976#	1984#	1991#	1994#	1996#	2007#	2063#	2065#	2067#	2070#	2079#	2164#
	2168#	2170#	2174#	2209#	2216#	2218#	2223#	2225#	2228#	2232#	2261#	2265#	2266#	2272#	2292#
	2300#	2304#	2309#	2311#	2317#	2321#	2327#	2385#	2390#						
$BRCOD	1075#	1081#	1345#	1373#	1505#	1567#	1598#	1691#	1698#	1708#	1747#	1749#	1808#	1813#	1820#
	1976#	1994#	2063#	2164#	2216#	2225#	2261#	2265#	2292#	2311#					
$CHECK	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#	1230#
	1239#	1254#	1300#	1316#	1353#	1433#	1479#	1517#	1521#	1557#	1566#	1574#	1584#	1630#	1692#
	1982#	1989#	2000#	2067#	2168#	2209#	2218#	2228#	2261#	2266#	2300#	2317#	2385#	2390#	
$CKBAS	1030#														
$CKIDB	1747#	1749#	1976#	1994#											
$CKOP1	1056#	1057#	1070#	1071#	1072#	1073#	1075#	1083#	1086#	1090#	1094#	1097#	1130#	1142#	1143#
	1144#	1148#	1153#	1160#	1187#	1188#	1190#	1191#	1204#	1213#	1224#	1227#	1231#	1249#	1250#
	1251#	1252#	1253#	1255#	1262#	1295#	1296#	1298#	1315#	1343#	1345#	1364#	1365#	1366#	1373#
	1377#	1378#	1379#	1380#	1384#	1389#	1395#	1411#	1414#	1415#	1416#	1417#	1419#	1425#	1426#
	1427#	1428#	1430#	1438#	1477#	1478#	1505#	1511#	1512#	1513#	1514#	1519#	1560#	1562#	1567#
	1576#	1583#	1587#	1589#	1598#	1611#	1613#	1615#	1627#	1640#	1642#	1649#	1691#	1693#	1695#
	1697#	1698#	1699#	1702#	1706#	1707#	1708#	1713#	1747#	1748#	1749#	1750#	1752#	1757#	1802#
	1803#	1807#	1808#	1809#	1813#	1814#	1820#	1821#	1827#	1828#	1872#	1976#	1979#	1931#	1985#
	1988#	1994#	2004#	2057#	2060#	2062#	2063#	2064#	2066#	2068#	2080#	2137#	2138#	2139#	2140#
	2141#	2142#	2143#	2144#	2164#	2212#	2216#	2221#	2225#	2227#	2240#	2251#	2256#	2257#	2262#
	2267#	2273#	2276#	2279#	2284#	2285#	2287#	2288#	2289#	2292#	2294#	2295#	2306#	2311#	2313#
	2323#	2326#	2328#	2374#	2386#	2391#									
$CKOP2	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1808#	1813#	1820#	1834#	1974#
	1977#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#	2219#	2225#	2226#	2229#
	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#	2387#	2389#		
$CKR6	2061#														
$CKSEL	1030#														
$CMND	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#	1230#
	1239#	1254#	1300#	1316#	1353#	1433#	1479#	1517#	1521#	1557#	1566#	1574#	1584#	1630#	1692#
	1982#	1989#	2000#	2067#	2168#	2209#	2218#	2228#	2261#	2266#	2300#	2317#	2385#	2390#	
$COMPA	1052#	1075#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#
	1230#	1239#	1254#	1300#	1316#	1345#	1353#	1373#	1413#	1424#	1433#	1479#	1505#	1517#	1521#
	1557#	1566#	1567#	1574#	1584#	1598#	1630#	1691#	1692#	1698#	1708#	1747#	1749#	1808#	1813#
	1820#	1976#	1982#	1989#	1994#	2000#	2063#	2067#	2164#	2168#	2209#	2216#	2218#	2225#	2228#
	2261#	2266#	2292#	2300#	2311#	2317#	2385#	2390#							
$DECR	1075#	1345#	1567#	1598#	1994#										
$DO	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#	1230#
	1239#	1254#	1300#	1316#	1353#	1433#	1479#	1517#	1521#	1557#	1566#	1574#	1584#	1630#	1692#
	1982#	1989#	2000#	2067#	2168#	2209#	2218#	2228#	2261#	2266#	2300#	2317#	2385#	2390#	

$ELSE 1085# 1138# 1149# 1156# 1189# 1215# 1225# 1244# 1694# 2170# 2270# 2304# 2321#
$ENDID 1091# 1347# 1406# 1515# 1569# 1600# 1705# 1711# 1712# 1751# 1755# 1815# 1822# 1835# 1996#
2007# 2065# 2174# 2223# 2232# 2309# 2327#
$ERRMS 1030# 1032# 1033# 1034# 1037# 1038# 1039# 1042# 1044# 1045# 1050# 1052# 1054# 1055# 1056#
