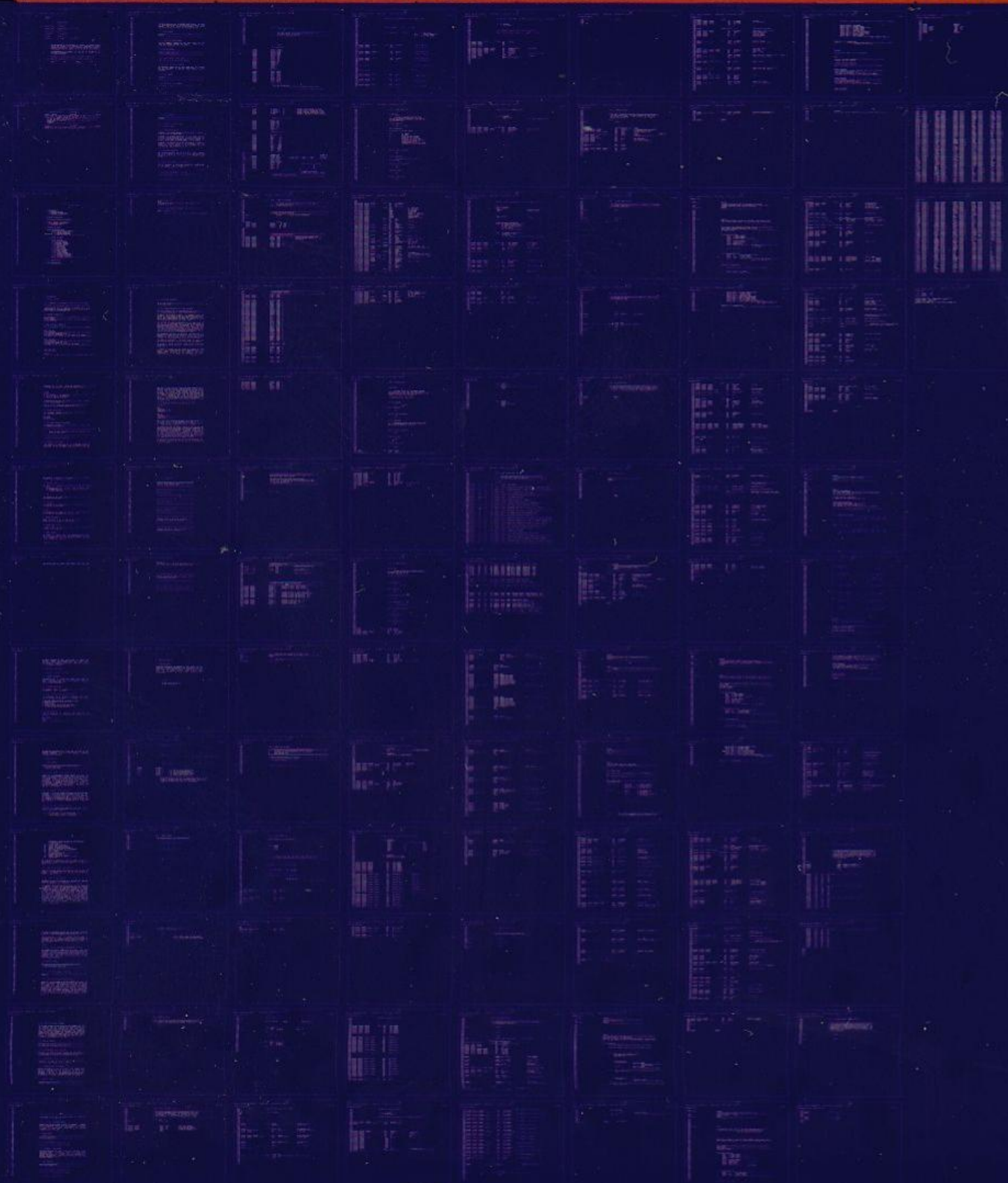


11/21+
KMV11A

KMV11A LINE CTRL DIAG
CNKMBB0

COPYRIGHT (c) 1982-84
AH-T845B-MC
FICHE 01 OF 01

OCT 1984
digital
Made In USA



KMOV11 A LINE CNT DIAGNOSTIC

MACRO M1200 17-APR-84 08:57 PAGE 2

.NLIST TOC
.REM @

IDENTIFICATION

PRODUCT CODE: AC-T844B-MC
PRODUCT NAME: CNKMBB0 KMOV11A LINE CNT DIAG
PRODUCT DATE: APRIL 1984
MAINTAINER: ISS DIAGNOSTICS
AUTHOR: MICHELET GUY
MODIFIED BY: JAKI BERG 9-APR-1984

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

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF
SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS
AFFILIATED COMPANIES.

COPYRIGHT (C) 1982,1984 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL
DEC

PDP
DECUS

UNIBUS
DECTAPE

MASSBUS

PROGRAM DOCUMENT

***** MODIFICATION HISTORY *****

43
44
45
46 REV A: ORIGINAL RELEASE GUY MICHELET 14-JAN-81
47
48 CVKMBA => CNKMBA JAKI PERG 9-APR-84
49 CHANGES WERE MADE TO CVKMBA TO PRODUCE CNDMBA FOR THE FALCON-PLUS PROJECT
50 (SBC-11/21+). CHANGES, MARKED BY ";JB REV A-0", ARE:
51 - SET THE ODT BREAK VECTOR (LOCATION 140) TO THE STARTING ADDRESS OF
52 FALCON'S ODT ROM (170000-OCTAL).
53 - CHANGE PRIORITY FROM LEVEL 7 TO LEVEL 6 TO ALLOW THE BREAK KEY TO
54 INTERRUPT.
55 - LOWERED RATE FROM 72KB TO 64KB IN TEST 06.
56
57 REV B0: JAKI BERG 9-JUL-84
58 LOWERED RATE FROM 72KB TO 64KB IN TEST 07. DEVELOPING ENGINEER IN ANNECY
59 ADVISED THAT TEST 6 AND TEST 7 SHOULD BE RUN AT 64KB.
60

PROGRAM DOCUMENT

TABLE OF CONTENTS

62	
63	
64	
65	
66	
67	
68	
69	
70	
71	1.0 INTRODUCTION
72	1.1 PROGRAM ABSTRACT
73	1.2 HARDWARE INTRODUCTION
74	1.3 DIAGNOSTIC DESCRIPTION
75	
76	2.0 HARDWARE REQUIREMENTS
77	
78	3.0 PRELIMINARY PROGRAM REQUIREMENTS
79	
80	4.0 GENERAL PROGRAM CONSIDERATIONS
81	4.1 DIAGNOSTIC SUPERVISOR
82	4.2 EXECUTION TIME
83	
84	5.0 PROGRAM LOAD MEDIA
85	
86	6.0 OPERATING INSTRUCTIONS
87	6.1 LOADING AND STARTING PROCEDURES
88	6.1.1 LOADING PROCEDURES
89	6.1.2 STARTING PROCEDURES
90	6.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION
91	
92	6.2 INITIAL DIALOGUE
93	
94	6.3 PROGRAM OPTIONS
95	6.3.1 START COMMAND
96	6.3.2 RESTART COMMAND
97	6.3.3 CONTINUE COMMAND
98	6.3.4 PROCEED COMMAND
99	6.3.5 ADD COMMAND
100	6.3.6 DROP COMMAND
101	6.3.7 PRINT COMMAND
102	6.3.8 DISPLAY COMMAND
103	6.3.9 FLAGS COMMAND
104	6.3.10 ZFLAGS COMMAND
105	6.3.11 CONTROL CHARACTERS
106	6.3.12 HARDWARE PARAMETERS
107	6.3.13 SOFTWARE PARAMETERS
108	6.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE
109	
110	7.0 TEST DESCRIPTIONS
111	
112	8.0 ERROR INFORMATION
113	8.1 ERROR REPORTING
114	

PROGRAM DOCUMENT

116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172

1.0 INTRODUCTION

1.1 PROGRAM ABSTRACT

THIS DIAGNOSTIC WAS DESIGNED TO TEST OUT THE KMV11 MODULE
THE PROGRAM WAS IMPLEMENTED USING THE DIAGNOSTIC SUPERVISOR.
THROUGH DIALOGUE WITH THE OPERATOR, THE PROGRAM WILL ALLOW
MODIFICATION OF DEVICE PARAMETERS, SUCH AS UNIBUS ADDRESS,
VECTOR ADDRESS, AND PROCESSOR TYPE.

1.2 HARDWARE INTRODUCTION

THIS DIAGNOSTIC WILL TEST ALL THE HARDWARE PART OF THE KMV11 A
MODULE (M7500).
TO TEST COMPLETELY THIS PART ,EXTERNAL LOOP BACK CONNECTOR
MUST BE INSTALLED.

EXTERNAL LOOP BACK CONNECTOR:

KMV11 A CAN OPERATE EITHER IN RS422 OR RS 423 LEVEL CONVERTERS

RS422 LOOP BACK:

TO TEST COMPLETELY A KMV11 B IN RS422 MODE ,RUN THIS DIAGNOSTIC
WITH LOOP BACK CONNECTOR PLUG :
-USE H3255 TO LOOP DIRECTLY AT THE OUTPUT OF THE MODULE
-USE H3251 PLUG AT THE END OF BC55U MODEM CABLE CONNECTOR ASSY.

RS423 LOOP BACK:

TO TEST COMPLETELY A KMV11-A IN RS423 MODE ,RUN THIS DIAGNOSTIC
WITH LOOP BACK CONNECTOR PLUG :
-USE H3255 TO LOOP AT THE OUTPUT OF THE MODULE
-USE H3251 PLUG AT THE END OF BC55H MODEM CABLE CONNECTOR ASSY.

RS232 LOOP BACK:

SAME AS FOR RS423.

CAUTION:

USE OF H325 LOOP BACK CONNECTOR WILL CAUSE MESSAGES ERROR IN TEST 8.

PROGRAM DOCUMENT

173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225

DIAGNOSTIC WILL TEST KMV11 CLOCKS, LINE INTERRUPTS, TX AND RX FUNCTION
IN INTERNAL AND EXTERNAL LOOP BACK AND MODEM SIGNALS.

CAUTION:

AT THE BEGINNING OF THE DIAGNOSTIC THE OPERATOR WILL ANSWER
BY "YES " OR "NO" AT THE QUESTION:
IS EXTERNAL CONNECTOR PLUGGED?

IF CONNECTOR NO PLUGGED THE DIAGNOSTIC WILL REPORT AN ERROR
AND EXIT CORRESPONDING TEST.

KMV11 A IS FULLY TESTED ONLY WHEN DIAGNOSTIC HAS BEEN RUN
SUCCESSFULLY IN BOTH RS422 AND RS423 LOOP BACK.

2.0 HARDWARE REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE KMV11 A
LINE CONTROLLER STATIC TESTS:

SBC-11/21+
16K MEMORY
CONSOLE TERMINAL

3.0 PRELIMINARY PROGRAM REQUIREMENTS

THE PROCESSOR AND MEMORY SHOULD BE THOROUGHLY TESTED PRIOR
TO RUNNING THIS DIAGNOSTIC.

*
* NOTE: THE KMV11 DIAGNOSTICS NKMDA AND NK MBA SHOULD BE *
* BEFORE RUNNING NK MCA. *
*

4.0 GENERAL PROGRAM CONSIDERATIONS

4.1 DIAGNOSTIC SUPERVISOR

THIS PROGRAM IS COMPATIBLE WITH THE STANDALONE DIAGNOSTIC
SUPERVISOR, AND MUST BE LOADED TO BE CO-RESIDENT WITH THE
SUPERVISOR, OR BE PREVIOUSLY COMBINED WITH THE SUPERVISOR

PROGRAM DOCUMENT

227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283

AND LOADED AS A SINGLE FILE. IN EITHER CASE, THE COMBINED PROGRAM WILL NOT EXCEED 16K OF MEMORY.

4.2 EXECUTION TIME

THE TOTAL TIME REQUIRED TO RUN THE KMV11 LINE CNT DIAGNOSTIC IS ABOUT :

- 160 SECONDS FROM TEST 1 TO TEST 6 (TEST IN INTERNAL LOOP).
- 260 SECONDS FROM TEST 1 TO TEST 8 (COMPLETE TEST, WITH EXTERNAL CONNECTOR).

4.3 XXDP.

THIS PROGRAM MAY BE LOADED UNDER XXDP., AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

4.4 ACT/SLIDE

THIS PROGRAM MAY BE LOADED UNDER ACT OR SLIDE AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

4.5 APT

THIS PROGRAM MAY BE LOADED BY THE APT SYSTEM (INCLUDING APT-RD) AND RUN IN PROGRAM MODE OR SCRIPT MODE.

4.6 MEMORY MANAGEMENT

MEMORY MANAGEMENT IS NOT UTILIZED IN THIS PROGRAM. IF IT IS INSTALLED, IT IS DISABLED BY THE PROGRAM.

4.7 MEMORY PARITY OPTION

IF PARITY MEMORY IS INSTALLED, MEMORY PARITY TRAPS ARE DISABLED BY THE PROGRAM.

4.8 ERROR LOGGING

THE NUMBER OF ERRORS WHICH HAVE OCCURRED ON EACH DEVICE UNDER TEST SINCE THE LAST START OR RESTART COMMAND IS KEPT IN AN ERROR LOG. THIS LOG MAY BE PRINTED BY USING THE "PRINT" COMMAND (SEE SECTION 6.3.8).

5.0 PROGRAM LOAD MEDIA

H1

KMV11A LINE CNT DIAG MACRO M1200 17-APR-84 08:57 PAGE 6-1

SEQ 0007

PROGRAM DOCUMENT

284
285
286

THIS PROGRAM CAN BE LOADED FROM PAPER TAPE USING THE
ABSOLUTE LOADER OR FROM ACT, SLIDE, OR APT SYSTEMS, OR FROM

PROGRAM DOCUMENT

288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342

ANY MEDIA SUPPORTED BY XXDP+. WHEN USING THE PAPER TAPE ABSOLUTE LOADER, THE PROGRAM SHOULD BE LOADED FIRST, FOLLOWED BY THE DIAGNOSTIC SUPERVISOR. WHEN USING XXDP+, THE DIAGNOSTIC SUPERVISOR SHOULD BE LOADED FIRST, FOLLOWED BY THE DIAGNOSTIC PROGRAM.

6.0 OPERATING INSTRUCTIONS

6.1 LOADING AND STARTING PROCEDURES

6.1.1 LOADING PROCEDURES

THIS PROGRAM MAY BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER. IT MAY ALSO BE LOADED FROM ANY XXDP+ LOAD MEDIA. WHEN LOADED UNDER XXDP+, THE DIAGNOSTIC SUPERVISOR WILL BE LOADED AUTOMATICALLY.

6.1.2 STARTING PROCEDURES

THE PROGRAM STARTS AT LOCATION 200. USE STANDARD DEC PROCEDURES TO START THE PROGRAM.

6.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

THE DIAGNOSTIC CAN BE EXECUTED STANDALONE UNDER XXDP+ WITHOUT READING THE REMAINDER OF THIS DOCUMENT, AS FOLLOWS:

- A) LOAD AND START DIAGNOSTIC USING RUN COMMAND
- B) RECEIVE DIAGNOSTIC SUPERVISOR PROMPT (DR>)
- C) ENTER STA<CR>
- D) ANSWER HARDWARE AND SOFTWARE QUESTIONS
- E) GET END OF PASS MESSAGES OR ERROR MESSAGES
- F) TO END EXECUTION, ENTER CONTROL/C

6.2 INITIAL DIALOGUE

AFTER THE PROGRAM AND THE SUPERVISOR ARE LOADED AND THE PROGRAM IS STARTED, THE FOLLOWING IDENTIFICATION IS TYPED:

DRS LOADED
DIAG. RUN-TIME SERVICES
NKMBAO
KMOV11 A LINE CONTROLLER DIAGNOSTIC
DR>

PROGRAM DOCUMENT

344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400

THE OPERATOR THEN PROCEEDS BY TYPING ONE OR MORE OF THE
COMMANDS DESCRIBED IN THE FOLLOWING SECTION 6.3.(FOR MORE
DETAILED INFORMATION, REFER TO THE DIAGNOSTIC SUPERVISOR
FUNCTIONAL SPECIFICATION).

6.3 PROGRAM OPTIONS

6.3.1 START COMMAND

STA(RT)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
<FLAG-LIST>/EOP:<INCR>

6.3.1.1 TESTS SWITCH (/TESTS:<TEST-LIST>)

<TEST-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (1:2 ETC.) OR
RANGES OF DECIMAL NUMBERS (1-5:8-10 ETC.) THAT SPECIFY THE
TESTS TO BE EXECUTED. THE NUMBERS ARE SEPARATED BY COLONS.
THE NUMBERS RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE
DIAGNOSTIC. THEY MAY BE SPECIFIED IN ANY ORDER. TESTS WILL
BE EXECUTED IN NUMERICAL ORDER REGARDLESS OF THE ORDER OF
SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS. ON
THIS AND ALL SWITCHES, THE ANGLE BRACKETS <> ARE PUNCTUATION
USED IN THE DEFINITION ONLY, AND ARE NOT TO BE TYPED BY THE
OPERATOR. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.2 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER
OF PASSES. A PASS IS DEFINED AS THE EXECUTION OF THE FULL
DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED.
THE DEFAULT IS NON-ENDING EXECUTION. IN THIS CASE EXIT FROM
THE PROGRAM IS ACCOMPLISHED EITHER BY TYPING A CONTROL/C OR
BY OCCURRENCE OF AN ERROR WITH THE HALT ON ERROR FLAG BEING
SET. THE EXIT IS A RETURN TO COMMAND MODE. SEE EXAMPLE AT
END OF 6.3.1.5.

6.3.1.3 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS A SEQUENCE OF ELEMENTS OF THE FORM <FLAG>,
<FLAG=1>, OR <FLAG=0>, SEPARATED BY COLONS, WHERE <FLAG> HAS
ONE OF THE FOLLOWING VALUES:

HOE HALT ON ERROR, CAUSING COMMAND MODE TO BE
ENTERED WHEN AN ERROR IS ENCOUNTERED
LOE LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP

PROGRAM DOCUMENT

402
403
404
405
406
407 CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK
408 OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAIN-
409 ING THE ERROR
410 IER INHIBIT ERROR REPORTING
411 IBE INHIBIT BASIC ERROR REPORTS
412 IXE INHIBIT EXTENDED ERROR REPORTS
413 PRI DIRECT ALL MESSAGES TO A LINE PRINTER
414 PNT PRINT NUMBER OF TEST BEING EXECUTED
415 BOE BELL ON ERROR
416 UAM RUN IN UNATTENDED MODE, BYPASSING MANUAL
417 INTERVENTION TESTS
418 ISR INHIBIT STATISTICAL REPORTS
419 IDU INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
420 LOT LOOP ON TEST

421
422 THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0
423 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS
424 SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT
425 END OF 6.3.1.5.
426
427

428 6.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)

429
430 <INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF
431 PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE
432 PRINTED. THE DEFAULT IS AT THE END OF EVERY PASS. SEE
433 EXAMPLE AT END OF 6.3.1.5.
434
435

436 6.3.1.5 EFFECT OF START COMMAND

437
438 THE EFFECT OF THE START COMMAND IS TO INITIATE THE HARDWARE
439 PARAMETER DIALOGUE, THE SOFTWARE PARAMETER DIALOGUE, AND
440 THEN THE DIAGNOSTIC TESTS THEMSELVES.
441

442 THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE QUESTION
443 "N UNITS?" TO WHICH THE OPERATOR REPLIES WITH A DECIMAL
444 NUMBER N FROM 1 TO 16. THE TERM "UNIT" REFERS TO THE DEVICE
445 TO WHICH THIS SERIES OF DIAGNOSTICS IS DEDICATED. FOLLOWING
446 THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES WILL
447 BE BUILT. EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING
448 ALL THE HARDWARE INFORMATION FOR ONE UNIT. THE OPERATOR
449 MUST SUPPLY N (NUMBER OF UNITS) VALUES FOR EACH QUESTION.
450 HE MAY DO THIS BY GIVING ONE ANSWER TO EACH QUESTION (IN
451 WHICH CASE THE SERIES OF QUESTIONS WILL BE POSED N TIMES) OR
452 BY GIVING N VALUES, SEPARATED BY COMMAS, TO EACH QUESTION
453 (SERIES WILL BE POSED ONCE). EACH QUESTION IS FOLLOWED BY
454 THE RESPONSE RADIX (D FOR DECIMAL, B FOR BINARY, O FOR
455 OCTAL, L FOR YES/NO) IN PARENTHESES AND THE DEFAULT VALUE
456 AFTER THE PARENTHESES.

PROGRAM DOCUMENT

458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512

FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE QUESTIONS TO BUILD THE SOFTWARE TABLES, WHICH DEFINE THE MODE (QUICK VFRIFY ETC.) THAT THE DIAGNOSTIC WILL EXECUTE IN.

WHEN THE QUESTION "# UNITS?" IS ANSWERED, MEMORY STORAGE IS ALLOCATED FOR THE P-TABLES, AND IF THERE IS NOT ENOUGH TO ACCOMMODATE THEM THE MESSAGE "TOO MANY UNITS" IS ISSUED. IN THIS CASE THE DIAGNOSTIC MUST BE EXECUTED MORE THAN ONCE TO TEST ALL UNITS.