1057# 1070# 1071# 1072# 1073# 1075# 1076# 1077# 1078# 1079# 1081# 1082# 1083# 1084# 1085#
1086# 1087# 1089# 1090# 1091# 1093# 1094# 1095# 1097# 1099# 1125# 1129# 1130# 1131# 1135#
1136# 1137# 1138# 1140# 1141# 1142# 1143# 1144# 1145# 1146# 1147# 1148# 1149# 1152# 1153#
1154# 1155# 1156# 1159# 1160# 1163# 1186# 1187# 1188# 1189# 1190# 1191# 1192# 1197# 1199#
1203# 1204# 1208# 1212# 1213# 1214# 1215# 1219# 1220# 1224# 1225# 1226# 1227# 1229# 1230#
1231# 1232# 1233# 1234# 1235# 1236# 1237# 1238# 1239# 1244# 1249# 1250# 1251# 1252# 1253#
1254# 1255# 1256# 1257# 1258# 1262# 1266# 1267# 1294# 1295# 1296# 1298# 1299# 1300# 1303#
1315# 1316# 1318# 1319# 1343# 1345# 1347# 1353# 1356# 1364# 1365# 1366# 1372# 1373# 1376#
1377# 1378# 1379# 1380# 1384# 1385# 1389# 1393# 1394# 1395# 1400# 1405# 1406# 1411# 1413#
1414# 1415# 1416# 1417# 1419# 1421# 1424# 1425# 1426# 1427# 1428# 1430# 1431# 1433# 1436#
1438# 1477# 1478# 1479# 1481# 1482# 1483# 1504# 1505# 1508# 1510# 1511# 1512# 1513# 1514#
1515# 1517# 1519# 1520# 1521# 1523# 1542# 1556# 1557# 1558# 1559# 1560# 1562# 1563# 1566#
1567# 1569# 1570# 1574# 1575# 1576# 1577# 1579# 1583# 1584# 1585# 1586# 1587# 1589# 1590#
1597# 1598# 1600# 1605# 1608# 1610# 1611# 1613# 1614# 1615# 1627# 1630# 1631# 1639# 1640#
1642# 1649# 1691# 1692# 1693# 1694# 1695# 1696# 1697# 1698# 1699# 1701# 1702# 1704# 1705#
1706# 1707# 1708# 1711# 1712# 1713# 1747# 1748# 1749# 1750# 1751# 1752# 1755# 1757# 1802#
1803# 1807# 1808# 1809# 1813# 1814# 1815# 1820# 1821# 1822# 1827# 1828# 1834# 1835# 1872#
1974# 1976# 1977# 1979# 1981# 1982# 1984# 1985# 1988# 1989# 1991# 1994# 1996# 1999# 2000#
2002# 2004# 2007# 2057# 2060# 2061# 2062# 2063# 2064# 2065# 2066# 2067# 2068# 2069# 2070#
2074# 2075# 2079# 2080# 2137# 2138# 2139# 2140# 2141# 2142# 2143# 2144# 2164# 2168# 2170#
2172# 2174# 2209# 2211# 2212# 2213# 2216# 2217# 2218# 2219# 2221# 2222# 2223# 2225# 2226#
2227# 2228# 2229# 2231# 2232# 2240# 2245# 2251# 2256# 2257# 2260# 2261# 2262# 2264# 2265#
2266# 2267# 2270# 2272# 2273# 2274# 2276# 2278# 2279# 2284# 2285# 2286# 2287# 2288# 2289#
2292# 2293# 2294# 2295# 2300# 2304# 2305# 2306# 2308# 2309# 2311# 2312# 2313# 2317# 2321#
2322# 2323# 2325# 2326# 2327# 2328# 2374# 2385# 2386# 2387# 2388# 2389# 2390# 2391# 2392#
$GENBR 1030# 1037# 1042# 1050 1052# 1075# 1076# 1079# 1081# 1085# 1091# 1093# 1125# 1129# 1138#
1141# 1149# 1156# 1186# 1189# 1197# 1208# 1212# 1215# 1219# 1220# 1225# 1226# 1230# 1235#
1239# 1244# 1254# 1300# 1316# 1319# 1345# 1347# 1353# 1373# 1406# 1413# 1424# 1433# 1479#
1483# 1505# 1515# 1517# 1521# 1557# 1566# 1567# 1569# 1574# 1584# 1598# 1600# 1630# 1631#
1691# 1692# 1694# 1698# 1705# 1708# 1711# 1712# 1747# 1749# 1808# 1813# 1815# 1820# 1822#
1835# 1976# 1982# 1989# 1994# 2000# 2063# 2065# 2067# 2070# 2079# 2164# 2168# 2170# 2174#
2209# 2216# 2218# 2223# 2225# 2228# 2232# 2261# 2265# 2266# 2272# 2292# 2300# 2304# 2309#
2311# 2317# 2321# 2327# 2385# 2390#
$GENTA 1030# 1032# 1037# 1042# 1050# 1052# 1054# 1075# 1076# 1079# 1081# 1085# 1087# 1091# 1093#
1095# 1125# 1129# 1138# 1141# 1149# 1154# 1155# 1156# 1163# 1186# 1189# 1192# 1197# 1199#
1208# 1212# 1215# 1219# 1220# 1225# 1226# 1230# 1234# 1235# 1236# 1237# 1238# 1239# 1244#
1254# 1256# 1257# 1258# 1300# 1303# 1316# 1319# 1345# 1347# 1353# 1356# 1373# 1406# 1421#