EXAMPLE:

STA/TESTS:1:2-4:6:8-10/PASS:3/FLAGS:IER:HOE=1:UAM:LOE

THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE, EACH PASS CONSISTING OF TESTS 1,2,3,4,6,8,9, AND 10 EXECUTED AGAINST ALL UNITS. THERE IS NO DIFFERENCE BETWEEN SAYING <FLAG> AND SAYING <FLAG=1>. THE NOTATION <FLAG=0> IS MEANINGFUL ONLY ON A COMMAND OTHER THAN START TO CLEAR A FLAG THAT WAS PREVIOUSLY SET. NOTE THAT ON ALL COMMANDS ONLY THE FIRST THREE LETTERS ARE SCANNED.

6.3.2 RESTART COMMAND

```
*****
RES(TART)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
<FLAG-LIST>/UNITS:<UNIT-LIST>
*****
```

6.3.2.1 TESTS, PASS, AND FLAGS SWITCHES

<TEST-LIST>, <PASS-CNT>, AND <FLAG-LIST> ARE AS IN THE START COMMAND.

6.3.2.2 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (0,1 ETC.) OR RANGES OF DECIMAL NUMBERS (0-5, 8-10 ETC.) THAT SPECIFY THE UNITS TO BE TESTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS MAY RANGE FROM 0 THRU N-1 (N IS THE NUMBER OF UNITS SPECIFIED IN THE PREVIOUS START COMMAND). THE NUMBER INDICATES THE POSITION OF THE P-TABLE AS THE DATA WAS ENTERED DURING THE HARDWARE DIALOGUE. THE UNITS WHICH ARE SELECTED MUST NOT HAVE BEEN DROPPED BY THE DROP COMMAND. SEE THE DISCUSSION OF ADD AND DROP COMMANDS BELOW. DEFAULT IS TO TEST ALL UNITS WHICH HAVE NOT BEEN DROPPED BY A DROP COMMAND.

PROGRAM DOCUMENT

514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567

6.3.2.3 EFFECT OF RESTART COMMAND

THE RESTART COMMAND DIFFERS FROM THE START COMMAND IN THAT THE P-TABLES FROM THE PREVIOUS START COMMAND (THERE MUST HAVE BEEN ONE) ARE USED, INSTEAD OF NEW ONES BEING BUILT. THE UNITS SWITCH GIVES THE ABILITY TO SELECT A SUBSET OF THESE. THE SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED (OPERATOR WILL BE ASKED). THE COMMAND CAN BE USED AFTER COMMAND MODE HAS BEEN REENTERED IN ANY OF THE THREE NORMAL WAYS: A) THE REQUESTED NUMBER OF PASSES HAVE BEEN MADE B) AN ERROR WAS ENCOUNTERED WITH THE HALT ON ERROR FLAG SET C) A CONTROL/C WAS ENTERED BY THE OPERATOR.

6.3.3 CONTINUE COMMAND

CON(TINUE)/PASS:<PASS-CNT/FLAGS:<FLAG-LIST>

6.3.3.1 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS SAME AS IN START COMMAND, BUT THE DEFAULT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART. IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.

6.3.3.2 FLAG SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS SAME AS IN START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

6.3.3.3 EFFECT OF CONTINUE COMMAND

CONTINUE MUST FOLLOW A START OR RESTART, AND COMMAND MODE MUST HAVE BEEN ENTERED DUE TO A HALT ON ERROR OR A CONTROL/C. THE EFFECT OF THE COMMAND IS TO GO TO THE BEGINNING OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT OR CONTROL/C TOOK PLACE. SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED. HARDWARE PARAMETERS MAY NOT BE CHANGED.

6.3.4 PROC'ED COMMAND

PRO(CEED)/FLAGS:<FLAG-LIST>

PROGRAM DOCUMENT

569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623

6.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS AS IN THE START COMMAND, BUT UNSPECIFIED
FLAGS RETAIN THEIR CURRENT VALUE.

6.3.4.2 EFFECT OF PROCEED COMMAND

PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE. COMMAND
MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR. THE EFFECT
OF THE COMMAND IS TO BEGIN EXECUTION AT THE LOCATION
FOLLOWING THE ERROR CALL. NEITHER HARDWARE NOR SOFTWARE
PARAMETERS MAY BE ALTERED.

6.3.5 ADD COMMAND

ADD/UNITS:<UNIT-LIST>

6.3.5.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.5.2 EFFECT OF ADD COMMAND

THE UNITS SPECIFIED ARE ADDED TO THE TEST SEQUENCE. EACH
UNIT MUST HAVE A P-TABLE IN MEMORY DUE TO AN EARLIER
HARDWARE DIALOGUE. THIS COMMAND MUST BE FOLLOWED BY A
RESTART OR CONTINUE. THE UNITS SWITCH MUST BE SPECIFIED.
THE ADD COMMAND IS MEANINGFUL ONLY FOR UNITS THAT WERE
PREVIOUSLY DROPPED.

6.3.6 DROP COMMAND

DRO(P)/UNITS:<UNIT-LIST>

6.3.6.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.6.2 EFFECT OF DROP COMMAND

PROGRAM DOCUMENT

625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679

THE UNITS SPECIFIED WILL BE DROPPED FROM TESTING. THE UNITS
WILL BE RESELECTED ONLY BY THE EXECUTION OF AN ADD OR START
COMMAND. THE UNITS SWITCH MUST BE ENTERED. THIS COMMAND
MUST BE FOLLOWED BY A RESTART OR A CONTINUE COMMAND.

6.3.7 PRINT COMMAND

PRI(NT)

6.3.7.1 EFFECT OF PRINT COMMAND

THE TOTAL NUMBER OF ERRORS FOR EACH UNIT SINCE THE LAST
START OR RESTART COMMAND ARE PRINTED. THE ISR (INHIBIT
STATISTICAL REPORTING) FLAG IS CLEARED.

6.3.8 DISPLAY COMMAND

DIS(PLAY)/UNITS:<UNIT-LIST>

6.3.8.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.8.2 EFFECT OF DISPLAY COMMAND

THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED
OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS
THAT WERE DROPPED BY THE OPERATOR "DROP" COMMAND ARE SO
DESIGNATED.

6.3.9 FLAGS COMMAND

FLA(GS)

6.3.9.1 EFFECT OF FLAGS COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

PROGRAM DOCUMENT

681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737

6.3.10 ZFLAGS COMMAND

ZFL(AGS)

6.3.10.1 EFFECT OF ZFLAGS COMMAND

ALL FLAGS ARE CLEARED.

6.3.11 CONTROL CHARACTERS

A CONTROL C (C) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES A RETURN TO COMMAND MODE.

A CONTROL Z (Z) ENTERED DURING ONE OF THE THREE OPERATOR DIALOGUES- INITIAL DIALOGUE (SEE 6.2), HARDWARE DIALOGUE (SEE 6.3.1.5), OR SOFTWARE DIALOGUE (SEE 6.3.1.5) CAUSES THE DEFAULTS TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.

A CONTROL O (O) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES ALL TELETYPE OUTPUT TO BE SUPPRESSED FOR THE REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER O IS TYPED, WHICH RESTORES NORMAL TELETYPE OUTPUT.

6.3.12 HARDWARE PARAMETERS

THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONSE.

2. MICRO-CPU CSR ADDRESS: (0) 177000?

THIS IS THE ADDRESS AT WHICH THE CSR REGISTERS (SELO) RESIDE ON THE UNIBUS. THE ALLOWABLE RANGE IS 160000-177776 (OCTAL), AND THE DEFAULT IS 177000.

3. MICRO CPU VECTOR ADDRESS: (0) 300?

THE ALLOWABLE RANGE IS 300-770, AND DEFAULT VALUE IS 300

4. MICRO CPU PRIORITY LEVEL: (4) ??

PROGRAM DOCUMENT

738
739
740
741
742
743
744
745
746
747
748
749
750
751

DEFAULT VALUE IS 4

NOTE:

M7500 AND M7501 MODULE MOUNTED WITH DC003 CHIPS CAN ONLY
INTERRUPT ON LEVEL 4

5. IS LOOP BACK CONNECTOR PLUGGED? 0=NO,1=YES (0) 1 ?

DEFAULT VALUE IS 1 (YES)

NOTE :

REFER TO CHAPTER 1.2 FOR LOOP BACK CONNECTOR DESCRIPTION.

PROGRAM DOCUMENT

753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETER QUESTIONS ARE ASKED BY PART 2 OF THE STATIC LOGIC TESTS.

6.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

THE FULL CAPABILITY OF THE HARDWARE DIALOGUE IS REVEALED BY THE FOLLOWING DISCUSSION OF WHAT HAPPENS INTERNALLY.

AS SOON AS THE QUESTION "N UNITS?" IS ANSWERED (WITH THE NUMBER N, SAY) SPACE IN CORE IS ALLOCATED FOR N P-TABLES. ALL OF THE P-TABLES ARE OF THE SAME FORMAT, AND THERE IS A ONE-TO ONE CORRESPONDENCE BETWEEN THE HARDWARE PARAMETER QUESTIONS AND THE SLOTS IN THE P-TABLE FORMAT.

ON THE FIRST TRIP THRU THE QUESTIONS, ALL OF THE SLOTS IN ALL OF THE P-TABLES ARE FILLED. IF THE OPERATOR TYPES IN LESS THAN N EXPLICIT VALUES IN RESPONSE TO A PARTICULAR QUESTION, THESE VALUES ARE PLACED IN THE P-TABLES (ONE VALUE GOING INTO THE PROPER SLOT OF EACH P-TABLE BEGINNING WITH THE FIRST P-TABLE) UNTIL THE STRING OF VALUES IS EXHAUSTED. THE LAST VALUE IN THE STRING BECOMES THE NEW DEFAULT AND IS USED TO FILL THAT SLOT IN THE REMAINING P-TABLES.

ON SUBSEQUENT TRIPS THRU THE QUESTIONS, THE SAME PROCESS IS CARRIED OUT, EXCEPT THAT THE EARLIEST P-TABLE NOT TO HAVE RECEIVED AN EXPLICIT VALUE IN ANY OF ITS SLOTS NOW ASSUMES THE ROLE THAT TABLE NUMBER ONE PLAYED IN THE FIRST TRIP.

THE SERIES OF QUESTIONS IS REISSUED UNTIL AT LEAST ONE QUESTION HAS RECEIVED N EXPLICIT VALUES FROM THE OPERATOR.

IN GIVING A STRING OF VALUES, COMMAS WITHOUT INTERVENING VALUES MAY BE USED TO INDICATE A REPETITION OF THE LAST NAMED VALUE.

A STRING OF VALUES MAY BE GIVEN AS A RANGE (6-10 FOR EXAMPLE). IF THE VALUES REPRESENT PURE NUMERICAL DATA, THIS SAMPLE RANGE TRANSLATES TO THE STRING 6,7,8,9,10 (AN INCREMENT OF 1). IF THE VALUES ARE ADDRESSES, THE SAMPLE RANGE TRANSLATES TO THE STRING 6,8,10 (AN INCREMENT OF 2).

PROGRAM DOCUMENT

805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853

NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO CONSTRUCT A SET OF P-TABLES. ASSUME THAT WE HAVE 16 UNITS, AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE SLOTS IN THE P-TABLE, THREE HARDWARE QUESTIONS IN THE DIALOGUE). LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE THE NUMBER 75 FOR ALL 16 TABLES. LET THE DESIRED VALUE FOR THE SECOND PARAMETER BE EQUAL TO THE UNIT NUMBER (0,1,2,...,15) EXCEPT FOR UNIT 12, WHICH SHOULD RECEIVE THE VALUE 11. LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE THE NUMBER 76 FOR THE FIRST 7 UNITS AND THE NUMBER 77 FOR THE LAST 9 UNITS.

THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

UNITS (0) ? 16

UNIT 1

<QUESTION 1> ? 75

<QUESTION 2> ? 0-6

<QUESTION 3> ? 76

UNIT 21

<QUESTION 1> ?

<QUESTION 2> ? 7-11,,13-15

<QUESTION 3> ? 77

THE FIRST TIME THE SERIES IS ASKED, SLOT ONE RECEIVES A 75 IN ALL 16 TABLES. SLOT TWO RECEIVES THE VALUES 0,1,2,...,6 IN TABLES 0 THRU 6 AND A CONSTANT 6 IN TABLES 7 THRU 15. SLOT THREE RECEIVES A CONSTANT 76 IN ALL 16 TABLES.

THE SECOND TIME THRU THE SERIES, TABLES 16 THRU THE END ARE GOING TO BE AFFECTED (NOTE THAT THIS PIECE OF INFORMATION IS PRINTED OUT FOR THE THE OPERATOR IN THE FORM "UNIT XX" AT THE BEGINNING OF EACH SERIES). QUESTION 1 IS RESPONDED TO BY A <CR>, SO SLOT ONE STAYS AT CONSTANT 75 IN TABLES 7 THRU 15, SINCE NO NEW EXPLICIT VALUES ARE TYPED IN. SLOT TWO GETS THE VALUES 7,8,9,10,11 IN TABLES 7 THRU 11, AND GETS A 11 IN SLOT 12, AND GETS THE VALUES 13,14,15 IN TABLES 13 THRU 15. SLOT THREE GETS THE VALUE 77 IN TABLES 7 THRU 15.

THE DIALOGUE IS TERMINATED WHEN THE SOFTWARE RECOGNIZES THAT 16 EXPLICIT VALUES HAVE BEEN GIVEN FOR AT LEAST ONE QUESTION (NAMELY QUESTION 2).

PROGRAM DOCUMENT

855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904

7.0 TEST DESCRIPTIONS

***** TEST 1 *****
*VERIFY THAT REFERENCED UNIBUS DEVICE REGISTERS
*DOES NOT CAUSE TIME OUT TRAP

***** TEST 2 *****
*
*PROM REVISION TEST
*

***** TEST 3 *****
*
*REAL TIME CLOCK TEST
*

***** TEST 4 *****
*
*BAUD RATE GENERATOR TEST
*

***** TEST 5 *****
*
*TRANSMIT FRAMES AT LOW SPEED IN INTERNAL LOOP
*ON CHANNEL A WITHOUT ANY INTERRUPT
*

***** TEST 6 *****
*
*TRANSMIT AND RECEIVE FRAMES IN INTERNAL LOOP AT
*DIFFERENT SPEED WITH INTERRUPT
*

PROGRAM DOCUMENT

906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931

CAUTION:
TEST NUMBER 7 AND 8 LOOP BACK CONNECTOR MUST BE INSTALLED.
REFER TO CHAPTER 1.2 FOR LOOP BACK DESCRIPTION

***** TEST 7 *****
*
*TRANSMIT AND RECEIVE FRAMES IN EXTERNAL LOOP BACK
*(WITH EXTERNAL LOOP BACK)
*

***** TEST 8 *****
*
*TEST ALL MODEM SIGNAL IN EXTERNAL LOOP BACK
*

PROGRAM DOCUMENT

933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964

8.0 ERROR INFORMATION

8.1 ERROR REPORTING

ERRORS ARE REPORTED BY THE PROGRAM AS THEY OCCUR (IF NOT INHIBITED). THE REPORT CONFORMS TO THE DIAGNOSTIC SUPERVISOR ERROR REPORT FORMAT, AND CONSISTS OF A DESCRIPTION OF THE ERROR, THE TEST NUMBER, SUBTEST NUMBER, PC OF THE ERROR CALL, DEVICE ADDRESS, AND BASIC AND EXTENDED ERROR INFORMATION.

9.0 HISTORY

- DESIGN STARTED ON MAY 82
- REVIEW ON DECEMBER 82

a

PROGRAM DOCUMENT

```

966          .TITLE KMOV11 A LINE CNT DIAGNOSTIC
974          002000          .-2000
975
976
977
978
979
980
981          .MCALL  SVC
982 002000          SVC          ; INITIALIZE SUPERVISOR MACROS
983
984
985
986
987
988 002000          BGNMOD  KMOV11A
989
990
991          000000          $LSTIN= 0
992          000000          $LSTTAG= 0
993          177777          SVCINS= -1      ; LIST INSTRUCTIONS, SHIFTED RIGHT
994          177777          SVCTS= -1      ; LIST TEST TAGS, SHIFTED RIGHT
995          177777          SVCSUB= -1     ; LIST SUBTEST TAGS, SHIFTED RIGHT
996          177777          SVCGBL= -1    ; LIST GLOBAL TAGS, SHIFTED RIGHT
997          177777          SVCTAG= -1    ; LIST OTHER TAGS, SHIFTED RIGHT
998
999          ;      CHANGE THE VALUES OF THE SVC... SYMBOLS TO BE ZERO IF YOU WISH
1000         ;      TO ALIGN THE MACRO CALLS AND THEIR EXPANSIONS.  CHANGE THE
1001         ;      SYMBOLS TO BE MINUS-ONE TO NOT LIST THE EXPANSIONS.  YOU MAY
1002         ;      CHANGE THE SYMBOLS AT ANY POINT IN YOUR PROGRAM.
1003
1004

```

PROGRAM HEADER

```
1006      .SBTTL  PROGRAM HEADER
1007      :
1008      : THE PROGRAM HEADER IS THE INTERFACE BETWEEN
1009      : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
1010      :
1011
1012 002000      POINTER BGNSW,BGNDU,BGNSETUP
1013
1014
1015
1033
1034 002000      HEADER  NKMBA0,A,0,240..0
1035
```

PROGRAM HEADER

```
1047
1048
1049      ;**
1050      ; THIS TABLE IS USED BY THE RUNTIME SERVICES
1051      ; TO PROTECT THE LOAD MEDIA.
1052      ;--
1053      002122      BGNPROT
1054
1055      002122      000000      0      ;OFFSET INTO P-TABLE FOR CSR ADDRESS
1056      002124      177777      -1      ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
1057      002126      177777      -1      ;OFFSET INTO P-TABLE FOR DRIVE NUMBER
1058
1059
1073
1074
1075      002130      ENDPROT
1076
```

DISPATCH 8

DEFAULT HARDWARE P-TABLE

1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106 002152
1107
1117
1118
1119 002154 177000
1120 002156 000300
1121 002160 004000
1122 002162 0C0001
1123 002164

.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 STRUCTURE OF THE RUN-TIME P-TABLE.
;/ AND IS USED AS A " TEMPLATE" FOR BUILDING THE P-TABLE
;/;;;

.ENABL AMA
BGNHW DFPTBL

.WORD	177000	;KMV11,CSRS ADDRESS
.WORD	300	;KMV11, VECTOR ADDRESS
.WORD	4000	;INTERRUPT PRIORITY LEVEL
.WORD	1	;LOOP BACK CONNECTOR?
ENDHW		

DEFAULT HARDWARE P-TABLE

1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1152
1153
1168
1169 002164

.SBTTL GLOBAL EQUATES SECTION

```

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

```

EQUALS

```

;
; BIT DEFINITIONS
;

```

100000	BIT15== 100000
040000	BIT14== 40000
020000	BIT13== 20000
010000	BIT12== 10000
004000	BIT11== 4000
002000	BIT10== 2000
001000	BIT09== 1000
000400	BIT08== 400
000200	BIT07== 200
000100	BIT06== 100
000040	BIT05== 40
000020	BIT04== 20
000010	BIT03== 10
000004	BIT02== 4
000002	BIT01== 2
000001	BIT00== 1
001000	BIT9== BIT09
000400	BIT8== BIT08
000200	BIT7== BIT07
000100	BIT6== BIT06
000040	BIT5== BIT05
000020	BIT4== BIT04
000010	BIT3== BIT03
000004	BIT2== BIT02
000002	BIT1== BIT01
000001	BIT0== BIT00

```

;
; EVENT FLAG DEFINITIONS
; EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
;
; BIT POSITION IN SECOND STATUS WORD

```

GLJBL EQUATES SECTION

```

000040 EF.START== 32. ; (100000) START COMMAND WAS ISSUED
000037 EF.RESTART== 31. ; (040000) RESTART COMMAND WAS ISSUED
000036 EF.CONTINUE== 30. ; (020000) CONTINUE COMMAND WAS ISSUED
000035 EF.NEW== 29. ; (010000) A NEW PASS HAS BEEN STARTED
000034 EF.PWR== 28. ; (004000) A POWER-FAIL/POWER-UP OCCURRED

```

; PRIORITY LEVEL DEFINITIONS

```

000340 PRI07== 340
000300 PRI06== 300
000240 PRI05== 240
000200 PRI04== 200
000140 PRI03== 140
000100 PRI02== 100
000040 PRI01== 40
000000 PRI00== 0

```

; OPERATOR FLAG BITS

```

000004 EVL== 4
000010 LOT== 10
000020 ADR== 20
000040 IDU== 40
000100 ISR== 100
000200 UAM== 200
000400 BOE== 400
001000 PNT== 1000
002000 PRI== 2000
004000 IXE== 4000
010000 IBE== 10000
020000 IER== 20000
040000 LOE== 40000
100000 HOE== 100000

```

; MAXPRI==340

MAXPRI==300

MAINT0==54000

MAINT1==44000

MCLR==40000

DATA1== 052525

DATA2== 125252

KB1.2== 5780.

KB64== 108.

KB56== 124.

KB68== 102.

KB72== 97.

;JB REV A-0

;JB REV A 0

;MASTER CLEAR = 1,MODE = 1 ,MAINT 1 = 1 ,T11=HOLD

;MASTER CLEAR = 1,MODE = 0 ,MAINT 1 = 0 ,T11=NOT HOLD

;OCTAL VALUE OF 1.2 KBAUDS

; " " " 64 "

; " " " 56 "

; " " " 68 "

; " " " 72 "

;DIVIDER CALCULATION

;DECIMAL VAUE = 6912:YYY KBAUDS

;*****

; PROGRAM EVENT FLAG DEFINITIONS

;*****

1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192

GLOBAL DATA SECTION

1194
1195
1196
1197
1198
1199
1200
1206
1207
1208
1209
1210
1211
1212
1213
1214
1227
1228

1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244

002164

002220

002220 000000
002222 000000
002224 000000
002226 000000

002230 000000
002232 000005
002234 000000
002236 000000
002240 000015
002242 000000
002244 000000
002246 000000
002250 000000

.SBTTL GLOBAL DATA SECTION

////////////////////////////////////
// THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
// IN MORE THAN ONE TEST.
////////////////////////////////////

; * STORAGE FOR DEVICE REGISTERS
; *****
DESCRIPT <KMV11A LINE CNT DIAGNOSTIC>

ERRTBL
ERRTYP: .WORD 0
ERRNBR: .WORD 0
ERRMSG: .WORD 0
ERRBLK: .WORD 0

; * PROGRAM CONTROL PARAMETERS
; *****
LOCK: .WORD 0 ;ADDRESS FOR LOCK CURRENT DATA
MAXERR: .WORD 5 ;MAX ERROR BEFORE DROPPING THE UNIT
ERRCNT: .WORD 0 ;ERROR COUNT
L\$SW: .WORD 0
L\$UIT: .WORD 15 ;MAX LINE UNIT
LOGDEV: .WORD 0
SAVPC: .WORD 0
PSTACK: .WORD 0
FTIME: .WORD 0

GLOBAL DATA SECTION

```

1246 ;*****
1247 ;* MISCELLANEOUS STORAGE
1248 ;*****
1249 002252 000000 SAVE4: .WORD 0
1250 002254 000000 SAVE6: .WORD 0
1251 002256 000000 FLAG: .WORD 0
1252
1253 002260 000000 DELCT1: .WORD 0
1254 002262 000000 DELCT2: .WORD 0
1255 002264 000000 GOOD: .WORD 0
1256 002266 000000 GOOD0: .WORD 0
1257 002270 000000 GOOD1: .WORD 0
1258 002272 000000 GOOD2: .WORD 0
1259 002274 000000 GOOD4: .WORD 0
1260 002276 000000 GOOD6: .WORD 0
1261 002300 000000 GOOD10: .WORD 0
1262 002302 000000 GOOD12: .WORD 0
1263 002304 000000 GOOD14: .WORD 0
1264 002306 000000 GOOD16: .WORD 0
1265 002310 000000 SEL0: .WORD 0
1266 002312 000000 SEL1: .WORD 0
1267 002314 000000 SEL2: .WORD 0
1268 002316 000000 SEL4: .WORD 0
1269 002320 000000 SEL6: .WORD 0
1270 002322 000000 SEL10: .WORD 0
1271 002324 000000 SEL12: .WORD 0
1272 002326 000000 SEL14: .WORD 0
1273 002330 000000 SEL16: .WORD 0
1274 002332 000000 BSEL1: .WORD 0
1275 002334 000000 RANST: .WORD 0
1276 002336 000000 RANSEL: .WORD 0
1277 002340 000000 RANMTA: .WORD 0
1278 002342 000000 RANDN: .WORD 0
1279 002344 000000 SAVPC1: .WORD 0
1280 002346 000000 SAVSTA: .WORD 0
1281 002350 000000 COUNT: .WORD 0
1282 002352 000000 NUMBER: .WORD 0
1283 002354 000000 ADDR: .WORD 0
1284 002356 000000 GDDAT: .WORD 0
1285 002360 000000 BDDAT: .WORD 0
1286
1287 002362 TTABLE: .BLKW 2000
1288 006362 RTABLE: .BLKW 2000
1289
1290 012362 000000 EXADDR: .WORD 0
1291 012364 000000 INTFLG: .WORD 0
1292 012366 000000 BAD: .WORD 0
1293 012370 000000 BSEL0: .WORD 0
1294 012372 000000 DATA: .WORD 0
1295 012374 000000 VECT: .WORD 0
1296
1297
1298 012376 000000 KIND: .WORD 0
1299 012400 000000 CHANEL: .WORD 0
1300
1301 012402 000000 TXDATA: .WORD 0
1302 012404 000000 RXDATA: .WORD 0

```

;=0 IF KMV11A ,=1 IF KMV11B

F3

KMV11 A LINE CNT DIAGNOSTIC

MACRO M1200 17-APR-84 08:57 PAGE 27-1

SEQ 0031

GLOBAL DATA SECTION

1303	012406	000000	TSPEED:	.WORD	0
1304	012410	000000	LENGTH:	.WORD	0
1305	012412	000000	NUB:	.WORD	0
1306	012414	000000	RXCNT:	.WORD	0
1307	012416	000000	STAERR:	.WORD	0
1308	012420	000000	WRDCNT:	.WORD	0
1309	012422	000000	UNIT:	.WORD	0

GLOBAL DATA SECTION

```
1311
1312 ;*****
1313 ;LOAD IN LOCATION "GDREV" THE PROM VERSION NUMBER THAT IS *
1314 ;COMPATIBLE WITH THIS DIAGNOSTIC *
1315 ; *
1316 ;EACH PROM CONTAIN A REV LEVEL AND A ECO LEVEL: *
1317 ;THE REV LEVEL IS MODIFIED EACH TIME A MODIFICATION IS DONE *
1318 ;THE ECO LEVEL IS MODIFIED WHEN THE PROM MODIFICATION NEED *
1319 ;A DIAGNOSTIC MODIFICATION *
1320 ;*****
1321
1322
1323
1324
1325
1326 012424 000001 GDREV: .WORD 1
1327
1328
1329
1330
1331
1332
1333
```

GLOBAL DATA SECTION

```

1335 ;*****
1336 ;* PROGRAM CONTROL FLAGS
1337 ;*****
1338 012426      000      INIFLG: .BYTE 0      ;PROGRAM INITIALIZING FLAG
1339                      .EVEN
1340 012430      000      LOKFLG: .BYTE 0      ;LOCK ON CURRENT TEST FLAG
1341 012431      000      QV.FLG: .BYTE 0      ;QUICK VERIFY FLAG
1342                      .EVEN
1343 012432      000000    UUT:      .WORD 0      ;CURRENT UNIT UNDER TEST
1344
1345
1346
1347
1348
1349
1350 ;*****
1351 ;* POINTERS TO KMOV11 VECTORS AND REGISTERS
1352 ;*****
1353 012434      000000    KMOV00: 0      ;POINTER TO KMOV11 INTRPT VECTOR 0
1354 012436      000000    KMOV01: 0      ;POINTER TO KMOV11 INTRPT SERVICE
1355 012440      000000    KMOV04: 0      ;POINTER TO KMOV11 INTRPT VECTOR 04
1356 012442      000000    KMOV02: 0      ;      "      "      "      "      02
1357 012444      000000    KMOV06: 0      ;      "      "      "      "      06
1358 012446      000000    KMTLVL: 0      ;POINTER TO KMOV11 TX INTRPT SERVICE PS
1359 012450      000000    KMVCSR: 0      ;POINTER TO KMOV11 CONTROL STATUS REGISTER
1360 012452      000000    KMVP02: 0      ;POINTER TO KMOV11 PORT REGISTER - SEL2
1361 012454      000000    KMVP04: 0      ;POINTER TO KMOV11 PORT REGISTER - SEL4
1362 012456      000000    KMVP06: 0      ;POINTER TO KMOV11 PORT REGISTER - SEL6
1363
1364 012460      000000    KMVP10: 0      ;POINTER TO KMOV11 PORT REG -SEL10
1365 012462      000000    KMVP12: 0      ;POINTER TO PORT REG -SEL 14
1366 012464      000000    KMOV14: 0      ;POINTER TO PORT REG -SEL14
1367 012466      000000    KMVP16: 0      ;POINTER TO PORT REG 16
1368
1369 012470      000000    LOOP: 0      ;POINTER TO LOOP BACK CONNECTOR

```

GLOBAL DATA SECTION

1371
1372
1373 012472
1374
1375
1376 012472
1377 012672

***** PRIMARY REG ADRS STORAGE FOR THIS UNIT *****
;THESE LOCATIONS WILL BE LOADED FOR THE CURRENT UNIT, IN INIT CODE
REGADR:

***** STACK USED FOR SUBROUTINE LINKAGE *****
;*****
.BLKW 100
SSTACK:

GLOBAL DATA SECTION

1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391 012672
1392
1393
1394
1395
1396
1397
1404
1405
1406
1407
1408

.SBTTL GLOBAL TEXT SECTION

```
*****  
;# THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,  
;# MESSAGES, AND ASCII INFORMATION THAT ARE USED IN  
;# MORE THAN ONE TEST.  
*****  
;*****  
;# NAMES OF DEVICES SUPPORTED BY PROGRAM  
;*****  
;# DEVTYP <KMOV11A>  
  
;  
; FORMAT STATEMENTS USED IN PRINT CALLS  
;
```

GLOBAL SUBROUTINES

```

1410      .SBTTL  GLOBAL SUBROUTINES
1411
1412      ;-----
1413      ;MACRO'S NEEDED TO CALL SUBROUTINES
1414      ;-----
1415
1416
1417      .MACRO  CLRMAR
1418              ROMCLK
1419              004000
1420      .ENDM   CLRMAR
1421
1422
1423
1424
1425
1426      ;////////////////////////////////////
1427      ;/      THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST
1428      ;////////////////////////////////////
1429
1430
1431
1432
1433      ;ROUTINE TO WAIT FOR EVENT OR TIMEOUT
1434
1435
1436
1437      ;CALLING SEQUENCE:      JSR    PC,WAIT1
1438      ;                      JSR    PC,WAIT2
1439
1440
1441      ;INPUTS PARAMETERS:      DELCT1,DELCT2
1442
1443
1444      ;                      INC DELCT1 UNTIL 0
1445      ;                      DEC DELCT2 UNTIL 0      DELCT2= NUMB OF WAIT1 PASSES
1446
1447
1448
1449
1450
1451
1452
1453      012702  005237  002260      WAIT2:  INC      DELCT1
1454      012706  001375              BNE      WAIT2
1455
1456      012710              BREAK
1457
1458
1459      012712  005337  002262              DEC      DELCT2
1460      012716  001371              BNE      WAIT2
1461
1462      012720  000207              RTS      PC
1463
1464
1465
1466

```

L3

KMV11 A LINE CNT DIAGNOSTIC MACRO M1200 17-APR-84 08:57 PAGE 32-1

SEQ 0037

GLOBAL SUBROUTINES

1467					
1468					
1469	012722	005237	002260	WAIT1:	INC DELCT1
1470	012726	001375			BNE WAIT1
1471					
1472	012730	000207			RTS PC

GLOBAL SUBROUTINES

```
1474      ;MACRO TO WAIT A FEW MS
1475
1476
1477      ;CALLING SEQUENCE:      WAITA      X      0<X<177777
1478      ;                      WAITB      X,Y    0<X OR Y<177777
1479
1480
1481      .MACRO WAITA      X
1482      MOV      @X,DELCT1      ;LOAD COUNT
1483      JSR      PC,WAIT1      ;WAIT
1484
1485      .ENDM
1486
1487
1488
1489
1490
1491
1492
1493      .MACRO WAITB      X,Y
1494      MOV      @X,DELCT1
1495      MOV      @Y,DELCT2
1496      JSR      PC,WAIT2
1497
1498      .ENDM
1499
1500
1501
```

GLOBAL SUBROUTINES

1503						;ROUTINE TO DROP UNIT AFTER 5 ERROR
1504						
1505						
1506						;JSR PC,CHKMAX
1507						
1508						
1509						
1510						
1511						
1512						
1513						
1514						
1515	012732					CHKMAX: INLOOP ;LOOPING ON ERROR?
1516	012734				1\$	BCOMPLETE ;IF YES, EXIT
1517						
1518						
1519	012736				RFLAGS RO	;GET OPERATOR FLAG
1520	012740	032700	000040		BIT #IDU,R0	;IS DROPPING INHIBITTED?
1521	012744	001026			BNE 1\$	;IF YES EXIT
1522						
1523						
1524	012746	005237	002234		INC ERRCNT	;UPDATE ERROR COUNT
1525	012752	023737	002234	002232	CMP ERRCNT,MAXERR	;TOO MANY ERROR?
1526	012760	003420			BLE 1\$	;IF NOT JUMP
1527						
1528						
1529	012762				PRINTF #NERRS,MAXERR,UUT	;TOO MANY ERROR!
1530	013012				DODU UUT	;DROP UNIT
1531						
1532	013020				DOCLN	;END THE SUBPASS
1533						
1534	013022	000207			1\$: RTS PC	
1535						
1536						
1537						
1538						
1539					.NLIST BEX	
1540	013024	045	116	045 NERRS:	.ASCIZ /#N#AMORE THAN #D3#A	ERRORS ON UNIT #D2/
1541					.LIST BEX	
1542					.EVEN	
1543						
1544						
1545						
1546						
1547						

GLOBAL SUBROUTINES

```

1549          ;ROUTINE TO CHECK REGISTER BSEL0 AND TO REPORT ERROR
1550
1551
1552
1553
1554
1555
1556          ;CALLING SEQUENCE:      JSR      PC,TSTERR
1557
1558
1559
1560          ;OUTPUT PARAMETERS:      RETURN TO      PC      IF TEST IS OK
1561          ;                          :              PC+2    IF TIMEOUT DURING TEST
1562          ;                          :              PC+4    IF NO KMV11 ANSWER
1563          ;                          :              PC+6    IF DATA CMP ERROR
1564
1565
1566
1567
1568
1569
1570 013074 004537 013644      TSTERR: JSR      R5,CBSEL0      ;LOOK IF BSEL0=0
1571 013100 000000              .WORD      0
1572 013102 000411              BR        1$      ;TEST IS OK ,RTS PC
1573
1574
1575 013104 004537 013644              JSR      R5,CBSEL0      ;LOOK IF BSEL0=200
1576 013110 000200              .WORD      200
1577 013112 000406              BR        2$      ;TIMEOUT DURING TEST,RTS PC+2
1578
1579
1580 013114 004537 013644              JSR      R5,CBSEL0      ;LOOK IF BSEL0=100
1581 013120 000100              .WORD      100
1582 013122 000405              BR        3$      ;DATA CMP ERROR,RTS PC+6
1583
1584
1585
1586 013124 000407              BR        4$      ;NO KMV11 ANSWER ,RTS PC+4
1587
1588
1589
1590 013126 000207      1$:      RTS      PC      ;TEST OK
1591
1592
1593 013130 062716 000002      2$:      ADD      @2,(SP)
1594 013134 000207              RTS      PC      ;TIMEOUT ERROR
1595
1596
1597 013136 062716 000006      3$:      ADD      @6,(SP)
1598 013142 000207              RTS      PC      ;DATA CMP ERROR
1599
1600
1601 013144 062716 000004      4$:      ADD      @4,(SF)
1602 013150 000207              RTS      PC      ;NO KMV11 ANSWER
1603

```

NUMBER GENERATOR

```

1605      .SBTTL  NUMBER GENERATOR
1606      :
1607      :
1608      :
1609      :
1610      :
1611      :
1612      :
1613      :
1614      :
1615      :
1616      :
1617      :
1618      :
1619      :
1620      :
1621      :
1622      :
1623      :
1624      :
1625      :
1626      :
1627      :
1628      :
1629      :
1630      :
1631      :
1632      :
1633      :
1634      :
1635      :
1636      :
1637      :
1638      :
1639      :
1640      :
1641      :
1642      :
1643      :
1644      :
1645      :
1646      :
1647      :
1648      :
1649      :
1650      :
1651      :
1652      :
1653      :
1654      :
1655      :
1656      :
1657      :
1658      :
1659      :
1660      :
1661      :

```

DESCRIPTION:

ROUTINE TO GENERATE DATA PATTERNS,
THE TYPE OF PATTERN IS SELECTED BY R3, AND THE
PATTERN GENERATED IS RETURNED IN LOCATION "DATA"
AND LOCATION "GOOD"

CALLING SEQUENCE:

JSR PC,GENER

INPUT PARAMETERS:

R3 CONTAINS THE PATTERN NUMBER

R3=0	ALL ZEROES
1	ALL ONES
2	010101 ETC BIT PATTERN
3	101010 ETC BIT PATTERN
4	ROTATING 1 IN A ZERO WORD
5	ROTATING 0 IN AN ALL ONE WORD
6	PSEUDO RANDOM NUMBER
7	INCREMENTING DATA PATTERN, GOOD CONTAINS THE VALUE TO BE UPDATED

IMPLICIT INPUT PARAMETERS.

NONE

OUTPUT PARAMETERS:

THE NUMBER GENERATED IS HELD IN
DATA AND GOOD.

IMPLICIT OUTPUT PARAMETERS:

NONE

COMPLETION CODES:

NONE

POSSIBLE ERROR CODES:

NONE

NUMBER GENERATOR

```

1662
1663
1664 013152 042703 177770
1665 013156 004737 013452
1666 013162 006303
1667 013164 000173 013170
1668 013170 013210
1669 013172 013214
1670 013174 013222
1671 013176 013230
1672 013200 013236
1673 013202 013246
1674 013204 013304
1675 013206 013424
1676 013210 005000
1677 013212 000507
1678 013214 005000
1679 013216 005100
1680 013220 000504
1681 013222 012525 052525
1682 013226 000501
1683 013230 012700 125252
1684 013234 000476
1685 013236 000241
1686 013240 004737 013260
1687 013244 000472
1688 013246 000241
1689 013250 004737 013260
1690 013254 005100
1691 013256 000465
1692 013260 006037 013302
1693 013264 001003
1694 013266 012737 100000 013302
1695 013274 013700 013302
1696 013300 000207
1697 013302 000001
1698 013304 012737 000005 002336
1699 013312 004737 013324
1700 013316 013700 002342
1701 013322 000443
1702 013324 013702 002342
1703 013330 001002
1704 013332 013702 002334
1705 013336 032737 000777 002336
1706 013344 001003
1707 013346 012737 000001 002336
1708 013354 013703 002336
1709 013360 013702 002342
1710 013364 033702 002340
1711 013370 001405
1712 013372 005102
1713 013374 033702 002340
1714 013400 001401
1715 013402 000402
1716 013404 000241
1717 013406 000401
1718 013410 000261

:
:
GENER: BIC #177770,R3
JSR PC,SAVREG
ASL R3
JMP @GENSEL(R3)
GENSEL: GEN0 ;ALL ZERO WORD
GEN1 ;ALL ONE WORD
GEN52 ;52 PATTERN
GEN25 ;25 PATTERN
GENR1 ;ROTATE '1' EACH CALL
GENR0 ;ROTATE '0' EACH CALL
GENRAN ;RANDOM NUMBER
GENINC ;INCREMENTING COUNT
GEN0: CLR R0 ;0>R0
BR GENEX
GEN1: CLR R0 ;NOT0>R0
COM R0
BR GENEX
GEN52: MOV #52525,R0 ;5252>R0
BR GENEX
GEN25: MOV #125252,R0 ;125252>R0
BR GENEX
GENR1: CLC
JSR PC,GENROT
BR GENEX ;SHIFT 1 > R0
GENR0: CLC
JSR PC,GENROT
COM R0 ;SHIFT 0 > R0
BR GENEX
GENROT: ROR GENISH ;ROTATE 1 PATTERN
BNE GENER1 ;= 0?
MOV #100000,GENISH ;YES, SET MSB
GENER1: MOV GENISH,R0 ;PUT 1 IN R0
RTS PC ;AND EXIT
GENISH: 1
GENRAN: MOV #5,RANSEL ;SET SELECT VALUE TO 5
JSR PC,RANGEN ;GENERATE RANDOM NUMBER IN R0
MOV RANDN,R0
BR GENEX
RANGEN: MOV RANDN,R2
BNE RAN1 ;IS RANDOM = 0
MOV RANST,R2 ;YES, PUT RANDOM START VALUE IN
RAN1: BIT #777,RANSEL ;NO;IS RANSEL SELECT VALUE = 0
BNE RAN2 ;NO
MOV #1,RANSEL ;YES: SET RANSEL = 1
RAN2: MOV RANSEL,R3
MOV RANDN,R2
BIT RANMTA,R2 ;GET R2 <0 AND 1>
BEQ RANCLC
COM R2
BIT RANMTA,R2
BEQ RANCLC
BR RANSEC
RANCLC: CLC
BR RAN4
RANSEC: SEC

```

E4

NUMBER GENERATOR

1719	013412	006037	002342	RAN4:	ROR	RANDN	;ROTATE C TO B15
1720	013416	005303			DEC	R3	;IS THIS NUMBER REQUIRED?
1721	013420	001357			BNE	RAN2+4	;NO, GET ANOTHER
1722	013422	000207		RANEX:	RTS	PC	;YES, EXIT
1723	013424	013700	002264	GENINC:	MOV	GOOD,R0	;INCREMENTS LOC. 'GOOD'
1724	013430	005200			INC	R0	
1725	013432	010037	002264	GENEX:	MOV	R0,GOOD	
1726	013436	004737	013532		JSR	PC,RSTREG	
1727	013442	013737	002264 012372		MOV	GOOD,DATA	
1728	013450	000207			RTS	PC	
1729							

```

SAVREG: GETPRI  SAVSTA
        SETPRI  MAXPRI
        MOV     (SP)+, SAVPC      ;SAVE PC FOR RETURN FROM THIS ROUTINE

```

SAVE REGISTERS

1788	013472	012637	002344	MOV	(SP)+,SAVPC1	
1789	013476	010546		MOV	R5,-(SP)	
1790	013500	010446		MOV	R4,-(SP)	
1791	013502	010346		MOV	R3,-(SP)	
1792	013504	010246		MOV	R2,-(SP)	
1793	013506	010146		MOV	R1,-(SP)	
1794	013510	010046		MOV	R0,-(SP)	
1795	013512	013746	002344	MOV	SAVPC1,-(SP)	
1796	013516	013746	002244	MOV	SAVPC,-(SP)	;PUT PC READY FOR
1797	013522			SETPRI	SAVSTA	
1798	013530	000207		RTS	PC	;RETURN
1799						
1800						
1801						

RESTORE REGISTERS

1803 .SBTTL RESTORE REGISTERS

1804 :

1805 :

1806 :

1807 :

1808 :

1809 :

1810 :

1811 :

1812 :

1813 :

1814 :

1815 :

1816 :

1817 :

1818 :

1819 :

1820 :

1821 :

1822 :

1823 :

1824 :

1825 :

1826 :

1827 :

1828 :

1829 :

1830 :

1831 :

1832 :

1833 :

1834 :

1835 :

1836 :

1837 :

1838 :

1839 :

1840 :

1841 :

1842 :

1843 :

1844 :

1845 :

1846 :

1847 :

1848 :

1849 :

1850 :

1851 :

1852 :

1853 :

1854 013532

1855 013540

1856 013546 012637 002244

1857 013552 012637 002344

1858 013556 012600

1859 013560 012601

DESCRIPTION:

RESTORE TO RESTORE THE GENERAL PURPOSE
REGISTERS. THE STACK IS LEFT IN THE SAME STATE AS IT
WAS WHEN SAVREG WAS CALLED.