1431# 1433# 1436# 1479# 1483# 1505# 1515# 1517# 1520# 1521# 1523# 1557# 1563# 1566# 1567#
1569# 1570# 1574# 1579# 1584# 1590# 1598# 1600# 1630# 1631# 1691# 1692# 1694# 1696# 1698#
1705# 1708# 1711# 1712# 1747# 1749# 1751# 1755# 1808# 1813# 1815# 1820# 1822# 1835# 1976#
1982# 1984# 1989# 1991# 1994# 1996# 2000# 2002# 2007# 2059# 2063# 2065# 2067# 2070# 2164#
2168# 2170# 2172# 2174# 2209# 2211# 2216# 2218# 2222# 2223# 2225# 2228# 2231# 2232# 2258#
2261# 2264# 2265# 2266# 2272# 2274# 2292# 2300# 2304# 2308# 2309# 2311# 2317# 2321# 2325#
2327# 2385# 2388# 2390# 2392#
$IDCHK 1075# 1345# 1373# 1505# 1567# 1598# 1691# 1698# 1708# 1747# 1749# 1808# 1813# 1820# 1976#
1994# 2063# 2164# 2216# 2225# 2292# 2311#
$IDFIX 1075# 1345# 1373# 1505# 1567# 1598# 1691# 1698# 1708# 1747# 1749# 1808# 1813# 1820# 1976#
1994# 2063# 2164# 2216# 2225# 2292# 2311#
$IF 1052# 1081# 1093# 1125# 1129# 1141# 1186# 1197# 1208# 1212# 1219# 1220# 1230# 1239# 1254#
1300# 1353# 1433# 1517# 1521# 1557# 1566# 1574# 1584# 1692# 2000# 2168# 2209# 2218# 2228#
2261# 2266# 2300# 2317# 2385# 2390#
$IFBRJ 1079# 1091# 1235# 1319# 1347# 1406# 1483# 1515# 1569# 1600# 1631# 1705# 1711# 1712# 1751#

	1755#	1815#	1822#	1835#	1984#	1991#	1996#	2007#	2065#	2070#	2174#	2223#	2232#	2309#	2327#
$IFCOD	1030#	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#
	1230#	1239#	1254#	1300#	1316#	1353#	1433#	1479#	1517#	1521#	1557#	1566#	1574#	1584#	1630#
	1692#	1982#	1989#	2000#	2067#	2079#	2168#	2209#	2218#	2228#	2261#	2265#	2266#	2300#	2317#
	2385#	2390#													
$IFCON	1413#	1424#													
$IFDEF	1030#	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#
	1230#	1239#	1254#	1300#	1316#	1353#	1413	1424	1433#	1479#	1517#	1521#	1557#	1566#	1574#
	1584#	1630#	1692#	1982#	1989#	2000#	2067#	2079#	2168#	2209#	2218#	2228#	2261#	2265#	2266#
	2300#	2317#	2385#	2390#											
$IFOPR	1030#	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#
	1230#	1239#	1254#	1300#	1316#	1353#	1413#	1424#	1433#	1479#	1517#	1521#	1557#	1566#	1574#
	1584#	1630#	1692#	1982#	1989#	2000#	2067#	2079#	2168#	2209#	2218#	2228#	2261#	2265#	2266#
	2300#	2317#	2385#	2390#											
$INCR	1373#	1505#	1691#	1698#	1708#	1747#	1749#	1808#	1813#	1820#	1976#	2063#	2164#	2216#	2225#
	2292#	2311#													
$INHRJ	1030#	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#
	1230#	1239#	1254#	1300#	1316#	1353#	1413#	1424#	1433#	1479#	1517#	1521#	1557#	1566#	1574#
	1584#	1630#	1692#	1982#	1989#	2000#	2067#	2079#	2168#	2209#	2218#	2228#	2261#	2265#	2266#
	2300#	2317#	2385#	2390#											
$JLPCN	1075#	1345#	1373#	1505#	1567#	1598#	1691#	1698#	1708#	1747#	1749#	1808#	1813#	1820#	1976#
	1994#	2063#	2164#	2216#	2225#	2292#	2311#								
$JUMP	1747#	1749#	1976#	1994#											
$LET	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1056#	1057#	1070#	1071#	1072#	1073#	1077#	1078#
	1082#	1083#	1084#	1086#	1089#	1090#	1094#	1097#	1099#	1130#	1131#	1135#	1136#	1137#	1140#
	1142#	1143#	1144#	1145#	1146#	1147#	1148#	1152#	1153#	1159#	1160#	1187#	1188#	1190#	1191#
	1203#	1204#	1213#	1214#	1224#	1227#	1229#	1231#	1232#	1233#	1249#	1250#	1251#	1252#	1253#
	1255#	1262#	1266#	1267#	1294#	1295#	1296#	1298#	1299#	1315#	1318#	1343#	1364#	1365#	1366#
	1372#	1376#	1377#	1378#	1379#	1380#	1384#	1385#	1389#	1393#	1394#	1395#	1400#	1405#	1411#