CAUTION: REGISTER R0 IS NOT SAVED

CALLING SEQUENCE:

JSR PC,RSTREG

INPUT PARAMETERS:

NONE

IMPLICIT INPUT PARAMETERS:

NONE

OUTPUT PARAMETERS:

R1 THRU R5 RESTORED

IMPLICIT OUTPUT PARAMETERS:

NONE

COMPLETION CODES:

NONE

POSSIBLE ERROR CODES:

NONE

RSTREG: GETPRI SAVSTA
SETPRI MAXPRI
MOV (SP)+,SAVPC
MOV (SP)+,SAVPC1
MOV (SP)+,R0
MOV (SP)+,R1

RESTORE REGISTERS

1860	013562	012602		MOV	(SP)+,R2	
1861	013564	012603		MOV	(SP)+,R3	
1862	013566	012604		MOV	(SP)+,R4	
1863	013570	012605		MOV	(SP)+,R5	
1864	013572	013746	002344	MOV	SAVPC1,-(SP)	
1865	013576	013746	002244	MOV	SAVPC,-(SP)	;PUT PC READY FOR
1866	013602			SETPRI	SAVSTA	
1867	013610	000207		RTS	PC	

RESTORE REGISTERS

```

1869                ;CHECK CONTENT OF ONE OF THE 8 REGISTERS
1870
1871                ; CALLING SEQUENCE
1872                ;      JSR      R5,CKSELN                ; N = REGISTER NUMBER
1873                ;      .WORD A                          A=EXPECTED CONTENT OF REGISTER N
1874
1875                ;OUTPUT PARAMETER:
1876                ;      BRANCH IN PC+2 IF ERROR DETECTED
1877                ;      BRANCH IN PC IF NO ERROR DETECTED
1878
1879
1880
1881
1882
1883 013612 012537 002264          CKSEL0: MOV      (R5)+,GOOD      ;WRITE GOOD
1884 013616 017737 176626 002310      MOV      @KMVCSR,SELO      ;READ SEL 0
1885 013624 023737 002310 002264      CMP      SELO,GOOD        ;CMP ?
1886 013632 001001                BNE      1$
1887 013634 000402                BR       2$
1888 013636 062705 000002          1$:  ADD      #2,R5
1889 013642 000205                2$:  RTS      R5
1890
1891
1892
1893
1894
1895
1896
1897 013644 005037 002264          CBSEL0: CLR      GOOD
1898 013650 012537 002264          MOV      (R5)+,GOOD
1899 013654 117737 176570 012370      MOVB     @KMVCSR,BSEL0
1900 013662 123737 012370 002264      CMPB     BSEL0,GOOD
1901 013670 001001                BNE      1$
1902 013672 000402                BR       2$
1903 013674 062705 000002          1$:  ADD      #2,R5
1904 013700 000205                2$:  RTS      R5

```

RESTORE REGISTERS

```

1906          ;ROUTINE TO CHECK ALL REGISTER FROM SEL0 TO SEL16
1907
1908
1909          ;CALLING SEQUENCE:
1910          ;      JSR R5,CKALL
1911          ;      .WORD A          A = EXPECTED VALUE FOR SEL0
1912          ;      .WORD B          B      "      "      SEL2
1913          ;      .WORD C          C      "      "      SEL4
1914          ;      .WORD D          D      "      "      SEL6
1915          ;      .WORD E          E      "      "      SEL10
1916          ;      .WORD F          F      "      "      SEL12
1917          ;      .WORD G          G      "      "      SEL14
1918          ;      .WORD H          H      "      "      SEL16
1919
1920
1921          ;OUTPUT PARAMETER:
1922          ;      BRANCH IN PC+2 IF ERROR
1923          ;      BRANCH IN PC IF NO ERROR
1924
1925
1926
1927 013702 012537 002266      CKALL:  MOV      (R5)+,G00D0
1928 013706 012537 002272      MOV      (R5)+,G00D2
1929 013712 012537 002274      MOV      (R5)+,G00D4
1930 013716 012537 002276      MOV      (R5)+,G00D6
1931 013722 012537 002300      MOV      (R5)+,G00D10
1932 013726 012537 002302      MOV      (R5)+,G00D12
1933 013732 012537 002304      MOV      (R5)+,G00D14
1934 013736 012537 002306      MOV      (R5)+,G00D16
1935
1936 013742 017737 176502 002310      MOV      @KMVCSR,SEL0          ;READ SEL0
1937 013750 000240                      NOP
1938 013752 017737 176474 002314      MOV      @KMVP02,SEL2          ;READ SEL2
1939 013760 000240                      NOP
1940 013762 017737 176466 002316      MOV      @KMVP04,SEL4          ;READ SEL4
1941 013770 000240                      NOP
1942 013772 017737 176460 002320      MOV      @KMVP06,SEL6          ;READ SEL6
1943 014000 000240                      NOP
1944 014002 017737 176452 002322      MOV      @KMVP10,SEL10         ;READ SEL10
1945 014010 000240                      NOP
1946 014012 017737 176444 002324      MOV      @KMVP12,SEL12         ;READ SEL12
1947 014020 000240                      NOP
1948 014022 017737 176436 002326      MOV      @KMVP14,SEL14         ;READ SEL14
1949 014030 000240                      NOP
1950 014032 017737 176430 002330      MOV      @KMVP16,SEL16         ;READ SEL16
1951
1952 014040 023737 002310 002266      CMP      SEL0,G00D0
1953 014046 001035                      BNE      1$
1954 014050 023737 002314 002272      CMP      SEL2,G00D2
1955 014056 001031                      BNE      1$
1956 014060 023737 002316 002274      CMP      SEL4,G00D4
1957 014066 001025                      BNE      1$
1958 014070 023737 002320 002276      CMP      SEL6,G00D6
1959 014076 001021                      BNE      1$
1960 014100 023737 002322 002300      CMP      SEL10,G00D10
1961 014106 001015                      BNE      1$
1962 014110 023737 002324 002302      CMP      SEL12,G00D12

```

RESTORE REGISTERS

1963	014116	001011			BNE	1\$
1964	014120	023737	002326	002304	CMP	SEL14,GOOD14
1965	014126	001005			BNE	1\$
1966	014130	023737	002330	002306	CMP	SEL16,GOOD16
1967	014136	001001			BNE	1\$
1968						
1969	014140	000402			BR	2\$
1970	014142	062705	000002	1\$:	ADD	#2,R5
1971	014146	000205		2\$:	RTS	R5

RESTORE REGISTERS

;ROUTINE TO CHECK SEL2 TO SEL16

```

1973
1974
1975
1976
1977
1978
1979 014150 012537 002272      CKREG:  MOV      (R5)+,G00D2
1980 014154 012537 002274      MOV      (R5)+,G00D4
1981 014160 012537 002276      MOV      (R5)+,G00D6
1982 014164 012537 002300      MOV      (R5)+,G00D10
1983 014170 012537 002302      MOV      (R5)+,G00D12
1984 014174 012537 002304      MOV      (R5)+,G00D14
1985 014200 012537 002306      MOV      (R5)+,G00D16
1986
1987
1988 014204 017737 176242 002314      MOV      @KMVP02,SEL2
1989 014212 000240              NOP
1990 014214 017737 176234 002316      MOV      @KMVP04,SEL4
1991 014222 000240              NOP
1992 014224 017737 176226 002320      MOV      @KMVP06,SEL6
1993 014232 000240              NOP
1994 014234 017737 176220 002322      MOV      @KMVP10,SEL10
1995 014242 000240              NOP
1996 014244 017737 176212 002324      MOV      @KMVP12,SEL12
1997 014252 000240              NOP
1998 014254 017737 176204 002326      MOV      @KMVP14,SEL14
1999 014262 000240              NOP
2000 014264 017737 176176 002330      MOV      @KMVP16,SEL16
2001
2002
2003
2004
2005 014272 023737 002314 002272      CMP      SEL2,G00D2
2006 014300 001031              BNE      1$
2007 014302 023737 002316 002274      CMP      SEL4,G00D4
2008 014310 001025              BNE      1$
2009 014312 023737 002320 002276      CMP      SEL6,G00D6
2010 014320 001021              BNE      1$
2011 014322 023737 002322 002300      CMP      SEL10,G00D10
2012 014330 001015              BNE      1$
2013 014332 023737 002324 002302      CMP      SEL12,G00D12
2014 014340 001011              BNE      1$
2015 014342 023737 002326 002304      CMP      SEL14,G00D14
2016 014350 001005              BNE      1$
2017 014352 023737 002330 002306      CMP      SEL16,G00D16
2018 014360 001001              BNE      1$
2019 014362 000402              BR       2$
2020
2021 014364 062705 000002      1$:  ADD      #2,R5
2022 014370 000205      2$:  RTS      R5

```

RESTORE REGISTERS

```

2024                               ;ROUTINE TO CLEAR KMOV11 MODULE
2025
2026
2027                               ;CALLING SEQUENCE:
2028                               ;      JSR PC,CLRKMV
2029
2030                               ;ROUTINE DESCRIPTION: CLEAR ALL CSR'S REGISTERS AND CHECK IF = 0
2031
2032
2033
2034 014372 005077 176052          CLRKMV: CLR      @KMOVCSR
2035 014376 012777 054000 176044  MOV      #MAINT0,@KMOVCSR
2036 014404                                WAITA    0
2037
2038
2039
2040
2041 014416 012702 000010          MOV      #10,R2
2042 014422 013701 012450          MOV      KMOVCSR,R1          ;LOAD ADDRESS
2043 014426 005021                1$:  CLR      (R1)+          ;CLEAR
2044 014430 005302                DEC      R2                ;ALL DONE
2045 014432 001375                BNE      1$                ;NO
2046 014434 004537 013702          JSR      R5,CKALL          ;CHECK ALL REG = 0
2047 014440 000000                .WORD    0
2048 014442 000000                .WORD    0
2049 014444 000000                .WORD    0
2050 014446 000000                .WORD    0
2051 014450 000000                .WORD    0
2052 014452 000000                .WORD    0
2053 014454 000000                .WORD    0
2054 014456 000000                .WORD    0
2055 014460 000404                BR       2$
2056 014462                                ERRHRD 1,EM0002,PRALL
2057 014472 000207                2$:  RTS      PC
2058

```

```

;OK BRANCH AT END
;CSR'S REGISTERS CAN'T BE CLEARED

```

RESTORE REGISTERS

```

2060          ;ROUTINE TO SET MAINT MODE 1 AND CHECK DCT11 CLEAR SELO AFTER HAVING DECODED
2061
2062
2063
2064          ;CALLING SEQUENCE:
2065          ;      JSR PC,MAINM1
2066
2067
2068
2069          ;GIVE AN ERROR IF MASTER CLEAR IS NOT CLEAR BY DCT11
2070          ;
2071          ;MAINT1= MASTER CLEAR=1 * MAINT 1 =0 * MODE = 1 : T11=HOLD
2072
2073
2074
2075
2076
2077
2078 014474 005077 175750      MAINM1: CLR      @KMOVCSR
2079 014500 000240              NOP
2080 014502 000240              NOP
2081 014504 000240              NOP
2082
2083 014506 012777 044000 175734      MOV      @MAINT1,@KMOVCSR      ;LOAD ADDRESS
2084 014514 012737 000000 002260      MOV      @0,DELCT1
2085 014522 012737 000001 002262      MOV      @1,DELCT2
2086 014530 004737 012702          JSR      PC,WAIT2
2087 014534 004537 013612          JSR      R5,CKSELO      ;CHECK SELO=0 BUT MODE BIT =1
2088 014540 004000              .WORD      4000
2089 014542 000404              BR         1$      ;OK BRANCH
2090 014544          ERRHRD      2,EM0001,PRSELO
2091 014554 000207      1$:      RTS      PC
2092
2093
2094
2095
2096

```

RESTORE REGISTERS

2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116

014556 012537 012412
014562 053777 012412 175660
014570 012737 000000 002260
014576 004737 012722
014602 000205

;ROUTINE TO SET TEST NUMBER ON BSELO

;CALLING SEQUENCE:
; JSR R5,TSTNUB
; .WORD A

A=TEST MICRO PROGRAM NUMBER

TSTNUB: MOV (R5)+,NUB
BIS NUB,@KMHVCSR
MOV #0000,DELCT1
JSR PC,WAIT1
RTS R5

;LOAD TEST NUMBFR

;WAIT

RESTORE REGISTERS

```

2118
2119 ;ROUTINE TO WRITE OR READ ONE OF THE KMV11 REGISTERS
2120
2121
2122
2123 ;CALLING SEQUENCE:
2124 ;JSR R5,WRITE
2125 ;.WORD A A=ADDRESS TO WRITE
2126 ;.WORD B B=DATA TO WRITE
2127 ;
2128 ;
2129 ;
2130 ;JSR R5,READ
2131 ;.WORD A A=ADDRESS TO READ
2132 ;
2133 ;
2134 ;
2135 ;MICRO DIAG NB 47 DESCRIPTION:
2136 ;WRITE: PUT ADDRESS TO WRITE IN SEL2
2137 ; PUT DATA TO WRITE IN SEL4
2138 ; SET BIT 0 OF SEL6(WRITE BIT)
2139 ; SET TEST NB 44
2140 ; KMV11 CLEAR BSEL0 WHEN DONE
2141 ;
2142 ;
2143 ;READ: PUT ADDRESS TO READ IN SEL2
2144 ; CLEAR BIT 0 IN SEL6
2145 ; SET TEST 47
2146 ; KMV11 READ ADDRESS IN SEL2 AND CLEAR BSEL0 WHEN DONE
2147 ;
2148 ;
2149 ;
2150
2151 014604 012577 175642 WRITE: MOV (R5),@KMVP02 ;WRITE ADDRESS
2152 014610 012577 175640 MOV (R5),@KMVP04 ; " DATA
2153 014614 012777 000001 175634 MOV @1,@KMVP06 ;BIT WRITE
2154
2155 014622 004537 014556 JSR R5,TSTNU8 ;SEND TEST NB 44
2156 014626 000047 .WORD 47
2157
2158 014630 000205 RTS R5 ;RETURN
2159
2160
2161
2162
2163
2164
2165 014632 012577 175614 READ: MOV (R5),@KMVP02 ;SET ADDRESS TO READ
2166 014636 005077 175612 CLR @KMVP04
2167 014642 005077 175610 CLR @KMVP06
2168
2169 014646 004537 014556 JSR R5,TSTNU8 ;SEND TEST NB 44
2170 014652 000047 .WORD 47
2171
2172
2173 014654 004737 013074 JSR PC,TSTERR ;CHECK BSEL 0
2174 014660 000412 BR 16 ;OK

```

RESTORE REGISTERS

2175	014662	000402				BR	2:	
2176	014664	000401				BR	2:	
2177	014666	000400				BR	2:	
2178								
2179	014670				2:	ERRHRD	4,EM0004	;NO KMV ANSWER
2180	014700	004737	012732			JSR	PC,CHKMAX	
2181	014704	000205				RTS	R5	
2182								
2183	014706	017737	175542	012366	1:	MOV	@KMVP04,BAD	;READ DATA IN BAD
2184	014714	000205				RTS	R5	
2185								
2186								
2187								
2188								
2189								

RESTORE REGISTERS

```
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
```

```

      .MACRO ED$CALL XY
      .LIST
;***** TEST 'XY' *****
      .NLIST
      .ENDM

      .MACRO BADHEAD
      .RADIX 10
      ED$CALL \T$TESTNUM+1
      .RADIX 8
      .ENDM
```

GLOBAL ERROR REPORT SECTION

```

2214 .SBTTL GLOBAL ERROR REPORT SECTION
2215
2216 ;////////////////////
2217 ;/ THE GLOBAL ERROR REPORT SECTION CONTAINS ERROR MESSAGES
2218 ;/ THAT ARE USED IN MORE THAN ONE TEST.
2219 ;////////////////////
2220
2221 .MLIST BEX
2222
2223 014716 040 102 125 TIM: .ASCIZ / BUS TIMEOUT/
2224
2225 014733 045 116 045 TFM36: .ASCIZ /%N%AREGISTER ADDRESS ERROR,ADDRESS = %06%A,UNIT = %02/
2226
2227 015021 115 101 123 EM0001: .ASCIZ /MASTER CLEAR FAIL TO RESET: DCT11 CAN'T CLEAR MASTER CLEAR /
2228
2229 015115 040 113 115 EM0002: .ASCIZ / KMOV11 REGISTERS CAN'T BE CLEARED /
2230
2231 015160 040 104 101 EM0003: .ASCIZ / DATA COMPARE ERROR ON KMOV11 REGISTER (SEL2 TO SEL16)/
2232
2233 015246 040 116 117 EM0004: .ASCIZ / NO ANSWER FROM KMOV11 /
2234
2235 015275 124 111 115 EM0006: .ASCIZ /TIMEOUT DURING KMOV11 MICRO TEST /
2236
2237 015336 111 116 124 EM0007: .ASCIZ /INTERUPT OCCURED ON KMOV11 AT INCORRECT VECTOR /
2238
2239 015416 113 115 126 EM0011: .ASCIZ /KMOV11 REAL TIME CLOCK FAILED TO INTERUPT /
2240
2241 015470 107 105 116 EM0012: .ASCIZ /GENERATOR COUNT CAN'T BE READ OR WRITE CORRECTLY /
2242
2243 015552 107 105 116 EM0013: .ASCIZ /GENERATOR OUTPUT ISN'T IN A GOOD STATE(NO ACTION ON OUTPUT)/
2244
2245 015646 116 117 040 EM0033: .ASCIZ /NO CHANGE IN BAUD RATE GENERATOR COUNT /
2246
2247 015716 116 117 040 EM0014: .ASCIZ /NO ACTION ON BAUD RATE GENERATOR OUTPUT /
2248
2249 015767 105 122 122 EM0015: .ASCIZ /ERROR WHEN TRANSMITTING IN INTERNAL LOOP WITHOUT INTERUPTS /
2250
2251 016063 105 122 122 EM0016: .ASCIZ /ERROR WHEN TRANSMITTING FRAMES IN INTERNAL LOOPBACK MODE /
2252
2253 016156 105 122 122 EM0017: .ASCIZ /ERROR WHEN TRANSMITTING FRAMES IN EXTERNAL LOOPBACK /
2254
2255 016243 105 122 122 EM0022: .ASCIZ /ERROR DURING TRANSMISSION AND RECEPTION OF FRAMES /
2256
2257 016326 122 105 101 EM0023: .ASCIZ /REAL TIME CLOCK INTERUPT OCCURED TOO EARLY /
2258
2259 016402 111 116 103 EM0024: .ASCIZ /INCORRECT KMOV11 REPLY /
2260
2261 016431 116 117 040 EM0027: .ASCIZ /NO LOOP BACK CONNECTOR,TEST NOT EXECUTED /
2262
2263 016503 105 122 122 EM0031: .ASCIZ /ERROR WHEN TRANSMITTING IN INTERNAL LOOP WITHOUT INTERUPTS /
2264
2265 016577 115 117 104 EM0032: .ASCIZ /MODEM SIGNAL ERROR ON CHANNEL IN EXTERNAL LOOPBACK /
2266
2267 016663 040 120 122 EM0035: .ASCIZ / PROM REVISION IS NOT COMPATIBLE WITH DIAGNOSTIC REVISION/
2268
2269 016755 040 111 116 EM0036: .ASCIZ / INTERUPT OCCURED ON DCT11 WHEN REAL TIME CLOCK IS DISABLE/
2270