	1414#	1415#	1416#	1417#	1419#	1425#	1426#	1427#	1428#	1430#	1438#	1477#	1478#	1481#	1482#
	1504#	1508#	1510#	1511#	1512#	1513#	1514#	1519#	1542#	1556#	1558#	1559#	1560#	1562#	1575#
	1576#	1577#	1583#	1585#	1586#	1587#	1589#	1597#	1605#	1608#	1610#	1611#	1613#	1614#	1615#
	1627#	1639#	1640#	1642#	1649#	1693#	1695#	1697#	1699#	1701#	1702#	1704#	1706#	1707#	1713#
	1748#	1750#	1752#	1757#	1802#	1803#	1807#	1809#	1814#	1821#	1827#	1828#	1834#	1872#	1974#
	1977#	1979#	1981#	1985#	1988#	1999#	2004#	2057#	2060#	2061#	2062#	2064#	2066#	2068#	2069#
	2074#	2075#	2080#	2137#	2138#	2139#	2140#	2141#	2142#	2143#	2144#	2212#	2213#	2217#	2219#
	2221#	2226#	2227#	2229#	2240#	2245#	2251#	2256#	2257#	2260#	2262#	2267#	2270#	2273#	2276#
	2278#	2279#	2284#	2285#	2286#	2287#	2288#	2289#	2293#	2294#	2295#	2305#	2306#	2312#	2313#
	2322#	2323#	2326#	2328#	2374#	2386#	2387#	2389#	2391#						
$LPCNT	1075#	1345#	1373#	1505#	1567#	1598#	1691#	1698#	1708#	1747#	1749#	1808#	1813#	1820#	1976#
	1994#	2063#	2164#	2216#	2225#	2292#	2311#								
$MCHIG	661#														
$MCLOW	661#														
$OPADD	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPAND	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#

	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPCD2	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPDEF	1030#	1033#	1034#	1037#	1038#	1039#	1042#	1044#	1045#	1050#	1052#	1055#	1056#	1057#	1069#
	1070#	1071#	1072#	1073#	1075#	1076#	1077#	1078#	1079#	1081#	1082#	1083#	1084#	1085#	1086#
	1089#	1090#	1091#	1093#	1094#	1097#	1098#	1099#	1125#	1129#	1130#	1131#	1135#	1136#	1137#
	1138#	1140#	1141#	1142#	1143#	1144#	1145#	1146#	1147#	1148#	1149#	1152#	1153#	1156#	1159#
	1160#	1164#	1181#	1186#	1187#	1188#	1189#	1190#	1191#	1197#	1198#	1203#	1204#	1208#	1212#
	1213#	1214#	1215#	1219#	1220#	1224#	1225#	1226#	1227#	1229#	1230#	1231#	1232#	1233#	1235#
	1239#	1244#	1249#	1250#	1251#	1252#	1253#	1254#	1255#	1262#	1266#	1267#	1270#	1294#	1295#
	1296#	1298#	1299#	1300#	1315#	1316#	1318#	1319#	1343#	1345#	1347#	1353#	1364#	1365#	1366#
	1372#	1373#	1376#	1377#	1378#	1379#	1380#	1384#	1385#	1389#	1393#	1394#	1395#	1400#	1405#
	1406#	1411#	1413#	1414#	1415#	1416#	1417#	1419#	1424#	1425#	1426#	1427#	1428#	1430#	1433#
	1438#	1476#	1477#	1478#	1479#	1481#	1482#	1483#	1484#	1504#	1505#	1508#	1510#	1511#	1512#
	1513#	1514#	1515#	1517#	1519#	1521#	1542#	1556#	1557#	1558#	1559#	1560#	1562#	1566#	1567#
	1569#	1574#	1575#	1576#	1577#	1583#	1584#	1585#	1586#	1587#	1589#	1597#	1598#	1600#	1605#
	1608#	1610#	1611#	1613#	1614#	1615#	1627#	1630#	1631#	1639#	1640#	1642#	1649#	1691#	1692#
	1693#	1694#	1695#	1697#	1698#	1699#	1701#	1702#	1704#	1705#	1706#	1707#	1708#	1711#	1712#
	1713#	1747#	1748#	1749#	1750#	1751#	1752#	1755#	1757#	1802#	1803#	1807#	1808#	1809#	1813#
	1814#	1815#	1820#	1821#	1822#	1827#	1828#	1834#	1835#	1872#	1974#	1976#	1977#	1979#	1981#
	1982#	1984#	1985#	1988#	1989#	1991#	1994#	1996#	1999#	2000#	2004#	2007#	2057#	2060#	2061#
	2062#	2063#	2064#	2065#	2066#	2067#	2068#	2069#	2070#	2074#	2075#	2079#	2080#	2137#	2138#
	2139#	2140#	2141#	2142#	2143#	2144#	2164#	2168#	2170#	2174#	2209#	2212#	2213#	2216#	2217#
	2218#	2219#	2221#	2223#	2225#	2226#	2227#	2228#	2229#	2232#	2240#	2245#	2251#	2256#	2257#
	2260#	2261#	2262#	2265#	2266#	2267#	2270#	2271#	2272#	2273#	2276#	2278#	2279#	2284#	2285#
	2286#	2287#	2288#	2289#	2292#	2293#	2294#	2295#	2300#	2304#	2305#	2306#	2309#	2311#	2312#
	2313#	2317#	2321#	2322#	2323#	2326#	2327#	2328#	2374#	2385#	2386#	2387#	2389#	2390#	2391#
$OPEQU	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPEXO	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1808#	1813#	1820#	1834#	1974#
	1977#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#	2219#	2225#	2226#	2229#
	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#	2387#	2389#		
$OPIDB	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#

	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPNAN	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPNOR	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPNOT	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPOR	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPRID	1747#	1749#	1976#	1994#											
$OPROT	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPRO	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1056#	1057#	1070#	1071#	1072#	1073#	1075#	1077#
	1078#	1082#	1083#	1084#	1086#	1089#	1090#	1094#	1097#	1099#	1130#	1131#	1135#	1136#	1137#
	1140#	1142#	1143#	1144#	1145#	1146#	1147#	1148#	1152#	1153#	1159#	1160#	1187#	1188#	1190#
	1191#	1203#	1204#	1213#	1214#	1224#	1227#	1229#	1231#	1232#	1233#	1249#	1250#	1251#	1252#
	1253#	1255#	1262#	1266#	1267#	1294#	1295#	1296#	1298#	1299#	1315#	1318#	1343#	1345#	1364#
	1365#	1366#	1372#	1373#	1376#	1377#	1378#	1379#	1380#	1384#	1385#	1389#	1393#	1394#	1395#
	1400#	1405#	1411#	1414#	1415#	1416#	1417#	1419#	1425#	1426#	1427#	1428#	1430#	1438#	1477#
	1478#	1481#	1482#	1504#	1505#	1508#	1510#	1511#	1512#	1513#	1514#	1519#	1542#	1556#	1558#
	1559#	1560#	1562#	1567#	1575#	1576#	1577#	1583#	1585#	1586#	1587#	1589#	1597#	1598#	1605#
	1608#	1610#	1611#	1613#	1614#	1615#	1627#	1639#	1640#	1642#	1649#	1691#	1693#	1695#	1697#
	1698#	1699#	1701#	1702#	1704#	1706#	1707#	1708#	1713#	1747#	1748#	1749#	1750#	1752#	1757#
	1802#	1803#	1807#	1808#	1809#	1813#	1814#	1820#	1821#	1827#	1828#	1834#	1872#	1974#	1976#
	1977#	1979#	1981#	1985#	1988#	1994#	1999#	2004#	2057#	2060#	2061#	2062#	2063#	2064#	2066#

	2068#	2069#	2074#	2075#	2080#	2137#	2138#	2139#	2140#	2141#	2142#	2143#	2144#	2164#	2212#
	2213#	2216#	2217#	2219#	2221#	2225#	2226#	2227#	2229#	2240#	2245#	2251#	2256#	2257#	2260#
	2262#	2267#	2270#	2273#	2276#	2278#	2279#	2284#	2285#	2286#	2287#	2288#	2289#	2292#	2293#
	2294#	2295#	2305#	2306#	2311#	2312#	2313#	2322#	2323#	2326#	2328#	2374#	2386#	2387#	2389#
	2391#														
$OPR1	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1056#	1057#	1070#	1071#	1072#	1073#	1077#	1078#
	1082#	1083#	1084#	1086#	1089#	1090#	1094#	1097#	1099#	1130#	1131#	1135#	1136#	1137#	1140#
	1142#	1143#	1144#	1145#	1146#	1147#	1148#	1152#	1153#	1159#	1160#	1187#	1188#	1190#	1191#
	1203#	1204#	1213#	1214#	1224#	1227#	1229#	1231#	1232#	1233#	1249#	1250#	1251#	1252#	1253#
	1255#	1262#	1266#	1267#	1294#	1295#	1296#	1298#	1299#	1315#	1318#	1343#	1364#	1365#	1366#
	1372#	1376#	1377#	1378#	1379#	1380#	1384#	1385#	1389#	1393#	1394#	1395#	1400#	1405#	1411#
	1414#	1415#	1416#	1417#	1419#	1425#	1426#	1427#	1428#	1430#	1438#	1477#	1478#	1481#	1482#
	1504#	1508#	1510#	1511#	1512#	1513#	1514#	1519#	1542#	1556#	1558#	1559#	1560#	1562#	1575#