```

GLOBAL ERROR REPORT SECTION

```

2272 017050      045      116      045 MSEL0: .ASCIZ /%N%A SEL0 = %06%A SHOULD BE = %06%N/
2273
2274 017116      045      116      045 MREG0: .ASCIZ /%N%A SEL0 = %06%A SHOULD BE = %06/
2275 017162      045      116      045 MREG2: .ASCIZ /%N%A SEL2 = %06%A SHOULD BE = %06/
2276 017226      045      116      045 MREG4: .ASCIZ /%N%A SEL4 = %06%A SHOULD BE = %06/
2277 017272      045      116      045 MREG6: .ASCIZ /%N%A SEL6 = %06%A SHOULD BE = %06/
2278 017336      045      116      045 MREG10: .ASCIZ /%N%A SEL10 = %06%A SHOULD BE = %06/
2279 017402      045      116      045 MREG12: .ASCIZ /%N%A SEL12 = %06%A SHOULD BE = %06/
2280 017446      045      116      045 MREG14: .ASCIZ /%N%A SEL14 = %06%A SHOULD BE = %06/
2281 017512      045      116      045 MREG16: .ASCIZ /%N%A SEL16 = %06%A SHOULD BE = %06/
2282
2283
2284 017556      045      116      045 MINT: .ASCIZ /%N%A GOOD = %06%A BAD = %06/
2285
2286 017612      045      116      045 MBSEL0: .ASCIZ /%N%A BSEL0 = %06%A SHOULD BE = %06/
2287
2288
2289 017654      045      116      045 MVECT: .ASCIZ /%N%A RECEIVE BAD VECT = %06%A SHOULD BE = %06/
2290
2291
2292
2293
2294
2295 017730      045      116      045 MT11V: .ASCIZ /%N%A RECEIVE VECTOR = %06%A SHOULD BE = %06/
2296 020006      045      116      045 MFRAM1: .ASCIZ /%N%A RECEIVE FRAME IS = %06%A SHOULD BE = %06/
2297 020064      045      116      045 MFRAM2: .ASCIZ /%N%A TRANSMIT SPEED IS = %06%A FRAME LENGTH = %06/
2298
2299
2300 020146      045      116      045 MSTER1: .ASCIZ /%N%A ERROR STATUS = %06/
2301 020200      045      116      045 MSTER2: .ASCIZ /%N%A WORD COUNT DISCREPANCY = %06/
2302
2303 020243      045      116      045 MODEM1: .ASCIZ /%N%A TESTED MODEM SIGNAL IS = %06/
2304 020307      045      116      045 MODEM2: .ASCIZ /%N%A RESULT OF TEST IS = %06/
2305 020346      045      116      045 MODEM3: .ASCIZ /%N%A MODEM SIGNAL STATE IS = %06/
2306 020411      045      116      045 MODEM4: .ASCIZ /%N%A SEE TEST HEADER FOR SIGNAL DESCRIPTION /
2307
2308 020466      045      116      045 MRAMEF: .ASCIZ /%N%A TXDATA = %06%A , RXDATA = %06/
2309
2310 020537      045      116      045 MLOOP: .ASCIZ /%N%A NO LOOP BACK CONNECTOR,TEST NOT EXECUTED/
2311 .EVEN

```

GLOBAL ERROR REPORT SECTION

2313 020616		BGNMSG	PRSELO	;REPORT SELO
2314 020616		PRINTB	#MSELO,SELO,GOOD	
2315 020646	004737 012732	JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
2316 020652		BREAK		
2317 020654		ENDMSG		
2318				
2319				
2320				
2321 020656		BGNMSG	PRINT	
2322 020656		PRINTB	#MINT,GOOD,BAD	
2323 020706	004737 012732	JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
2324 020712		BREAK		
2325 020714		ENDMSG		
2326				
2327				
2328 020716		BGNMSG	PRALL	;REPORT CONTENT OF ALL CSR'S
2329 020716		PRINTB	#MREG0,SELO,GOOD0	
2330 020746		PRINTB	#MREG2,SEL2,GOOD2	
2331 020776		PRINTB	#MREG4,SEL4,GOOD4	
2332 021026		PRINTB	#MREG6,SEL6,GOOD6	
2333 021056		PRINTB	#MREG10,SEL10,GOOD10	
2334 021106		PRINTB	#MREG12,SEL12,GOOD12	
2335 021136		PRINTB	#MREG14,SEL14,GOOD14	
2336 021166		PRINTB	#MREG16,SEL16,GOOD16	
2337 021216	004737 012732	JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
2338 021222		BREAK		
2339 021224		ENDMSG		
2340				
2341				
2342				
2343				
2344				
2345				
2346 021226		BGNMSG	PRREG	;REPORT ALL CSR'S BUT SELO
2347 021226		PRINTB	#MREG2,SEL2,GOOD2	
2348 021256		PRINTB	#MREG4,SEL4,GOOD4	
2349 021306		PRINTB	#MREG6,SEL6,GOOD6	
2350 021336		PRINTB	#MREG10,SEL10,GOOD10	
2351 021366		PRINTB	#MREG12,SEL12,GOOD12	
2352 021416		PRINTB	#MREG14,SEL14,GOOD14	
2353 021446		PRINTB	#MREG16,SEL16,GOOD16	
2354 021476	004737 012732	JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
2355 021502		BREAK		
2356 021504		ENDMSG		
2357				
2358				
2359				
2360				
2361 021506		BGNMSG	PADFLT	;ADDRESS TEST
2362 021506		PRINTB	#TFM36,ADDR,UNIT	
2363 021536	004737 012732	JSR	PC,CHKMAX	
2364 021542		ENDMSG		
2365				
2366				
2367				
2368				
2369				

GLOBAL ERROR REPORT SECTION

2370				
2371				
2372				
2373	021544		BGNMSG	PBSELO
2374	021544		PRINTB	#MBSELO,BSELO,GOOD
2375	021574	004737 012732	JSR	PC,CHKMAX
2376	021600		BREAK	
2377	021602		ENDMSG	
2378				
2379				
2380				
2381				
2382				
2383				
2384				
2385				
2386	021604		BGNMSG	PVECT
2387	021604		PRINTB	#MVECT,VECT,GOOD
2388	021634	004737 012732	JSR	PC,CHKMAX
2389	021640		BREAK	
2390	021642		ENDMSG	
2391				
2392				
2393				
2394				
2395	021644		BGNMSG	PRT11V
2396	021644		PRINTB	#MT11V,VECT,GOOD
2397	021674	004737 012732	JSR	PC,CHKMAX
2398	021700		BREAK	
2399	021702		ENDMSG	
2400				
2401				
2402				
2403				
2404	021704		BGNMSG	PFRAME
2405	021704		PRINTB	#MFRAM1,RXDATA,TXDATA
2406	021734		PRINTB	#MFRAM2,TSPEED,LENGTH
2407	021764	004737 012732	JSR	PC,CHKMAX
2408	021770		BREAK	
2409	021772		ENDMSG	
2410				
2411				
2412				
2413				
2414				
2415	021774		BGNMSG	PMODEM
2416	021774		PRINTB	#MODEM1,GOOD
2417	022020		PRINTB	#MODEM2,BAD
2418	022044		PRINTB	#MODEM3,DATA
2419	022070		PRINTB	#MODEM4
2420	022110	004737 012732	JSR	PC,CHKMAX
2421	022114		BREAK	
2422	022116		ENDMSG	
2423				
2424				
2425				
2426				

GLOBAL ERROR REPORT SECTION

2427
2428
24292430 022120
2431 022120
2432 022150
2433 022152BGNMSG PRAMEF
PRINTB @MRAMEF,TXDATA,RXDATA
BREAK
ENDMSG

;SHORT REPORT FOR FRAME ERROR

2434
2435
2436
24372438 022154
2439 022154
2440 022200
2441 022224

004737 012732

2442 022230
2443 022232BGNMSG PRSTER
PRINTB @MSTER1,STAERR
PRINTB @MSTER2,WRDCNT
JSR PC,CHKMAX
BREAK
ENDMSG

;REPORT ERROR STATUS ,WORD CNT

;CHECK IF TOO MANY ERROR

2444
2445
2446
2447
2448
2449
2450

GLOBAL ERROR REPORT SECTION

2452
2453
2454
2455
2456
2457
2458
2459
2460
2461 022234
2462
2468
2469 022234
2470
2477
2478 022240
2479
2480

.SBTTL REPORT CODING SECTION

; THE REPORT CODING SECTION CONTAINS THE
; "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.

BGNRPT

EXIT RPT

ENDRPT

INITIALIZE SECTION

```

2482          .SBTTL  INITIALIZE SECTION
2483
2484          ;//////////
2485          ;/ THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
2486          ;/ AT THE BEGINNING OF EACH PASS.
2487          ;//////////
2488
2489 022242          BGNINIT
2490
2491
2526          .EVEN
2527
2528          .EVEN
2529
2530
2531
2532 022242          SETVEC  #140,#170000,#340          ;ODT ROM ADDRESS          ;JB REV A-0
2533
2534          ;INITIALIZE SUBROUTINE STACK
2535 022270 012705 012672          MOV      #SSTACK,R5
2536          ;STORE BASE LEVEL PROGRAM STACK POINTER
2537 022274 010637 002246          MOV      SP,PSTACK
2538 022300 005737 002250          TST      FTIME
2539 022304 001011          BNE      1$
2540 022306 013737 000004 002252          MOV      @#4,SAVE4
2541 022314 013737 000006 002254          MOV      @#6,SAVE6
2542 022322 012737 000001 002250          MOV      #1,FTIME
2543 022330 013737 002252 000004 1$: MOV      SAVE4,@#4
2544 022336 013737 002254 000006          MOV      SAVE6,@#6
2545
2546 022344          READEF  #EF.START          ;START COMMAND?
2547 022352          BCOMPLETE          SETUP          ;IF YES BRANCH
2548
2549 022354          READEF  #EF.CONTINUE          ;CONTINUE COMMAND?
2550 022362          BCOMPLETE          END
2551
2552 022364          READEF  #EF.NEW          ;NEW PASS?
2553 022372          BNCOMPLETE          NEXT          ;IF NOT EXIT SETUP
2554
2555 022374 012737 177777 012432 SETUP: MOV      #-1,UUT          ;INITIALISE UNIT NUMBER
2556
2557 022402 005237 012432          NEXT:  INC      UUT          ;POINT NEXT UNIT
2558 022406 023737 012432 002240          CMP      UUT,L$UIT          ;ALL DONE?
2559 022414 001521          BEQ      ABORT          ;IF YES END OF PASS
2560
2561 022416 013701 012432          MOV      UUT,R1
2562 022422          PRINTF  #RUNNING,R1          ;PRINT RUNNING MESSAGE
2563          .EVEN
2564
2565
2566 022444          GPWARD  UUT,R1          ;GET P TABLE
2567 022454          BNCOMPLETE          NEXT          ;IF NOT AVAILABLE GET NEXT
2568
2569
25 0 022456          GETPRM:
2571
2572 022456 011137 012450          MOV      (R1),KMOVCSR          ;GET ADDRESS OF KMOV11

```

INITIALIZE SECTION

Address	Op Code	Op 1	Op 2	Op 3	Op 4	Op 5	Op 6	Op 7	Op 8	Op 9	Op 10	Op 11	Op 12	Op 13	Op 14	Op 15	Op 16	Op 17	Op 18	Op 19	Op 20	Op 21	Op 22	Op 23	Op 24	Op 25	Op 26	Op 27	Op 28	Op 29	Op 30	Op 31	Op 32	Op 33	Op 34	Op 35	Op 36	Op 37	Op 38	Op 39	Op 40	Op 41	Op 42	Op 43	Op 44	Op 45	Op 46	Op 47	Op 48	Op 49	Op 50	Op 51	Op 52	Op 53	Op 54	Op 55	Op 56	Op 57	Op 58	Op 59	Op 60	Op 61	Op 62	Op 63	Op 64	Op 65	Op 66	Op 67	Op 68	Op 69	Op 70	Op 71	Op 72	Op 73	Op 74	Op 75	Op 76	Op 77	Op 78	Op 79	Op 80	Op 81	Op 82	Op 83	Op 84	Op 85	Op 86	Op 87	Op 88	Op 89	Op 90	Op 91	Op 92	Op 93	Op 94	Op 95	Op 96	Op 97	Op 98	Op 99	Op 100	Op 101	Op 102	Op 103	Op 104	Op 105	Op 106	Op 107	Op 108	Op 109	Op 110	Op 111	Op 112	Op 113	Op 114	Op 115	Op 116	Op 117	Op 118	Op 119	Op 120	Op 121	Op 122	Op 123	Op 124	Op 125	Op 126	Op 127	Op 128	Op 129	Op 130	Op 131	Op 132	Op 133	Op 134	Op 135	Op 136	Op 137	Op 138	Op 139	Op 140	Op 141	Op 142	Op 143	Op 144	Op 145	Op 146	Op 147	Op 148	Op 149	Op 150	Op 151	Op 152	Op 153	Op 154	Op 155	Op 156	Op 157	Op 158	Op 159	Op 160	Op 161	Op 162	Op 163	Op 164	Op 165	Op 166	Op 167	Op 168	Op 169	Op 170	Op 171	Op 172	Op 173	Op 174	Op 175	Op 176	Op 177	Op 178	Op 179	Op 180	Op 181	Op 182	Op 183	Op 184	Op 185	Op 186	Op 187	Op 188	Op 189	Op 190	Op 191	Op 192	Op 193	Op 194	Op 195	Op 196	Op 197	Op 198	Op 199	Op 200	Op 201	Op 202	Op 203	Op 204	Op 205	Op 206	Op 207	Op 208	Op 209	Op 210	Op 211	Op 212	Op 213	Op 214	Op 215	Op 216	Op 217	Op 218	Op 219	Op 220	Op 221	Op 222	Op 223	Op 224	Op 225	Op 226	Op 227	Op 228	Op 229	Op 230	Op 231	Op 232	Op 233	Op 234	Op 235	Op 236	Op 237	Op 238	Op 239	Op 240	Op 241	Op 242	Op 243	Op 244	Op 245	Op 246	Op 247	Op 248	Op 249	Op 250	Op 251	Op 252	Op 253	Op 254	Op 255	Op 256	Op 257	Op 258	Op 259	Op 260	Op 261	Op 262	Op 263	Op 264	Op 265	Op 266	Op 267	Op 268	Op 269	Op 270	Op 271	Op 272	Op 273	Op 274	Op 275	Op 276	Op 277	Op 278	Op 279	Op 280	Op 281	Op 282	Op 283	Op 284	Op 285	Op 286	Op 287	Op 288	Op 289	Op 290	Op 291	Op 292	Op 293	Op 294	Op 295	Op 296	Op 297	Op 298	Op 299	Op 300	Op 301	Op 302	Op 303	Op 304	Op 305	Op 306	Op 307	Op 308	Op 309	Op 310	Op 311	Op 312	Op 313	Op 314	Op 315	Op 316	Op 317	Op 318	Op 319	Op 320	Op 321	Op 322	Op 323	Op 324	Op 325	Op 326	Op 327	Op 328	Op 329	Op 330	Op 331	Op 332	Op 333	Op 334	Op 335	Op 336	Op 337	Op 338	Op 339	Op 340	Op 341	Op 342	Op 343	Op 344	Op 345	Op 346	Op 347	Op 348	Op 349	Op 350	Op 351	Op 352	Op 353	Op 354	Op 355	Op 356	Op 357	Op 358	Op 359	Op 360	Op 361	Op 362	Op 363	Op 364	Op 365	Op 366	Op 367	Op 368	Op 369	Op 370	Op 371	Op 372	Op 373	Op 374	Op 375	Op 376	Op 377	Op 378	Op 379	Op 380</
---------	---------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	----------

B6

KMV11 A LINE CNT DIAGNOSTIC

MACPO M1200 17-APR-84 08:57 PAGE 51-2

SEQ 0066

INITIALIZE SECTION

2630 022724

END: ENDINIT

2631

2632

2633

2634

AUTODROP SECTION

2636						.SBITL AUTODROP SECTION	
2637							
2638						***	
2639						; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF	
2640						; THE "ADR" FLAG WAS SET. THE UNIT(S) UNDER TEST ARE CHECKED TO	
2641						; SEE IF THEY WILL RESPOND. THOSE THAT DON'T ARE IMMEDIATELY	
2642						; DROPPED FROM TESTING.	
2643						---	
2644						.EVEN	
2645	022726					BGNAUTO	
2646							
2653							
2654							
2655							
2656						;DEVICE DOES NOT HAVE A "READY"	
2657	022726	013701	012450			MOV KMVCSR,R1	;R1 CONTAINS BASE KMV11 ADDRESS
2658	022732	012705	000007			MOV 07,R5	;7 REGISTERS TO BE TESTED
2659	022736	012737	022770	000004		MOV 02,R4	;SET OUT TIMEOUT TRAP
2660						MOV 0340,6	;LEVEL 7
2661	022744	012737	000300	000006		MOV 0300,6	;LEVEL 6
2662	022752	005711			1\$:	TST (R1)	;REFERENCE DEVICE REGISTERS
2663	022754	000240				NOP	
2664	022756	062701	000002			ADD 02,R1	;NEXT REGISTER
2665	022762	005305				DEC R5	;DEC REGISTER COUNT
2666	022764	001372				BNE 1\$	;BR IF NOT LAST REGISTER
2667	022766	000405				BR 3\$	
2668							
2669	022770	062706	000004		2\$:	ADD 04,SP	
2670	022774					DODU LOGDEV	
2671							
2672	023002	013737	002252	000004	3\$:	MOV SAVE4,4	
2673	023010	013737	002254	000006		MOV SAVE6,6	
2674	023016					ENDAUTO	
2675							
2676							
2677							
2678							

CLEANUP CODING SECTION

2680
2681
2682
2683
2684
2685
2686
2687 023020
2688
2689
2709
2710
2711
2712 023020
2713
2714 023022

.SBTTL CLEANUP CODING SECTION

;/;;;
;/ THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
;/ AT THE END OF EACH PASS.
;/;;;

BGNCLN

BRESET

ENDCLN

CLEANUP CODING SECTION

```
2716
2717      .SBTTL  DROP UNIT SECTION
2718
2719      ;//////////
2720      ;// THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
2721      ;// TO NO LONGER BE TESTED.
2722      ;//////////
2723
2724 023024      BGNDU
2725
2726
2727
2736
2737
2749
2750      .EVEN
2751
2752 023024      PRINTF  @DROPD,RO      ;UNIT DROPPED
2753
2754 023046      EXIT    DU
2755
2756
2757
2758
2759
2760 023052      045      116      045  DROPD:  .NLIST  BEX
2761      .ASCIZ  /#N#A UNIT #D2#A DROPPED/
2762      .LIST   BEX
2763      .EVEN
2764 023102      ENDDU
2765
2766
2767
2768
2769
```

ADD UNIT SECTION

2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2789
2790 023104
2791 023104
2792
2793
2794
2795
2796
2797

.SBTTL ADD UNIT SECTION

```
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
;/ THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
;/ TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING. IF
;/ "EF.AUNIT" IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
```

BGNAU
ENDAU

HARDWARE TESTS

2799
2800
2801
2802
2803
2804 023106
2805
2806
2807
2814
2820
2821
2822
2828
2829
2830
2842
2843
2844
2845
2851

.SBTTL HARDWARE TESTS

; START OF CODE BLOCK WHICH IS USED AS DATA
ROMMAP:;...
; TEST TO ...
;--

; BGNTST

; EXIT TST

; .EVEN
; .ENDTST

HARDWARE TESTS

2853 023106

BADHEAD

;***** TEST1 *****

2854

;*VERIFY THAT REFERENCING UNIBUS DEVICE REGISTERS

2855

;*DOES NOT CAUSE A TIME OUT TRAP

2856 023106

BADHEAD

;***** TEST1 *****

2857

BGNTST

2858 023106

MOV KMOVCSR,R1

;R1 CONTAINS KMOV11 ADDRESSES

2859 023106 013701 012450

MOV #7,R5

;7 REGISTERS TO BE TESTED

2860 023112 012705 000007

MOV #2\$,4

;SET OUT TIMEOUT TRAP

2861 023116 012737 023154 000004

; MOV #340,6

;LEVEL 7

;JB REV A-0

2862

; MOV #300,6

;LEVEL 6

;JB REV A-0

2863 023124 012737 000300 000006

1\$:

TST (R1)

;REFERENCE DEVICE REGISTERS

2864 023132 005711

2865 023134 000240

NOP

2866 023136

ESCAPE

TST

2867 023142 062701 000002

ADD #2,R1

;NEXT REGISTER

2868 023146 005305

DEC

R5

;DEC REGISTER COUNT

2869 023150 001370

BNE

1\$

;BR IF NOT LAST REGISTER

2870 023152 000413

BR

3\$

2871

2872 023154 062706 000004

2\$:

ADD #4,SP

2873 023160 010137 002354

MOV R1,ADDR

2874 023164 013737 012432 012422

MOV UUT,UNIT

2875 023172

ERRHRD 0,TIM,PADFLT

;TIME OUT ERROR

2876

2877 023202 013737 002252 000004

3\$:

MOV SAVE4,4

2878 023210 013737 002254 000006

MOV SAVE6,6

2879 023216

ESCAPE

TST

2880

2881 023222

ENDTST

2882

.EVEN

2883

HARDWARE TESTS

2885 023224

BADHEAD

2886

;***** TEST2 *****

2887 023224

;CHECK PROM REVISION TO SEE IF COMPATIBLE WITH DIAGNOSTIC

BADHEAD

;***** TEST2 *****

2888

2889

2890

2891

2892 023224

STARS 1

2893

;READ LOCATION 2 OF THE PROM (ADDRESS 160002) WHICH CONTAINS PROM VERSION

2894

; NUMBER

2895

;CHECK IF DIAGNOSTIC AND PROM ARE COMPATIBLE AND GIVE AN ERROR IF NOT

2896 023224

STARS 1

2897

2898

2899

2900

2901

2902 023224

BGNTST

2903 023224 004737 014372

JSR

PC,CLRKMV

;CLEAR ALL REGISTERS

2904 023230 004737 014474

JSR

PC,MAINM1

;SET MAINT MODE

2905

2906

2907 023234 004537 014632

RFVPRO:

JSR

R5,READ

;READ LOCATION 160002

2908 023240 160002

.WORD

160002

2909

2910

2911 023242 023737 012424 012366

CMP

GDREV,BAD

;LOOK IF COMPATIBLE

2912 023250 001410

BEQ

14

;YES

2913

2914 023252

ERRHRD

7,EM0035

;REPORT THE ERROR

2915 023262 004737 012732

JSR

PC,CHKMAX

;CHECK IF TOO MANY ERROR

2916 023266

ESCAPE TST

2917 023272

i4:

2918 023272

ENDTST

HARDWARE TESTS

2920
2921
2922 023274

BADHEAD

;***** TEST3 *****

;REAL TIME CLOCK TEST

2923
2924 023274

BADHEAD

;***** TEST3 *****

2925
2926
2927
2928
2929
2930
2931 023274

STARS 1

;THIS TEST CHECK KMOV11 REAL TIME CLOCK.

;THE DCT11 FULLY EXECUTE THIS MICRO TEST AND GIVE A RESULT VIA CSR'S
;TO THE HOST. (TIMING IN CHECKED BY DCT11)

2932
2933
2934
2935
2936
2937

;TEST DESCRIPTION:

2938
2939
2940

;DCT11 ENABLE KMOV11 CLOCK, AND THEN SET UP A 80 MS PERIODE CLOCK.

2941
2942
2943
2944

;DCT11 WAIT FOR AT LEAST 80 MS AND CHECK IF AN INTERRUPT OCCUR
;ON DCT11 CHIP AT VECTOR 130

2945
2946
2947

;DCT11 TURN OF CLOCK, WAIT AGAIN FOR MORE THAN 80 MS AND CHECK THAT
;NO INTERRUPT OCCUR

2948
2949
2950

;ERROR REPORTING:

BSELO=200

IF TIMEOUT DURING TEST

BSELO=100

IF ERROR DURING TEST

BSELO=TEST NUB

IF NO KMOV11 ANSWER

BSELO=0

IF TEST IS OK

2951
2952
2953
2954

;IF ERROR

SEL6=1

IF NO INTERRUPT OCCUR

SEL6=2

IF BAD VECTOR

SEL6=4

IF INTERRUPT OCCUR WHEN CLOCK

2955
2956
2957
2958

IS NOT ENABLE

SEL6=10

INTERRUPT OCCUR TOO EARLY

2959
2960
2961

SEL2=EXPECTED VECTOR

2962
2963
2964

; MICRO TEST NB= 27

2965
2966
2967

2968
2969
2970

;CAUTION:

KMOV11 CRISTAL FREQUENCY CAN'T BE CHECKED WITH THIS TEST;
FOR THAT THE OPERATOR MUST SCOPE THE CRISTAL SIGNAL
DIRECTLY ON THE MODULE ON IC Y2 (13824 KHZ)

2971
2972
2973

2974 023274

STARS 1

HARDWARE TESTS

2976								
2977	023274				BGNTST			
2978	023274	004737	014372			JSR	PC,CLRKMV	;CLR REG
2979	023300	004737	014474			JSR	PC,MAINM1	;SET MAINT MODE
2980	023304	004537	014556		RTCLK:	JSR	R5,TSTNU8	
2981	023310	000027				.WORD	27	
2982								
2983	023312					WAITB	0,2	;WAIT FOR TEST EXECUTION
2984								
2985								
2986	023332	004737	013074			JSR	PC,TSTERR	;CHECK BSELO
2987	023336	000522				BR	1\$	;TEST OK
2988	023340	000423				BR	2\$	;TIMEOUT ERROR
2989	023342	000432				BR	3\$	;NO KMV ANSWER
2990								
2991								
2992	023344	022777	000001	167104		CMP	#1,@KMVP06	;ERROR DURING TEST ,SEE WHICH ONE
2993	023352	001436				BEQ	4\$	;NO INTERRUPT OCCUR
2994								
2995	023354	022777	000002	167074		CMP	#2,@KMVP06	
2996	023362	001442				BEQ	5\$	;INT ON BAD VECTOR
2997								
2998	023364	022737	000004	012456		CMP	#4,KMVP06	
2999	023372	001454				BEQ	6\$	;INT OCCUR WHEN CLOCK IS DESABLE
3000								
3001								
3002								
3003	023374	022737	000010	012456		CMP	#10,KMVP06	;INTERUPT OCCUR TOO EARLY
3004	023402	001460				BEQ	7\$	
3005								
3006	023404	000137	023564			JMP	10\$	;WRONG KMOV11 ANSWER
3007								
3008								
3009								
3010								
3011	023410				2\$:	ERRHRD	8,EM0006	;TIMEOUT ERROR
3012	023420	004737	012732			JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
3013	023424					ESCAPE	TST	
3014								
3015								
3016								
3017	023430				3\$:	ERRHRD	9,EM0004	;NO KMOV11 ANSWER
3018	023440	004737	012732			JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
3019	023444					ESCAPE	TST	
3020								
3021								
3022	023450				4\$:	ERRHRD	10,EM0011	;NO INTERRUPT OCCUR
3023	023460	004737	012732			JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
3024	023464					ESCAPE	TST	
3025								
3026								
3027								
3028	023470	017737	166760	012374	5\$:	MOV	@KMVP04,VECT	;READ BAD VECT
3029	023476	012737	000130	002264		MOV	#130,GOOD	
3030	023504					ERRHRD	11,EM0007	;INTERUPT OCCUR AT A BAD VECTOR
3031	023514	004737	012732			JSR	PC,CHKMAX	;CHECK IF TOO MANY ERROR
3032	023520					ESCAPE	TST	

HARDWARE TESTS

```
3033
3034
3035 023524      004737 012732      6$:  ERRHRD 12,EM0036      ;INT OCCUR W/ .N CHIP IS DESABLE
3036 023534      ;CHECK IF TOO MANY ERROR
3037 023540      JSR      PC,CHKMAX
3038      ESCAPE TST
3039
3040
3041
3042 023544      004737 012732      7$:  ERRHRD 13,EM0023      ;INTERUPT OCCUR TOO EARLY ON KMOV11
3043 023554      ;CHECK IF TOO MANY ERROR
3044 023560      JSR      PC,CHKMAX
3045      ESCAPE TST
3046
3047
3048
3049
3050
3051 023564      004737 012732      10$: ERRHRD 14,EM0024      ;INCORRECT KMOV11 RESULT
3052 023574      ;CHECK IF TOO MANY ERROR
3053 023600      JSR      PC,CHKMAX
3054      ESCAPE TST
3055
3056
3057
3058 023604      000240      1$:  NOP
3059 023606      ENDTST
3060
3061
3062
```

HARDWARE TESTS

3064 023610

BADHEAD

3065

;***** TEST4 *****

3066 023610

;BAUD RATE GENERATOR TEST

BADHEAD

;***** TEST4 *****

3067

3068

3069

3070

3071

3072

3073

3074

3075 023610

STARS 1

;THIS TEST READ THE STATUS AND THE OUTPUT OF THE BAUD RATE GENERATOR

;DURING EACH PHASE OF THE CLOCK PULSE.

;NOTE:THIS TEST AND ALL THE VERIFICATIONS ARE MADE BY THE DCT11 WHICH

;ONLY GIVE TEST RESULT VIA CSR'S TO THE HOST(TIMING IS CHECKED BY DCT11)

3079

3080

3081

3082

;TEST DESCRIPTION:

;-DCT11 LOAD GENERATOR COUNT WITH MAX COUNT (=+4.74 MSEC)

3083

3084

3085

;-READ BACK GENERATOR COUNT ,STATUS AND VALIDATE REPONSE.

3086

3087

3088

3089

3090

STEP 1:READ COUNT AFTER STARTING CLOCK

3091

CLOCK COUNT MUST BE NEGATIVE

3092

OUTPUT MUST BE = 1

3093

3094

3095

3096

ERROR REPORTING:
IF COUNT=POSITIVEBSELO=100=ERROR
SEL6 =1 =GENE COUNT CAN'T BE READ OR
WRITEN CORRECTLY

3098

3099

3100

IF OUTPUT=0

BSELO=100=ERROR
SEL6 =2 =GENE OUTPUT IS NOT CORRECT

3101

3102

3103

STEP 2: WAIT 2.5MSEC AND READ BACK AGAIN GENERATOR COUNT AND STATUS

3104

OUTPUT MUST BE = 0

3105

3106

3107

3108

3109

ERROR REPORTING:
IF OUTPUT =1BSELO=100= ERROR
SEL6=10 =OUTPUT ISN'T IN A GOOD STATE

3111

3112

3113

STEP3:WAIT 2.5 MSEC MORE AND READ BACK AGAIN GENERATOR COUNT AND STATUS

3114

OUTPUT MUST BE = 1

3115

3116

3117

3118

HARDWARE TESTS

```
3119      :      ERROR REPORTING:
3120      :
3121      :      IF OUTPUT=0      BSEL0=100=ERROR
3122      :      -ELSE EXIT      SEL6=40  =NO ACTION ON GENERATOR OUTPUT
3123      :
3124      :
3125      :
3126      :
3127      :
3128      :TEST 30=      TEST GENERATOR A
3129 023610      STARS 1
```

HARDWARE TESTS

3131	023610				BGNTST			
3132	023610	004737	014372		JSR	PC,CLARKMV		;CLR REG
3133	023614	004737	014474		JSR	PC,MAINM1		;SET MAINT MODE
3134	023620	004537	014556		JSR	R5,TSTNUB		
3135	023624	000030			.WORD	30		
3136								
3137								
3138	023626				BDRGEN: WAITB	0,1		;WAIT FOR TEST EXECUTION
3139								
3140	023646	004737	013074		JSR	PC,TSTERR		;CHECK BSELO TO SEE IF ERROR
3141	023652	000137	024052		JMP	BDROKO		;TEST OK BR AT END
3142	023656	000402			BR	2:		;TIME OUT ERROR
3143	023660	000401			BR	2:		;NO KMV11 ANSWER
3144	023662	000410			BR	3:		;ERROR DURING TEST
3145								
3146								
3147								
3148	023664				2:	ERRHRD 15,EM0004		;NO KMV11 ANSWER
3149	023674	004737	012732		JSR	PC,CHKMAX		;CHECK IF TOO MANY ERROR
3150	023700				ESCAPE	TST		
3151								
3152								
3153								
3154	023704				3:			;LOOK WHICH ERROR
3155	023704	017737	166546	002320	MOV	8KMVP06,SEL6		;READ SEL6
3156	023712	022737	000001	002320	CMP	#1,SEL6		;LOOK IF ERROR 1
3157	023720	001010			BNE	4:		;NO
3158								
3159	023722				ERRHRD	16,EM0012		;GENE COUNT CAN'T BE READ OR WRITTE CORRECTLY
3160	023732	004737	012732		JSR	PC,CHKMAX		;CHECK IF TOO MANY ERROR
3161	023736				ESCAPE	TST		
3162								
3163								
3164	023742	022737	000002	002320	4:	CMP	#2,SEL6	;LOOK IF ERROR 2
3165	023750	001010			BNE	5:		;NO
3166								
3167								
3168								
3169	023752				ERRHRD	17,EM0013		;GENE OUTPUT ISN'T IN A GOOD STATE
3170	023762	004737	012732		JSR	PC,CHKMAX		;CHECK IF TOO MANY ERROR
3171	023766				ESCAPE	TST		
3172								
3173								
3174								
3175	023772	022737	000010	002320	5:	CMP	#10,SEL6	;EROR10?
3176	024000	001414			BEQ	GENOUT		
3177	024002	022737	000040	002320	CMP	#40,SEL6		
3178	024010	001410			BEQ	GENOUT		
3179								
3180	024012				ERRHRD	18,EM0024		;WRONG KMV11 ANSWER
3181	024022	004737	012732		JSR	PC,CHKMAX		;CHECK IF TOO MANY ERROR
3182	024026				ESCAPE	TST		
3183								
3184								
3185								
3186								
3187								

C7

KMV11 A LINE CNT DIAGNOSTIC MACRO M1200 17-APR-84 08:57 PAGE 62-1

SEQ 0080

HARDWARE TESTS

3188
3189 024032
3190 024042 004737 012732
3191 024046
3192
3193
3194
3195 024052
3196 024052

GENOUT: ERRHRD 19.EM0014
 JSR PC,CHKMAX
 ESCAPE TST

;NO ACTION ON GENERATOR OUTPUT
;CHECK IF TOO MANY ERROR

BDROKO:
ENDTST

HARDWARE TESTS

3198
3199 024054

BADHEAD

;***** TEST5 *****
;TRANSMIT DIFFERENT FRAMES (OF 500 WORDS) AT 1.2 KBAUDS SPEED IN
;INTERNAL MODE WITHOUT ANY INTERRUPT ON CHANNEL A .

3200
3201
3202 024054

BADHEAD

;***** TEST5 *****

3203
3204
3205
3206
3207
3208
3209
3210 024054

STARS 1

;QBUS WRITE DIFFERENT TX TABLE OF 500 WORDS, LOAD IN KMOV11 CSR'S
;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED

3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236

;

;

;

;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHA AND WRITE BACK

;IN RX TABLE (TRANSFER FROM QBUS TO KMOV11 -DMA)

;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF

;RX TABLE =TX TABLE

;

;

;PARAMETERS SELECTION:

;SEL2= TX TABLE ADDRESS

;SEL4= TX TABLE LENGTH

;BSEL6= EXTENDED ADDRESS OF TX TABLE

;BSEL7= " " RX "

;SEL12= RX TABLE ADDRESS

;SEL14= SPEED SELECTION

;BSEL16= ERROR STATUS

;SEL10= RECEIVED BYTE COUNT DIFFERENCE BETWEEN RX AND TX TABLE

>0 IF TX>RX

<0 IF TX<RX

;BSEL0= TEST STATUS

;

;

;

;TEST STATUS DESCRIPTION:

;BSEL0= 0 -TEST DONE CHECK RX TABLE

;BSEL0= 200 -TIMEOUT ERROR

;BSEL0= TSTNB -NO KMOV11 ANSWER

;BSEL0= 100 -ERROR DURING TEST ,IN THAT CASE SEE WHICH KIND OF
ERROR BY TESTING BSEL16.

;

;

;

;ERROR STATUS DESCRIPTION:

;

;

;

;

;

;

;

;

;

;

;

;

;

;

;

;

;

WHEN BSEL0=100,GIVE STATUS AND WORD COUNT DISCREPANCY

BSEL16= BIT14=1 -FCS ERROR

BSEL16= BIT13=1 -OVERRUN ERROR

3251
3252

HARDWARE TESTS

```
3253      ;      BSEL16= BIT8 =1 =ILLEGAL INTERRUPT ERROR
3254      ;      BSEL16= BIT7 =1 =RX ABORT ERROR
3255      ;      BSEL16= BIT6 =1 =UNDERRUN ERROR
3256      ;      BSEL16= BIT5 =1 =WORD COUNT DISCREPANCY
3257      ;      BSEL16= BIT4 =1 =DMA IN TIMEOUT ERROR
3258      ;      BSEL16= BIT3 =1 =DMA OUT TIMEOUT ERROR
3259      ;      BSEL16= BIT2 =1 =CLOCK PROBLEM (NO BUFFER EMPTY)
3260      ;      BSEL16= BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USED
3261      ;                                     ONLY DURING SELF TEST)
3262      ;
3263      ;MICRO DIAG TEST DESCRIPTION:
3264      ;TEST 36      =TRANSMIT FRAMES AT 1,2KB SPEED ON CHANNEL A WITHOUT INTERRUPT
3265      ;
3266      ;
3267 024054      STARS 1
```

HARDWARE TESTS

```
3269 024054          BGNTST
3270 024054 004737 014372      JSR    PC,CLARKMV      ;CLR REG
3271 024060 005037 012400      CLR    CHANEL
3272 024064 005037 002256      CLR    FLAG
3273 024070 004737 014474      JSR    PC,MAINM1      ;SET MAINT MODE
3274 024074 012737 000500 012410  MOV    #500,LENGTH  ;SELECT LENGTH
3275
3276 024102 012737 013224 012406  MOV    #KB1.2,TSPEED  ;SELECT SPEED
3277
3278 024110 012703 000001      INTTX: MOV    #1,R3      ;SELECT A PATTERN
3279
3280
3281 024114 005203      TXSTAR: INC    R3      ;NEW ONE
3282 024116          BREAK
3283 024120 013704 012410      MOV    LENGTH,R4      ;LOAD LENGTH
3284 024124 012702 002362      MOV    #TTABLE,R2      ;TX TABLE ADDRESS
3285 024130 004737 013152 10$:  JSR    PC,GENER      ;WRITE TX TABLE
3286 024134 013722 012372      MOV    DATA,(R2)+
3287 024140 005304      DEC    R4
3288 024142 001372      BNE    10$      ;ALL DONE?
3289
3290
3291
3292 024144 013704 012410      MOV    LENGTH,R4
3293 024150 012702 006362      MOV    #RTABLE,R2      ;CLEAR RX TABLE
3294 024154 005022 11$:  CLR    (R2)+
3295 024156 005304      DEC    R4
3296 024160 001375      BNE    11$
3297
3298
3299
3300
3301 024162 013777 012406 166274  MOV    TSPEED,@KMVP14      ;SEND TX SPEED
3302 024170 012777 002362 166254  MOV    #TTABLE,@KMVP02      ;SEND TX TABLE ADDRESS
3303 024176 013777 012410 166250  MOV    LENGTH,@KMVP04      ;LOAD TX TABLE ADDRESS
3304 024204 012777 006362 166250  MOV    #RTABLE,@KMVP12      ;LOAD RX TABLE ADDRESS
3305 024212 005077 166240      CLR    @KMVP06
3306
3307
3308
3309
3310
3311 024216 004537 014556      JSR    R5,TSTNU8
3312 024222 000036          .WORD 36      ;DO TEST 36= CHA TEST
3313
3314
3315
3316 024224          WAITB 0.20      ;WAIT FOR TEST EXECUTION
3317
3318
3319 024244 004737 013074      JSR    PC,TSTERR      ;CHECK BSELO
3320
3321 024250 000427          BR    6$      ;TEST OK CHECK RX TABLE
3322 024252 000402          BR    3$      ;TIMEOUT ERROR
3323 024254 000401          BR    3$      ;NO KMOV11 ANSWER
3324 024256 000410          BR    4$      ;CHECK SEL16 TO SEE WHICH ONE
3325
```

HARDWARE TESTS

```

3326
3327 024260          3$:  ERRHRD  25,EM0004      ;NO KMOV11 ANSWER
3328 024270 004737 012732    JSR    PC,CHKMAX    ;CHECK IF TOO MANY ERROR
3329 024274          ESCAPE  TST
3330
3331
3332
3333 024300          4$:                      ;ERROR DURING TEST READ ERROR STATUS
3334                      ;TO CHECK WHICH ONE
3335
3336 024300 017737 166162 012416    MOV    @KMVP16,STAERR    ;READ ERROR STATUS
3337
3338 024306 017737 166146 012420    MOV    @KMVP10,WRDCNT    ;READ WORD COUNT DISCREPANCY
3339
3340 024314          ERRHRD  26,EM0031,PRSTER    ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3341                      ESCAPE  TST              ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3342 024324
3343
3344
3345
3346
3347
3348
3349
3350
3351 024330 012702 002362          6$:  MOV    @TTABLE,R2      ;LOAD TXTABLE ADDRESS
3352 024334 012705 006362          MOV    @RTABLE,R5      ; "  RXTABLE ADDRESS
3353 024340 013704 012410          MOV    LENGTH,R4      ;TABLE LENGTH
3354
3355 024344 022225          RXCK:  CMP    (R2)+,(R5)+      ;CHECK RX AND TX TABLE
3356 024346 001007          BNE    RXERR                  ;
3357 024350 005304          DEC    R4                      ;ALL CHECK?
3358 024352 001374          BNE    RXCK                    ;NO BRANCH
3359
3360
3361
3362 024354 022703 000005          CMP    #5,R3          ;ALL KIND OF PATTERN DONE?
3363 024360 001255          BNE    TXSTAR                  ;NO TRY WHI H NEW ONE
3364
3365 024362 000137 024504          JMP    RXEND
3366
3367 024366 162705 000002          RXERR:  SUB    #2,R5
3368 024372 162702 000002          SUB    #2,R2
3369
3370 024376 011237 012402          MOV    (R2),TXDATA
3371 024402 011537 012404          MOV    (R5),RXDATA
3372
3373 024406 005737 002256          TST    FLAG          ;LOOK IF 1ST ERROR
3374 024412 001014          BNE    7$
3375
3376 024414          ERRHRD  27,EM0015,PFRAME    ;DATA CMP ERROR
3377 024424 005237 002256          INC    FLAG
3378 024430 062702 000002          ADD    #2,R2          ;POINT NEXT ADDRESS
3379 024434 062705 000002          ADD    #2,R5
3380 024440 000137 024344          JMP    RXCK
3381
3382 024444          7$:  ERRHRD  27,0,PRAMEF      ;SHORT REPORT

```

HARDWARE TESTS

3383	024454	005237	002256	INC	FLAG	
3384	024460	062702	000002	ADD	#2,R2	
3385	024464	062705	000002	ADD	#2,R5	
3386	024470	022737	000010	CMP	#10,FLAG	;POINT NEXT ADDRESS
3387	024476	001322	002256	BNE	RXCK	;LOOK IF 10 REPORT
3388						
3389	024500			ESCAPE	TST	
3390						
3391						
3392	024504					
3393						
3394						
3395						
3396	024504					

RXEND:

ENDTST

HARDWARE TESTS

3398
3399
3400
3401 024506

BADHEAD

;***** TEST6 *****
;TRANSMIT DIFFERENT FRAME OF VARIOUS LENGTH (FROM 2BYTES TO 2K BYTES)
;AT 64 KBAUDS IN INTERNAL MODE ON CHANNEL A (TRANSMISSION WITH INTERRUPT)
BADHEAD
;***** TEST6 *****

3405
3406
3407
3408
3409
3410
3411
3412 024506

STARS 1

;QBUS WRITE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMOV11 CSR'S
;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED

3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437

;DCT11 EXECUTE THE TRANSFER IN INTERNAL MODE ON CHA AND WRITE BACK
;IN RX TABLE

;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF
;RX TABLE =TX TABLE
;SPEED=64 KBAUDS