	1576#	1577#	1583#	1585#	1586#	1587#	1589#	1597#	1605#	1608#	1610#	1611#	1613#	1614#	1615#
	1627#	1639#	1640#	1642#	1649#	1693#	1695#	1697#	1699#	1701#	1702#	1704#	1706#	1707#	1713#
	1748#	1750#	1752#	1757#	1802#	1803#	1807#	1809#	181 #	1821#	1827#	1828#	1834#	1872#	1974#
	1977#	1979#	1981#	1985#	1988#	1999#	2004#	2057#	2060#	2061#	2062#	2064#	2066#	2068#	2069#
	2074#	2075#	2080#	2137#	2138#	2139#	2140#	2141#	2142#	2143#	2144#	2212#	2213#	2217#	2219#
	2221#	2226#	2227#	2229#	2240#	2245#	2251#	2256#	2257#	2260#	2262#	2267#	2270#	2273#	2276#
	2278#	2279#	2284#	2285#	2286#	2287#	2288#	2289#	2293#	2294#	2295#	2305#	2306#	2312#	2313#
	2322#	2323#	2326#	2328#	2374#	2386#	2387#	2389#	2391#						
$OPR2	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1056#	1057#	1070#	1071#	1072#	1073#	1075#	1077#
	1078#	1082#	1083#	1084#	1086#	1089#	1090#	1094#	1097#	1099#	1130#	1131#	1135#	1136#	1137#
	1140#	1142#	1143#	1144#	1145#	1146#	1147#	1148#	1152#	1153#	1159#	1160#	1187#	1188#	1190#
	1191#	1203#	1204#	1213#	1214#	1224#	1227#	1229#	1231#	1232#	1233#	1249#	1250#	1251#	1252#
	1253#	1255#	1262#	1266#	1267#	1294#	1295#	1296#	1298#	1299#	1315#	1318#	1343#	1345#	1364#
	1365#	1366#	1372#	1373#	1376#	1377#	1378#	1379#	1380#	1384#	1385#	1389#	1393#	1394#	1395#
	1400#	1405#	1411#	1414#	1415#	1416#	1417#	1419#	1425#	1426#	1427#	1428#	1430#	1438#	1477#
	1478#	1481#	1482#	1504#	1505#	1508#	1510#	1511#	1512#	1513#	1514#	1519#	1542#	1556#	1558#
	1559#	1560#	1562#	1567#	1575#	1576#	1577#	1583#	1585#	1586#	1587#	1589#	1597#	1598#	1605#
	1608#	1610#	1611#	1613#	1614#	1615#	1627#	1639#	1640#	1642#	1649#	1691#	1693#	1695#	1697#
	1698#	1699#	1701#	1702#	1704#	1706#	1707#	1708#	1713#	1748#	1750#	1752#	1757#	1802#	1803#
	1807#	1808#	1809#	1813#	1814#	1820#	1821#	1827#	1828#	1834#	1872#	1974#	1977#	1979#	1981#
	1985#	1988#	1999#	2004#	2057#	2060#	2061#	2062#	2063#	2064#	2066#	2068#	2069#	2074#	2075#
	2080#	2137#	2138#	2139#	2140#	2141#	2142#	2143#	2144#	2164#	2212#	2213#	2216#	2217#	2219#
	2221#	2225#	2226#	2227#	2229#	2240#	2245#	2251#	2256#	2257#	2260#	2262#	2267#	2270#	2273#
	2276#	2278#	2279#	2284#	2285#	2286#	2287#	2288#	2289#	2292#	2293#	2294#	2295#	2305#	2306#
	2311#	2312#	2313#	2322#	2323#	2326#	2328#	2374#	2386#	2387#	2389#	2391#			
$OPR2A	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1808#	1813#	1820#	1834#	1974#
	1977#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#	2219#	2225#	2226#	2229#
	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#	2387#	2389#		
$OPR2B	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1808#	1813#	1820#	1834#	1974#
	1977#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#	2219#	2225#	2226#	2229#
	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#	2387#	2389#		
$OPSHF	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#

	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPSRE	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1808#	1813#	1820#	1834#	1974#
	1977#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#	2219#	2225#	2226#	2229#
	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#	2387#	2389#		
$OPSUB	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OPXOR	1033#	1034#	1038#	1039#	1044#	1045#	1055#	1075#	1077#	1078#	1082#	1084#	1089#	1099#	1131#
	1135#	1136#	1137#	1140#	1145#	1146#	1147#	1152#	1159#	1203#	1214#	1229#	1232#	1233#	1266#