;PARAMETERS SELECTION:

;SEL2= TX TABLE ADDRESS
;SEL4= TX TABLE LENGTH
;BSEL6= EXTENDED ADDRESS OF TX TABLE
;BSEL7= " " RX "
;SEL12= RX TABLE ADDRESS
;SEL14= SPEED SELECTION (= 154 IF 64KBAUDS)
;BSEL16= ERROR STATUS
;BSEL0= TEST STATUS
;SEL10= BYTE COUNT DISCREPANCY >0 IF TX>RX
;<0 IF TX<RX

3438
3439
3440
3441
3442
3443

;TEST STATUS DESCRIPTION:

;BSEL0= 0 =TEST DONE CHECK RX TABLE
;BSEL0= 200 =TIMEOUT ERROR
;BSEL0= TSTNB =NO KMOV11 ANSWER
;BSEL0= 100 =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16

3444
3445
3446
3447
3448
3449
3450
3451
3452

;ERROR STATUS DESCRIPTION:

;WHEN BSEL0=100,GIVE CONTAINMENT OF ERROR STATUS AND WORD COUNT DISCREPANCY

;BSEL16= BIT14=1 =FCS ERROR
;BSEL16= BIT13=1 =OVERRUN ERROR

HARDWARE TESTS

```
3453      ;      BSEL16= BIT8 =1 =ILLEGAL INTERRUPT ERROR
3454      ;      BSEL16= BIT7 =1 =RX ABORT ERROR
3455      ;      BSEL16= BIT6 =1 =UNDERRUN ERROR
3456      ;      BSEL16= BIT5 =1 =BYTE COUNT DISCREPANCY
3457      ;      BSEL16= BIT4 =1 =DMA IN TIMEOUT ERROR
3458      ;      BSEL16= BIT3 =1 =DMA OUT TIMEOUT ERROR
3459      ;      BSEL16= BIT2 =1 =CLOCK PROBLEM (NO BUFFER EMPTY)
3460      ;      BSEL16= BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
3461      ;                                     ONLY DURING SELF TEST)
3462      ;
3463      ;
3464      ;MICRO DIAG TEST DESCRIPTION:
3465      ;TEST 40      =TRANSMIT VARIOUS LENGTH FRAME AT 64 KBAUDS ON CHANNEL A
3466      ;
3467      ;
3468 024506 STARS 1
```

HARDWARE TESTS

```

3470 024506          BGNTST
3471 024506 004737 014372      JSR    PC,CL.RKMV      ;CLR REG
3472 024512 005037 012400      CLR    CHANEL
3473 024516 004737 014474      JSR    PC,MAINM1      ;SET MAINT MODE
3474 024522 005037 002256      CLR    FLAG
3475
3476
3477 024526 012703 000005      MOV    #5,R3          ;SELECT RANDOM PATTERN
3478          ; THE FOLLOWING RATE WAS CHANGED FROM 72KB TO 64KB.
3479          ;      MOV    #KB72,TSPEED      ;SELECT SPEED      ;JB REV A-0
3480 024532 012737 000154 012406      MOV    #KB64,TSPEED      ;SELECT SPEED      ;JB REV A-0
3481
3482 024540 012737 000001 012410      TXLTAR: MOV    #1,LENGTH      ;START WITH 2 CHARACTERS
3483
3484 024546 013704 012410      TXLBGN: MOV    LENGTH,R4
3485 024552 012702 002362      MOV    #TABLE,R2
3486 024556 004737 013152      10$:  JSR    PC,GENER      ;WRITE TX TABLE
3487 024562 013722 012372      MOV    DATA,(R2)+
3488 024566 005304          DEC    R4
3489 024570 001372          BNE    10$
3490
3491 024572          BREAK
3492
3493 024574 013704 012410      MOV    LENGTH,R4          ;CLEAR RX TABLE
3494 024600 012702 006362      MOV    #TABLE,R2
3495 024604 005022          20$:  CLR    (R2)+
3496 024606 005304          DEC    R4
3497 024610 001375          BNE    20$
3498
3499
3500
3501
3502
3503
3504 024612 013777 012406 165644      MOV    TSPEED,@KMVP14      ;SEND TX SPEED
3505 024620 012777 002362 165624      MOV    #TABLE,@KMVP02      ; " TX TABLE ADDRESS
3506 024626 013777 012410 165620      MOV    LENGTH,@KMVP04      ; " " " LENGTH
3507 024634 012777 006362 165620      MOV    #TABLE,@KMVP12      ;SEND RX TABLE ADDRESS
3508 024642 005077 165610      CLR    @KMVP06      ;CLR EXTENDED ADDRESS
3509
3510
3511
3512 024646 004537 014556          JSR    R5,TSTNUB
3513 024652 000040          .WORD    40          ;DO TEST 40= CHA TEST
3514
3515 024654          WAITB    0,2          ;WAIT FOR TEST EXECUTION
3516
3517
3518 024674 004737 013074          JSR    PC,TSTERR      ;CHECK BSELO
3519
3520 024700 000427          BR    6$          ;TEST OK CHECK RX TABLE
3521 024702 000402          BR    3$          ;TIMEOUT ERROR
3522 024704 000401          BR    3$          ;NO KMV11 ANSWER
3523 024706 000410          BR    4$          ;CHECK SEL16 TO SEE WHICH ONE
3524
3525
3526 024710          3$:  ERRHRD    28,EM0004      ;NO KMV11 ANSWER

```

HARDWARE TESTS

```

3527 024720 004737 012732      JSR      PC,CHKMAX      ;CHECK IF TOO MANY ERROR
3528 024724      ESCAPE    TST
3529
3530 024730      4$:
3531
3532
3533 024730 017737 165532 012416      MOV      @KMVP16,STAERR      ;READ ERROR STATUS
3534
3535 024736 017737 165516 012420      MOV      @KMVP10,WRDCNT      ;READ WORD COUNT DISCREPANCY
3536
3537 024744      ERRHRD    29,EM0022,PRSTER      ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3538
3539 024754      ESCAPE    TST      ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3540
3541
3542
3543
3544 024760 012702 002362      6$:      MOV      @TABLE,R2      ;LOAD TX TABLE ADDRESS
3545 024764 012705 006362      MOV      @RTABLE,R5      ; " RX " "
3546 024770 013704 012410      MOV      LENGTH,R4      ; " TX TABLE LENGTH
3547
3548
3549 024774 022522      RXLCK:  CMP      (R5),.(R2),      ;CMP TX AND RX TABLE
3550 024776 001015      BNE      RXLERR      ;BR IF ERROR
3551 025000 005304      DEC      R4      ;ALL DONE
3552 025002 001374      BNE      RXLCK      ;NO
3553
3554 025004 062737 000400 012410      ADD      @400,LENGTH      ;CHANGE LENGTH
3555 025012 022737 002000 012410      CMP      @2000,LENGTH      ;IS IT MAX?
3556 025020 100252      BPL      TXLBGN      ;NO DO TEST AGAIN WITH NEW TABLE
3557
3558
3559 025022 005303      DEC      R3      ;SELECT OTHER PATERNS
3560 025024 001245      BNE      TXLTAR
3561
3562 025026 000137 025150      JMP      RXLEND
3563
3564
3565
3566 025032 162705 000002      RXLERR: SUB      @2,R5
3567 025036 162702 000002      SUB      @2,R2
3568
3569 025042 011237 012402      MOV      (R2),TXDATA
3570 025046 011537 012404      MOV      (R5),RXDATA
3571
3572 025052 005737 002256      TST      FLAG      ;LOOK IF 1ST ERROR
3573 025056 001014      BNE      30$
3574
3575 025060      ERRHRD    30,EM0016,PFRAME      ;DATA CMP ERROR
3576 025070 005237 002256      INC      FLAG
3577 025074 062702 000002      ADD      @2,R2      ;POINT NEXT ADDRESS
3578 025100 062705 000002      ADD      @2,R5
3579 025104 000137 024744      JMP      RXCK
3580
3581 025110      30$:      ERRHRD    30,0,PRAMEF      ;SHORT REPORT
3582 025120 005237 002256      INC      FLAG
3583 025124 062702 000002      ADD      @2,R2

```

M7

KMV11 A LINE CNT DIAGNOSTIC I ACRO M1200 17-APR-84 08:57 PAGE 66-2

SEQ 0090

HARDWARE TESTS

3584	025130	062705	000002		ADD	02,R5		;POINT NEXT ADDRESS
3585	025134	022737	000010	002256	CMP	010,FLAG		;LOOK IF 10 REPORT
3586	025142	001314			BNE	RXLCK		
3587								
3588	025144				ESCAPE	TST		
3589								
3590								
3591								
3592								
3593	025150				RXLEND:			
3594	025150				ENDTST			

HARDWARE TESTS

3596
3597 025152

BADHEAD

;***** TEST7 *****

;TRANSMIT DIFFERENT FRAMES OF VARIOUS LENGTH IN EXTERNAL LOOP BACK

;MODE ON CHANNEL A AT 64KB

3598
3599
3600 025152

BADHEAD

;***** TEST7 *****

3601
3602
3603
3604
3605
3606
3607 025152

STARS 1

;

;AT BEGINNING OF TEST ,CHECK IF LOOP BACK CONNECTORS ARE INSTALLED

;OR NOT:IF NOT INSTALLED = EXIT TEST AND GIVE ERROR MESSAGE

3611
3612
3613
3614
3615
3616
3617
3618
3619
3620
3621
3622
3623
3624
3625
3626
3627

;

;

;

;

;

;

;QBUS WRITE DIFFERENT TX TABLE OF VARIOUS LENGTH, LOAD IN KMOV11 CSR'S

;THE TX AND RX TABLE ADDRESS ,THE TABLE LENGTH AND TRANSFER SPEED (64KB)

3619
3620
3621
3622
3623
3624
3625
3626
3627

;

;

;

;DCT11 EXECUTE THE TRANSFER IN EXTERNAL MODE ON CHA AND WRITE BACK

;IN RX TABLE

;QBUS CHECK BSEL0 TO SEE THE STATUS OF THE TEST AND IF TEST DONE CHECK IF

;RX TABLE =TX TABLE

3626
3627

;

;

;PARAMETERS SELECTION:

3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3640
3641

;

SEL2= TX TABLE ADDRESS

;

SEL4= TX TABLE LENGTH

;

BSEL6= EXTENDED ADDRESS OF TX TABLE

;

BSEL7= " " RX "

;

SEL12= RX TABLE ADDRESS

;

SEL14= SPEED SELECTION (=154 IF 64KB)

;

BSEL16= ERROR STATUS

;

BSEL0= TEST STATUS

;

SEL10= RECEIVE BYTE COUNT >0 IF TX>RX

;

<0 IF TX<RX

3639
3640
3641

;

;

;

;TEST STATUS DESCRIPTION:

3643
3644
3645
3646
3647
3648
3649
3650

;

BSEL0= 0 =TEST DONE CHECK RX TABLE

;

BSEL0= 200 =TIMEOUT ERROR

;

BSEL0= TSTNB =NO KMOV11 ANSWER

;

BSEL0= 100 =ERROR DURING TEST ,LOOK WHICH ONE BY TESTING BSEL16

;

;

;

;ERROR STATUS DESCRIPTION:

HARDWARE TESTS

```

3651      ;
3652      ;      WHEN BSELO=100,GIVE CONTAINIT OF ERROR STATUS AND WORD COUNT DISCREPANCY
3653      ;
3654      ;
3655      ;      BSEL16= BIT14=1 =FCS ERROR
3656      ;      BSEL16= BIT13=1 =OVERRUN ERROR
3657      ;      BSEL16= BIT8 =1 =ILLEGAL INTERRUPT ERROR
3658      ;      BSEL16= BIT7 =1 =RX ABORT ERROR
3659      ;      BSEL16= BIT6 =1 =UNDERRUN ERROR
3660      ;      BSEL16= BIT5 =1 =BYTE COUNT DISCREPANCY
3661      ;      BSEL16= BIT4 =1 =DMA IN TIMEOUT ERROR
3662      ;      BSEL16= BIT3 =1 =DMA OUT TIMEOUT ERROR
3663      ;      BSEL16= BIT2 =1 =CLOCK PROBLEM
3664      ;      BSEL16= BIT1 =1 =DATA COMPARE ERROR BETWEEN TX AND RX TABLE (USE
3665      ;                               ONLY DURING SELF TEST)
3666      ;
3667      ;
3668      ;MICRO DIAG TEST DESCRIPTION:
3669      ;TEST 42      =TRANSMIT VARIOUS LENGTH FRAME AT 64 KBAJDS SPEED ON CHANNEL A
3670      ;                               IN EXTERNAL LOOP BACK MODE
3671      ;
3672      ;
3673      ;
3674      ;
3675      ;CAUTION:
3676      ;-----
3677      ;RUN ONLY WITH EXTERNAL. LOOP BACK CONNECTOR:
3678      ;
3679      ;
3680      ;NOTE:
3681      ;
3682      ;TO FULLY TEST KMOV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
3683      ;EXTERNAL LOOP BACK CONECTOR
3684      ;
3685      ;EXTERNAL LOOP BACK CONNECTOR:
3686      ;-----
3687      ;KMOV11 A CAN OPERATE EITHER IN RS422 OR RS 423 LEVEL CONVERTERS
3688      ;
3689      ;
3690      ;RS422 LOOP BACK:
3691      ;TO TEST COMPLETELY A KMOV11 B IN RS422 MODE ,RUN THIS DIAGNOSTIC
3692      ;WHITH LOOP BACK CONNECTOR PLUG :
3693      ;-USE H3255 TO LOOP DIRECTLY AT THE OUTPUT OF THE MODULE
3694      ;-USE H3251 PLUG AT THE END OF BC55U MODEM CABLE CONNECTOR ASSY.
3695      ;
3696      ;
3697      ;RS423 LOOP BACK:
3698      ;TO TEST COMPLETELY A KMOV11-A IN RS423 MODE ,RUN THIS DIAGNOSTIC
3699      ;WHITH LOOP BACK CONNECTOR PLUG :
3700      ;-USE H3255 TO LOOP AT THE OUTPUT OF THE MODULE
3701      ;-USE H3251 PLUG AT THE END OF BC55H MODEM CABLE CONNECTOR ASSY.
3702      ;
3703      ;
3704      ;
3705      ;RS232 LOOP BACK:
3706      ;SAME AS FOR RS423.
3707      ;

```

HARDWARE TESTS

3708
3709
3710
3711
3712
3713
3714
3715
3716 025152

;CAUTION:
;USE OF H325 LOOP BACK CONNECTOR WILL CAUSE MESSAGES ERROR IN TEST 8.
;
;
;
;
;
;
;
STARS 1

HARDWARE TESTS

```

3718 025152          BGNTST
3719 025152 004737 014372      JSR    PC,CLRKMV      ;CLEAR REGISTERS
3720 025156 005737 012470      TST     LOOP        ;IS LOOP BIT=1?
3721 025162 001012              BNE     BGNTXA      ;YES GO ON TEST
3722 025164              PRINTF  #MLOOP          ;NO LOOP BACK CONNECTOR
3723                      ;TEST NOT EXECUTED
3724 025204          EXIT     TST
3725
3726
3727 025210 004737 014474      BGNTXA: JSR    PC,MAINM1      ;SET MAINT MODE
3728 025214 005037 002256      CLR     FLAG
3729
3730 025220 012703 000005      MOV     #5,R3          ;SELECT RANDOM PATTERN
3731      ; THE FOLLOWING RATE WAS CHANGED FROM 72KB TO 64KB.
3732      ;
3733 025224 012737 000154 012406  MOV     #KB72,TSPEED      ;SELECT SPEED      ;JB REV A-0
3734                      MOV     #KB64,TSPEED      ;SELECT SPEED      ;JB REV A-0
3735
3736 025232 012737 000001 012410 TXATAR: MOV     #1,LENGTH      ;1ST TABLE LENGTH(1 WORD)
3737
3738 025240 013704 012410      TXABGN: MOV     LENGTH,R4
3739 025244          BREAK
3740
3741 025246 012702 002362      MOV     #TABLE,R2
3742 025252 004737 013152      10$:  JSR     PC,GENER          ;WRITE TABLE
3743 025256 013722 012372      MOV     DATA,(R2)+
3744 025262 005304          DEC     R4
3745 025264 001372          BNE     10$
3746
3747
3748
3749 025266 013704 012410      MOV     LENGTH,R4          ;CLEAR RX TABLE
3750 025272 012702 006362      MOV     #RTABLE,R2
3751 025276 005022      20$:  CLR     (R2)+
3752 025300 005304          DEC     R4
3753 025302 001375          BNE     20$
3754
3755
3756
3757
3758
3759
3760
3761 025304 013777 012406 165152  MOV     TSPEED,@KMVP14      ;SEND TX SPEED
3762 025312 012777 002362 165132  MOV     #TABLE,@KMVP02      ; " TX TABLE ADDRESS
3763 025320 013777 012410 165126  MOV     LENGTH,@KMVP04      ; " " " LENGTH
3764 025326 012777 006362 165126  MOV     #RTABLE,@KMVP12      ;SEND RX TABLE ADDRESS
3765 025334 005077 165116          CLR     @KMVP06      ;CLR EXTENDED ADDRESS
3766
3767
3768
3769
3770
3771 025340 004537 014556      1$:  JSR     R5,TSTNUB
3772 025344 000042          .WORD  42          ;DO TEST 42= CHB TEST
3773
3774

```

HARDWARE TESTS

```

3775
3776 025346          2$:  WAITB  0,3          ;WAIT FOR TEST EXECUTION
3777
3778
3779 025366 004737 013074      JSR      PC,TSTERR      ;CHECK BSELO
3780
3781 025372 000427      BR      6$          ;TEST OK CHECK RX TABLE
3782 025374 000402      BR      3$          ;TIMEOUT ERROR
3783 025376 000401      BR      3$          ;NO KMOV11 ANSWER
3784 025400 000410      BR      4$          ;CHECK SEL16 TO SEE WHICH ONE
3785
3786
3787 025402          3$:  ERRHRD  32,EM0004      ;NO KMOV11 ANSWER
3788 025412 004737 012732      JSR      PC,CHKMAX      ;CHECK IF TOO MANY ERROR
3789 025416          ESCAPE  TST
3790
3791
3792 025422          4$:
3793
3794
3795 025422 017737 165040 012416      MOV      @KMVP16,STAERR      ;READ ERROR STATUS
3796
3797 025430 017737 165024 012420      MOV      @KMVP10,WRDCNT      ;READ WORD COUNT DISCREPANCY
3798
3799 025436          ERRHRD  33,FM0022,PRSTER      ;ERROR WHILE TX,RX FRAMES,GIVE ERROR
3800
3801 025446          ESCAPE  TST      ;GIVE ERROR STATUS,WORD CNT DISCREPANCY
3802
3803
3804
3805
3806
3807 025452 012702 002362          6$:  MOV      @TABLE,R2      ;LOAD TABLE PARAMETERS
3808 025456 012705 006362          MOV      @TABLE,R5
3809 025462 013704 012410          MOV      LENGTH,R4
3810
3811 025466 022225          RXACK:  CMP      (R2)*,(R5)*      ;CHECK TX AND RX TABLE
3812 025470 001015          BNE      RXAERR
3813 025472 005304          DEC      R4
3814 025474 001374          BNE      RXACK
3815
3816 025476 062737 000400 012410      ADD      @400,LENGTH      ;CHANGE LENGTH
3817 025504 022737 002000 012410      CMP      @2000,LENGTH
3818 025512 100252          BPL      TXABGN
3819 025514 005303          DEC      R3
3820 025516 001245          BNE      TXATAR      ;SELECT NEW PATTERN
3821 025520 000137 025642          JMP      RXAEND      ;ALL DONE
3822
3823
3824
3825 025524 162705 000002          RXAERR:  SUB      @2,R5
3826 025530 162702 000002          SUB      @2,R2
3827
3828 025534 011237 012402          MOV      (R2),TXDATA
3829 025540 011537 012404          MOV      (R5),RXDATA
3830
3831 025544 005737 002256          TST      FLAG      ;LOOK IF 1ST ERROR

```

HARDWARE TESTS

```
3832 025550 001014          BNE      30$
3833
3834 025552          ERRHRD  34,EM0015,PFRAME      ;DATA CMP ERROR
3835 025562 005237 002256    INC      FLAG
3836 025566 062702 000002    ADD      02,R2          ;POINT NEXT ADDRESS
3837 025572 062705 000002    ADD      02,R5
3838 025576 000137 024774    JMP      RXLCK
3839
3840 025602          30$:  ERRHRD  34,0,PRAMEF      ;SHORT REPORT
3841 025612 005237 002256    INC      FLAG
3842 025616 062702 000002    ADD      02,R2
3843 025622 062705 000002    ADD      02,R5          ;POINT NEXT ADDRESS
3844 025626 022737 000010 002256    CMP      010,FLAG      ;LOOK IF 10 REPORT
3845 025634 001314          BNE      RXACK
3846
3847 025636          ESCAPE  TST
3848
3849
3850
3851
3852
3853 025642          RXAEND:
3854 025642          ENDTST
```


HARDWARE TESTS

```

3905
3906      ;MOCFM SIGNAL LINK:
3907      ;
3908      ;          SEND                      RECEIVE
3909      ;
3910      ;
3911      ;          CCITT 105 (RTS)              CCITT 106 (CTS)
3912      ;          -----
3913      ;
3914      ;
3915      ;
3916      ;
3917      ;
3918      ;
3919      ;
3920      ;
3921      ;          CCITT 108 (DTR)              CCITT 107 (DSR)
3922      ;          -----
3923      ;
3924      ;
3925      ;
3926      ;
3927      ;          CCITT 111 (DTE)              CCITT 112 (DCE)
3928      ;          -----
3929      ;
3930      ;
3931      ;
3932      ;
3933      ;          CCITT 141 (LL)              CCITT 142 (TM)
3934      ;          -----
3935      ;
3936      ;
3937      ;
3938      ;
3939      ;          TIS                      CCITT 125 (RING)
3940      ;          -----
3941      ;
3942      ;
3943      ;
3944      ;
3945      ;
3946      ;
3947      ;CAUTION:
3948      ;-----
3949      ;RUN ONLY WITH EXTERNAL LOOP BACK CONNECTOR:
3950      ;
3951      ;
3952      ;
3953      ;TO BE FULLY TESTED ,KMOV11 DIAGNOSTIC MUST BE RUN WITH RS422 AND RS423
3954      ;EXTERNAL LOOP BACK CONECTOR
3955      ;
3956      ;EXTERNAL LOOP BACK CONNECTOR:
3957      ;-----
3958      ;KMOV11 A CAN OPERATE EITHER IN RS422 OR RS 423 LEVEL CONVERTERS
3959      ;
3960      ;
3961      ;RS422 LOOP BACK:

```

HARDWARE TESTS

```
3962 ;TO TEST COMPLETELY A K1V11 B IN RS422 MODE ,RUN THIS DIAGNOSTIC
3963 ;WITH LOOP BACK CONNECTOR PLUG :
3964 ;-USE H3255 TO LOOP DIRECTLY AT THE OUTPUT OF THE MODULE
3965 ;-USE H3251 PLUG AT THE END OF BC55U MODEM CABLE CONNECTOR ASSY.
3966 ;
3967 ;
3968 ;RS423 LOOP BACK:
3969 ;TO TEST COMPLETELY A K1V11-A IN RS423 MODE ,RUN THIS DIAGNOSTIC
3970 ;WITH LOOP BACK CONNECTOR PLUG :
3971 ;-USE H3255 TO LOOP AT THE OUTPUT OF THE MODULE
3972 ;-USE H3251 PLUG AT THE END OF BC55H MODEM CABLE CONNECTOR ASSY.
3973 ;
3974 ;
3975 ;
3976 ;RS232 LOOP BACK:
3977 ;SAME AS FOR RS423.
3978 ;
3979 ;CAUTION:
3980 ;USE OF H325 LOOP BACK CONNECTOR WILL CAUSE MESSAGES ERROR IN TEST 8.
3981 ;
3982 ;
3983 ;
3984 ;
3985 025644 STARS 1
3986
3987
```

HARDWARE TESTS

```

3989 025644          BGNTST
3990 025644 004737 014372      JSR      PC,CLARKMV      ;CLEAR ALL REGISTERS
3991
3992 025650 005737 012470      TST      LOOP
3993 025654 001012              BNE      MODSIG          ;LOOP BACK PRESENT GO ON
3994
3995 025656              PRINTF  @MLOOP          ;NO LOOP BACK CONNECTOR
3996
3997
3998 025676              EXIT      TST              ;GO TO FOLLOWING TEST
3999
4000
4001
4002 025702 004737 014474      MODSIG: JSR      PC,MAINM1      ;SET MAINTENANCE MODE
4003 025706 004537 014556      JSR      R5,TSTNUB
4004 025712 000045              .WORD  45              ;SEND TEST 45
4005
4006 025714              WAITB   0,4
4007
4008 025734 004737 013074      JSR      PC,TSTERR
4009 025740 000430              BR       3$
4010 025742 000402              BR       4$
4011 025744 000401              BR       4$
4012 025746 000406              BR       5$
4013
4014
4015
4016 025750              4$:    ERRHRD  36,EM0004      ;NO ANSWER
4017 025760              ESCAPE  TST
4018
4019 025764 017737 164462 002264 5$:    MOV      @KMVP02,GOOD      ;READ WHICH SIGNAL WAS TESTED
4020 025772 017737 164456 012366      MOV      @KMVP04,BAD
4021 026000 017737 164454 012372      MOV      @KMVP10,DATA      ; " " IS THE RESULT OF TEST
4022
4023 026006              ERRHRD  37,EM0032,PMODEM      ;READ SIGNAL VALUE
4024 026016              ESCAPE  TST
4025
4026 026022              3$:
4027 026022              MODEND:
4028
4029
4030 026022              ENDTST

```

HARDWARE TESTS

4032
4033
4034
4035
4036
4037
4038
4039
4040
4041
4042
4043
4044
4045
4046
4047
4048
4049
4050

.SBTTL HARDWARE PARAMETER CODING SECTION

```

////////////////////////////////////
;/ THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
;/ THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
;/ MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
;/ INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
;/ MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
;/ WITH THE OPERATOR.
////////////////////////////////////

```

4051 026024
4052
4053 026026
4054 026036
4055 026046
4056 026060
4057 026072
4058
4065
4066

BGNHRD

```

GPRMA  ADDRES,0,0,160000,177776,YES
GPRMA  VECTOR,2,0,0,674,YES
GPRMD  PRIPTY,4,0,7000,4,7,YES
GPRMD  LOOPBK,6,0,1,0,1,YES
ENDHRD

```

4067	026072	115	111	103	ADDRES: .ASCIZ /MICRO-CPU CSR ADDRESS : /
	026075	122	117	055	
	026100	103	120	125	
	026103	040	040	103	
	026106	123	122	040	
	026111	101	104	104	
	026114	122	105	123	
	026117	123	040	072	
	026122	040	000		
4068	026124	115	111	103	VECTOR: .ASCIZ /MICRO-CPU VECTOR ADDRESS : /
	026127	122	117	055	
	026132	103	120	125	
	026135	040	126	105	
	026140	103	124	117	
	026143	122	040	101	
	026146	104	104	122	
	026151	105	123	123	
	026154	040	072	040	
	026157	000			
4069	026160	115	111	103	PRIPTY: .ASCIZ /MICRO-CPU PRIORITY LEVEL : /
	026163	122	117	055	
	026166	103	120	125	
	026171	040	120	122	
	026174	111	117	122	
	026177	111	124	131	
	026202	040	114	105	
	026205	126	105	114	
	026210	040	072	040	

HARDWARE PARAMETER CODING SECTION

	026213	000			
4070	026214	111	123	040	LOOPBK: .ASCIZ /IS LOOP BACK CONNECTOR PLUGGED? 0=NO,1=YES: /
	026217	114	117	117	
	026222	120	040	102	
	026225	101	103	113	
	026230	040	103	117	
	026233	116	116	105	
	026236	103	124	117	
	026241	122	040	120	
	026244	114	125	107	
	026247	107	105	104	
	026252	077	040	060	
	026255	075	116	117	
	026260	054	061	075	
	026263	131	105	123	
	026266	072	040	000	
4071					.EVEN
4072					
4073					
4074					
4075					
4076					
4077					

SOFTWARE PARAMETER CODING SECTION

4079
4080
4081
4082
4083
4084
4085
4086
4087
4088
4089
4090
4091 026272
4092
4101
4102
4103 026274
4104
4105
4112
4113

.SBTTL SOFTWARE PARAMETER CODING SECTION

```
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
;/ THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
;/ THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
;/ MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
;/ INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
;/ MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
;/ WITH THE OPERATOR.
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
```

BGNSFT

ENDSFT

SOFTWARE PARAMETER CODING SECTION

```
4115
4116 026274          $PATCH::
4117 026274          .BLKW   50
4118
4125
4126 026414          LASTAD
         026420      L$LAST::
4127 026420          ENDMOD
4128
4129
```

SOFTWARE PARAMETER CODING SECTION

4131			
4132			
4145			
4146	026420	BGNSETUP	1
4147	026420	BGNPTAB	
4148	025424 177000	.WORD	177000
4149	026426 000300	.WORD	300
4150	026430 004000	.WORD	4000
4151	026432 000001	.WORD	1
4152	026434	ENDPTAB	
4153	026434	ENDSETUP	
4154			
4155			
4156			
4157			
4158			
4159	000001	.END	

SYMBOL TABLE

ABORT	022660	C#CVEC=	000036	EM0002	015115	GENER1	013274	I#DU	=	000041
ADDR	002354	C#DCLN=	000044	EM0003	015160	GENEX	013432	I#HRD	=	000041
ADDRES	026072	C#DODU=	000051	EM0004	015246	GENINC	013424	I#INIT=	000041	
ADR	= 000020	C#DRPT=	000024	EM0006	015275	GENISH	013302	I#MOD	=	000041
ASSEMB=	000010	C#DU	= 000053	EM0007	015336	GENOUT	024032	I#MSG	=	000041
BAD	012366	C#EDIT=	000003	EM0011	015416	GENRAN	013304	I#PROT=	000040	
BDDAT	002360	C#ERDF=	000055	EM0012	015470	GENROT	013260	I#PTAB=	000041	
BDRGEN	023626	C#ERHR=	000056	EM0013	015552	GENRO	013246	I#PWR	=	000041
BDROKC	024052	C#ERRO=	000060	EM0014	015716	GENR1	013236	I#RPT	=	000041
BGNIXA	025210	C#ERSF=	000054	EM0015	015767	GENSEL	013170	I#SEG	=	000041
BIT0	= 000001	C#ERSO=	000057	EM0016	016063	GENO	013210	I#SETU=	000041	
BIT00	= 000001	C#ESCA=	000010	EM0017	016156	GEN1	013214	I#SFT	=	000041
BIT01	= 000002	C#ESEG=	000005	EM0022	016243	GEN25	013230	I#SRV	=	000041
BIT02	= 000004	C#ESUB=	000003	EM0023	016326	GEN52	013222	I#SUB	=	000041
BIT03	= 000010	C#ETST=	000001	EM0024	016402	GETPRM	022456	I#TST	=	000041
BIT04	= 000020	C#EXIT=	000032	EM0027	016431	GOOD	002264	J#JMP	=	000167
BIT05	= 000040	C#GETB=	000026	EM0031	016503	GOOD0	002266	KB1.2	=	013224 G
BIT06	= 000100	C#GETW=	000027	EM^032	016577	GOOD1	002270	KB56	=	000174 G
BIT07	= 000200	C#GMAN=	000043	EM0033	015646	GOOD10	002300	KB64	=	000154 G
BIT08	= 000400	C#GPHR=	000042	EM0035	016663	GOOD12	002302	KB68	=	000146 G
BIT09	= 001000	C#GPL0=	000030	EM0036	016755	GOOD14	002304	KB72	=	000141 G
BIT1	= 000002	C#GPRI=	000040	END	022724	GOOD16	002306	KIND		012376
BIT10	= 002000	C#INIT=	000011	ERRBLK	002226	GOOD2	002272	KMTLVL		012446
BIT11	= 004000	C#INLP=	000020	ERRCNT	002234	GOOD4	002274	KMVCSSR		012450
BIT12	= 010000	C#MANI=	000050	ERRMSG	002224	GOOD6	002276	KMVLVL		012436
BIT13	= 020000	C#MEM	= 000031	ERRNBR	002222	G#CNT0=	000200	KMVP02		012452
BIT14	= 040000	C#MSG	= 000023	ERRTYP	002220	G#DELM=	000372	KMVP04		012454
BIT15	= 100000	C#OPEN=	000034	EVL	= 000004	G#DISP=	000003	KMVP06		012456
BIT2	= 000004	C#PNTB=	000014	EXADDR	012362	G#EXCP=	000400	KMVP10		012460
BIT3	= 000010	C#PNTF=	000017	E#END	= 002100	G#HILI=	000002	KMVP12		012462
BIT4	= 000020	C#PNTS=	000016	E#LOAD=	000035	G#LULI=	000001	KMVP14		012464
BIT5	= 000040	C#PNTX=	000015	FLAG	002256	G#NO	= 000000	KMVP16		012466
BIT6	= 000100	C#QIO	= 000377	FTIME	002250	G#OFFS=	000400	KMVV00		012434
BIT7	= 000200	C#RDBU=	000007	F#AU	= 000015	G#OFFSI=	000376	KMVV02		012442
BIT8	= 000400	C#REFG=	000047	F#AUTO=	000020	G#PRMA=	000001	KMVV04		012440
BIT9	= 001000	C#RESE=	000033	F#BGN	= 000040	G#PRMD=	000002	KMVV06		012444
BOE	= 000400	C#REVI=	000003	F#CLEA=	000007	G#PRML=	000000	KMV11A		002000 G
BSELO	012370	C#RFLA=	000021	F#DU	= 000016	G#RADA=	000140	LENGTH		012410
BSEL1	002332	C#RPT	= 000025	F#END	= 000041	G#RADB=	000000	LOCK		002230
CBSELO	013644	C#SEFG=	000046	F#HARD=	000004	G#RADD=	000040	LOE	=	040000 G
CHANEL	012400	C#SPRI=	000041	F#HW	= 000013	G#RADL=	000120	LOGDEV		002242
CHKMAX	012732	C#SVEC=	000037	F#INIT=	000006	G#RADO=	000020	LOKFLG		012430
CKALL	013702	C#TPRI=	000013	F#JMP	= 000050	G#XFER=	000004	LOOP		012470
CKREG	014150	DATA	012372	F#MOD	= 000000	G#YES	= 000010	LOOPBK		026214
CKSELO	013612	DATA1	= 052525	F#MSG	= 000011	HELP	= 000000	LOT	=	000010 G
CLRKMV	014372	DATA2	= 125252	F#PROT=	000021	HOE	= 100000	L#ACP		002110 G
COUNT	002350	DELCT1	002260	F#PWR	= 000017	IBE	= 010000	L#APT		002036 G
C#AU	= 000052	DELCT2	002262	F#RPT	= 000012	IDU	= 000040	L#AU		023104 G
C#AUTO=	000061	DFPTBL	002154	F#SEG	= 000003	IER	= 020000	L#AUT		002070 G
C#BRK	= 000022	DIAGMC=	000000	F#SOFT=	000005	INIFLG	012426	L#AUTO		022726 G
C#BSEG=	000004	DROPD	023052	F#SRV	= 000010	INTFLG	012364	L#CCP		002106 G
C#BSUB=	000002	EF.CON=	000036	F#SUB	= 000002	INTTX	024110	L#CLEA		023020 G
C#CEFG=	000045	EF.NEW=	000035	F#SW	= 000014	ISR	= 000100	L#CO		002032 G
C#CLCK=	000062	EF.PWR=	000034	F#TEST=	000001	IXE	= 004000	L#DEPO		002011 G
C#CLEA=	000012	EF.RES=	000037	GDDAT	002356	I#AU	= 000041	L#DESC		002164 G
C#CLOS=	000035	EF.STA=	000040	GOREV	012424	I#AUTO=	000041	L#DESP		002076 G
C#CLP1=	000006	EM0001	015021	GENER	013152	I#CLN	= 000041	L#DEVP		002060 G

SYMBOL TABLE

L\$DISP	002132	G	L10016	022240	0\$BGNS=	000000	RXEND	024504	T\$NSO	=	000000	
L\$DLY	002116	G	L10017	022724	0\$DU	=	000001	RXERR	024366	T\$NS1	=	000005
L\$DTP	002040	G	L10020	023016	0\$ERRT=	000000	RXLCK	024774	T\$PCNT=	000000		
L\$DTYP	002034	G	L10021	023022	0\$GNSW=	000001	RXLEND	025150	T\$PTAB=	010037		
L\$DU	023024	G	L10022	023102	0\$POIN=	000001	RXLERR	025032	T\$PTHV=	000001		
L\$DUT	002072	G	L10023	023104	0\$SETU=	000001	SAVE4	002252	T\$PTNU=	000001		
L\$DVTY	012672	G	L10024	023222	PADFLT	021506	SAVE6	002254	T\$SAVL=	177777		
L\$EF	002052	G	L10025	023272	PBSELO	021544	SAVPC	002244	T\$SEGL=	177777		
L\$ENVI	002044	G	L10026	023606	PFRAME	021704	SAVPC1	002344	T\$SIZE=	000006		
L\$ERRT	002220	G	L10027	024052	PMODEM	021774	SAVREG	013452	T\$SLBN=	000000		
L\$ETP	002102	G	L10030	024504	PNT	=	001000	SAVSTA	002346	T\$TAGL=	177777	
L\$EXP1	002046	G	L10031	025150	PRALL	020716	SEL0	002310	T\$TAGN=	010041		
L\$EXP4	002064	G	L10032	025642	PRAMEF	022120	SEL1	002312	T\$TEMP=	000000		
L\$EXP5	002066	G	L10033	026022	PRI	=	002000	SEL10	002322	T\$TEST=	000010	
L\$HARD	026026	G	L10034	026072	PRINT	020656	SEL12	002324	T\$TSTM=	177777		
L\$HIME	002120	G	L10035	026274	PRIPTY	026160	SEL14	002326	T\$TSTS=	000001		
L\$HPCP	002016	G	L10036	026424	PRI00	=	000000	SEL16	002330	T\$AU	=	010023
L\$HPTP	002022	G	L10040	026434	PRI01	=	000040	SEL2	002314	T\$AUT=	010020	
L\$HW	002154	G	MAINM1	014474	PRI02	=	000100	SEL4	002316	T\$CLE=	010021	
L\$ICP	002104	G	MAINT0=	054000	PRI03	=	000140	SEL6	002320	T\$DAT=	010040	
L\$INIT	022242	G	MAINT1=	044000	PRI04	=	000200	SETUP	022374	T\$DU	=	010022
L\$LADP	002026	G	MAXERR	002232	PRI05	=	000240	SSTACK	012672	T\$HAR=	010034	
L\$LAST	026420	G	MAXPRI=	000300	PRI06	=	000300	STAERR	012416	T\$HW	=	010001
L\$LOAD	002100	G	MBSELO	017612	PRI07	=	000340	SVCGBL=	000000	T\$INI=	010017	
L\$LUN	002074	G	MCLR	=	PRREG	021226	SVCINS=	177777	T\$MSG=	010015		
L\$MREV	002050	G	MFRAM1	020006	PRSELO	020616	SVCSUB=	177777	T\$PC	=	000001	
L\$NAME	002000	G	MFRAM2	020064	PRSTER	022154	SVCTAG=	177777	T\$PRO=	010000		
L\$PRIO	002042	G	MINT	017556	PRT11V	021644	SVCTS	=	177777	T\$PTA=	010037	
L\$PROT	002122	G	MLOOP	020537	PSTACK	002246	SVCTST=	177777	T\$RPT=	010016		
L\$PRT	002112	G	MODEM1	020243	PVECT	021604	S\$LSYM=	010000	T\$SOF=	010035		
L\$REPP	002062	G	MODEM2	020307	QV.FLG	012431	TFM36	014733	T\$TES=	010033		
L\$REV	002010	G	MODEM3	020346	RANCLC	013404	TIM	014716	T1	023106	G	
L\$RPT	022234	G	MODEM4	020411	RANEX	002342	TSPEED	012406	T2	023224	G	
L\$SOFT	026274	G	MODEND	026022	RANGEN	013324	TSTERR	013074	T3	023274	G	
L\$SPC	002056	G	MODSIG	025702	RANMTA	002340	TSTNUB	014556	T4	023610	G	
L\$SPCP	002020	G	MRAHEF	020466	RANSEC	013410	TTABLE	002362	T5	024054	G	
L\$SPTP	002024	G	MREG0	017116	RANSEL	002336	TXABGN	025240	T6	024506	G	
L\$STA	002030	G	MREG10	017336	RANST	002334	TXATAR	025232	T7	025152	G	
L\$SW	002236		MREG12	017402	RAN1	013336	TXDATA	012402	T8	025644	G	
L\$TEST	002114	G	MREG14	017446	RAN2	013354	TXLBGN	024546	UAM	=	000200	G
L\$TIML	002014	G	MREG16	017512	RAN4	013412	TXLTAR	024540	UNIT	012422		
L\$UIT	002240		MREG2	017162	READ	014632	TXSTAR	024114	UUT	012432		
L\$UNIT	002012	G	MREG4	017226	REGADR	012472	T\$ARGC=	000001	VECT	012374		
L10001	002164		MREG6	017272	REVPRO	023234	T\$CODE=	003032	VECTOR	026124		
L10002	020654		MSELO	017050	ROMMAP	023106	T\$ERRN=	000045	WAIT1	012722		
L10003	020714		MSTER1	020146	RSTREG	013532	T\$EXCP=	000000	WAIT2	012702		
L10004	021224		MSTER2	020200	RTABLE	006362	T\$FLAG=	000040	WRDCNT	012420		
L10005	021504		MT11V	017730	RTCLK	023304	T\$FREE=	026434	WRITE	014604		
L10006	021542		MVECT	017654	RUNIN	022666	T\$GMAN=	000000	X\$ALWA=	000000		
L10007	021602		NERRS	013024	RXACK	025466	T\$HILI=	000001	X\$FALS=	000040		
L10010	021642		NEXT	022402	RXAEND	025642	T\$LAST=	000001	X\$OFFS=	000400		
L10011	021702		NUB	012412	RXAERR	025524	T\$LOLI=	000000	X\$TRUE=	000020		
L10012	021772		NUMBER	002352	RXCNT	012414	T\$LSYM=	010000	\$LSTIN=	000000		
L10013	022116		0\$APTS=	000000	RXCK	024344	T\$LTNO=	000010	\$LSTTA=	000000		
L10014	022152		0\$AU	=	RXDATA	012404	T\$NEST=	177777	\$PATCH	026274	G	
L10015	022232		0\$BGNR=	000000								

SYMBOL TABLE

. ABS. 026434 000
 000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 28944 WORDS (114 PAGES)

DYNAMIC MEMORY: 20060 WORDS (77 PAGES)

ELAPSED TIME: 00:04:12

CNKMB80.BIC,CNKMB80.SEQ/-SP=SVC34.MLB/ML,CNKMB80.MAC