	1267#	1294#	1299#	1318#	1345#	1372#	1373#	1376#	1385#	1393#	1394#	1400#	1405#	1481#	1482#
	1504#	1505#	1508#	1510#	1542#	1556#	1558#	1559#	1567#	1575#	1577#	1585#	1586#	1597#	1598#
	1605#	1608#	1610#	1614#	1639#	1691#	1698#	1701#	1704#	1708#	1747#	1749#	1808#	1813#	1820#
	1834#	1974#	1976#	1977#	1994#	1999#	2061#	2063#	2069#	2074#	2075#	2164#	2213#	2216#	2217#
	2219#	2225#	2226#	2229#	2245#	2260#	2270#	2278#	2286#	2292#	2293#	2305#	2311#	2312#	2322#
	2387#	2389#													
$OR	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#	1230#
	1239#	1254#	1300#	1316#	1353#	1433#	1479#	1517#	1521#	1557#	1566#	1574#	1584#	1630#	1692#
	1982#	1989#	2000#	2067#	2168#	2209#	2218#	2228#	2261#	2266#	2300#	2317#	2385#	2390#	
$RANGE	1030#	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#
	1230#	1239#	1254#	1300#	1316#	1353#	1413#	1424#	1433#	1479#	1517#	1521#	1557#	1566#	1574#
	1584#	1630#	1692#	1982#	1989#	2000#	2067#	2079#	2168#	2209#	2218#	2228#	2261#	2265#	2266#
	2300#	2317#	2385#	2390#											
$SUBON	1050#	1054#	1075#	1079#	1085#	1087#	1091#	1095#	1138#	1149#	1154#	1155#	1156#	1163#	1189#
	1192#	1199#	1215#	1225#	1234#	1235#	1236#	1237#	1238#	1244#	1256#	1257#	1258#	1303#	1319#
	1345#	1347#	1356#	1373#	1406#	1421#	1431#	1436#	1483#	1505#	1515#	1520#	1523#	1563#	1567#
	1569#	1570#	1579#	1590#	1598#	1600#	1631#	1691#	1694#	1696#	1698#	1705#	1708#	1711#	1712#
	1751#	1755#	1808#	1813#	1815#	1820#	1822#	1835#	1984#	1991#	1996#	2002#	2007#	2063#	2065#
	2070#	2079#	2164#	2170#	2172#	2174#	2211#	2216#	2222#	2223#	2225#	2231#	2232#	2264#	2265#
	2272#	2274#	2292#	2304#	2308#	2309#	2311#	2321#	2325#	2327#	2388#	2392#			
$THEN	1052#	1076#	1081#	1093#	1125#	1129#	1141#	1186#	1197#	1208#	1212#	1219#	1220#	1226#	1230#
	1239#	1254#	1300#	1316#	1353#	1433#	1479#	1517#	1521#	1557#	1566#	1574#	1584#	1630#	1692#
	1982#	1989#	2000#	2067#	2168#	2209#	2218#	2228#	2261#	2266#	2300#	2317#	2385#	2390#	
$UNTL2	2079#	2265#													
$UNTL3	2079#	2265#													
$WHILE	1076#	1226#	1316#	1479#	1630#	1982#	1989#	2067#							
$$BASE	1030#														
$$ENDS	1050#														
$$GEN	1030#	1032#	1037#	1042#	1050#	1054#	1075#	1076#	1079#	1081#	1085#	1087#	1091#	1095#	1138#
	1149#	1154#	1155#	1156#	1163#	1189#	1192#	1199#	1215#	1225#	1226#	1234#	1235#	1236#	1237#
	1238#	1244#	1256#	1257#	1258#	1303#	1316#	1319#	1345#	1347#	1356#	1373#	1406#	1421#	1431#
	1436#	1479#	1483#	1505#	1515#	1520#	1523#	1563#	1567#	1569#	1570#	1579#	1590#	1598#	1600#
	1630#	1631#	1691#	1694#	1696#	1698#	1705#	1708#	1711#	1712#	1747#	1749#	1751#	1755#	1808#
	1813#	1815#	1820#	1822#	1835#	1976#	1982#	1984#	1989#	1991#	1994#	1996#	2002#	2007#	2059#
	2063#	2065#	2067#	2070#	2164#	2170#	2172#	2174#	2211#	2216#	2222#	2223#	2225#	2231#	2232#

```
          2258#  2261#  2264#  2265#  2272#  2274#  2292#  2304#  2308#  2309#  2311#  2321#  2325#  2327#  2388#
          2392#
$$GETS    1032#  1037#  1042#  1050#  1054#  1075#  1079#  1085#  1087#  1091#  1095#  1138#  1149#  1154#  1155#
          1156#  1163#  1189#  1192#  1199#  1215#  1225#  1234#  1235#  1236#  1237#  1238#  1244#  1256#  1257#
          1258#  1303#  1319#  1345#  1347#  1356#  1373#  1406#  1421#  1431#  1436#  1483#  1505#  1515#  1520#
          1523#  1563#  1567#  1569#  1570#  1579#  1590#  1598#  1600#  1631#  1691#  1694#  1696#  1698#  1705#
          1708#  1711#  1712#  1751#  1755#  1808#  1813#  1815#  1820#  1822#  1835#  1984#  1991#  1996#  2002#
          2007#  2063#  2065#  2070#  2079#  2164#  2170#  2172#  2174#  2211#  2216#  2222#  2223#  2225#  2231#
          2232#  2264#  2265#  2272#  2274#  2292#  2304#  2308#  2309#  2311#  2321#  2325#  2327#  2388#  2392#
$$GETT    1032#  1037#  1042#  1050#  1085#  1138#  1149#  1156#  1189#  1215#  1225#  1244#  1694#  2170#  2272#
          2304#  2321#
$$JLPC    1747#  1749#  1976#  1994#
$$LPCN    1075#  1345#  1373#  1505#  1567#  1598#  1691#  1698#  1708#  1808#  1813#  1820#  2063#  2164#  2216#
          2225#  2292#  2311#
$$POP     1050#  1054#  1075#  1079#  1085#  1087#  1091#  1095#  1138#  1149#  1154#  1155#  1156#  1163#  1189#
          1192#  1199#  1215#  1225#  1234#  1235#  1236#  1237#  1238#  1244#  1256#  1257#  1258#  1303#  1319#
          1345#  1347#  1356#  1373#  1406#  1421#  1431#  1436#  1483#  1505#  1515#  1520#  1523#  1563#  1567#
          1569#  1570#  1579#  1590#  1598#  1600#  1631#  1691#  1694#  1696#  1698#  1705#  1708#  1711#  1712#
          1751#  1755#  1808#  1813#  1815#  1820#  1822#  1835#  1984#  1991#  1996#  2002#  2007#  2063#  2065#
          2070#  2079#  2164#  2170#  2172#  2174#  2211#  2216#  2222#  2223#  2225#  2231#  2232#  2264#  2265#
          2272#  2274#  2292#  2304#  2308#  2309#  2311#  2321#  2325#  2327#  2388#  2392#
$$PUSH    1030#  1052#  1075#  1076#  1079#  1081#  1085#  1093#  1125#  1129#  1138#  1141#  1149#  1156#  1186#
          1189#  1197#  1208#  1212#  1215#  1219#  1220#  1225#  1226#  1230#  1235#  1239#  1244#  1254#  1300#
          1316#  1319#  1345#  1353#  1373#  1413#  1424#  1433#  1479#  1483#  1505#  1517#  1521#  1557#  1566#
          1567#  1574#  1584#  1598#  1630#  1631#  1691#  1692#  1694#  1698#  1708#  1747#  1749#  1751   1755
          1808#  1813#  1820#  1976#  1982#  1984#  1989#  1991#  1994#  1996   2000#  2007   2059#  2063#  2067#
          2070#  2164#  2168#  2170#  2209#  2216#  2218#  2225#  2228#  2258#  2261#  2266#  2272#  2292#  2300#
          2304#  2311#  2317#  2321#  2385#  2390#
$$SELE    1030#
$$SETS    1030#  1032#  1037#  1042#  1050#  1052#  1075#  1076#  1079#  1081#  1085#  1093#  1125#  1129#  1138#
          1141#  1149#  1156#  1186#  1189#  1197#  1208#  1212#  1215#  1219#  1220#  1225#  1226#  1230#  1235#
          1239#  1244#  1254#  1300#  1316#  1319#  1345#  1353#  1373#  1413#  1424#  1433#  1479#  1483#  1505#
          1517#  1521#  1557#  1566#  1567#  1574#  1584#  1598#  1630#  1631#  1691#  1692#  1694#  1698#  1708#
          1747#  1749#  1751#  1755#  1808#  1813#  1820#  1976#  1982#  1984#  1989#  1991#  1994#  1996#  2000#
          2007#  2059#  2063#  2067#  2070#  2164#  2168#  2170#  2209#  2216#  2218#  2225#  2228#  2258#  2261#
          2266#  2272#  2292#  2300#  2304#  2311#  2317#  2321#  2385#  2390#
$$SETT    1030#  1032#  1037#  1042#  1050#

. ABS.  022370     000     CON   RO    REL   GBL   D

 ERRORS DETECTED:  0

 CZLNAA,CZLNAA.SEQ/CRF/DOC=SPMACJ/ML,SVC33/ML,CZLNAA.P11
 RUN-TIME: 73 70 6 SECONDS
 RUN-TIME RATIO: 176/149=1.1
 CORE USED:  30K  (59 PAGES)

 DOCUMENT PAGES:  108
```