

# VT62, DUP11

DIAG  
CZDPGB0

AH-A887B-MC

MAY 1979

COPYRIGHT 77-79

**digital**

FICHE 1 OF 1

MADE IN USA

This microfiche card contains a grid of frames. The frames on the left side contain technical diagrams, including what appears to be a block diagram of a system with various components and connections. The frames on the right side contain data, likely in the form of tables or lists of parameters, organized in columns and rows. The text and diagrams are very small and difficult to read due to the low resolution of the scan.

2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37

000000

.REPT 0

IDENTIFICATION

PRODUCT CODE: AC-A886B-MC  
PRODUCT NAME: CZDPGB0 VT62/DUP11 DIAG  
DATE: 1-JAN-79  
MAINTAINER: DIAGNOSTIC ENGINEERING  
AUTHOR: MICHAEL DENSMORE

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 WITHIN.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

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

(COPYRIGHT (C) 1977,1978,1979 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.



38  
39  
40  
41  
42  
43  
44

REVISION HISTORY

REVISION  
A

BY:  
M. DENSMORE ORIGINAL RELEASE

45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82

TABLE OF CONTENTS

-----

1. GENERAL	PG. 4
1.1 ABSTRACT	PG. 4
1.2 DEFINITIONS	PG. 4
1.3 DESIGN NOTES	PG. 4
1.4 DDCMP	PG. 5
2. EQUIPMENT REQUIREMENTS	PG. 5
3. LOAD PROCEDURE	PG. 5
4. STARTING AND RUNNING THE PROGRAM	PG. 5
4.1 START	PG. 6
4.2 RESTART	PG. 8
4.3 CONTINUE	PG. 8
4.4 ZERO FLAGS	PG. 9
4.5 ADD UNITS	PG. 9
4.6 DROP UNITS	PG. 9
4.7 DISPLAY PTABLE	PG. 9
4.8 SWITCH OPTIONS	PG. 9
5. ERROR AND INFORMATIONAL MESSAGES	PG. 10
6. RESTRICTIONS	PG. 11
7. MISC.	PG. 11
8. TEST DESCRIPTIONS	PG. 12
8.1 MICRODIAGNOSTIC	PG. 12
8.2 LOOPBACK	PG. 12
9. INDEX	PG. 13

83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135

1.0  
1.1

GENERAL  
ABSTRACT

Z2-CZDPC RUNS THE SELFTEST FOR THE VT62 TERMINAL AND EXERCISES THE LINK BETWEEN THE TERMINAL AND THE HOST. THIS PARTICULAR VERSION SUPPORTS MULTIDROP, FULL DUPLEX COMMUNICATION VIA THE DUP-11 COMMUNICATION INTERFACE. OTHER VERSIONS (MAINDEC-11-DZDZC AND MAINDEC-11-DZDVP) SUPPORT THE DZ-11 AND DV-11 COMMUNICATION INTERFACES RESPECTIVELY. THE PROGRAM IS XXDP COMPATIBLE AND SUPPORTS SEQUENTIAL TESTING OF UP TO 8 VT62'S. PLEASE READ ALL SECTIONS OF THIS DOCUMENT CAREFULLY BEFORE USING THE PROGRAM. PAY PARTICULAR ATTENTION TO SECTIONS 3.0 THROUGH 6.0

1.2

DEFINITIONS

MODEM - EQUIPMENT, ALSO KNOWN AS A "DATASET", WHICH PERFORMS THE MODULATION/DEMULATION OF DATA SIGNALS. THIS UNIT PROVIDES TIMING FOR SYNCHRONOUS COMMUNICATION DEVICES.

MODEM ELIMINATOR - EQUIPMENT WHICH CONNECTS TWO SYNCHRONOUS DEVICES USING A CABLE RATHER THAN TELEPHONE VOICE LINES. THE ELIMINATOR HAS A CLOCK FOR TIMING.

DDCMP PROTOCOL - DEC'S COMMUNICATION LINE PROTOCOL.

UUT - UNIT UNDER TEST.

PTABLE - A TABLE OF INFORMATION, WITH ENTRIES FOR EACH UNIT UNDER TEST, THAT DESCRIBES THE HARDWARE (BUS ADDRESS, VECTOR, ETC.)

DUP - SINGLE LINE, SYNCHRONOUS COMMUNICATION INTERFACE.

DZ - EIGHT LINE, MULTIPLEXED, SYNCHRONOUS COMMUNICATION INTERFACE.

DV - SIXTEEN LINE, MULTIPLEXED, SYNCHRONOUS COMMUNICATION INTERFACE.

CRC CHECKSUM - CHECKSUM USED IN DEC NETWORK PROTOCOL (CYCLIC REDUNDANCY CHECK).

MICRODIAGNOSTIC - THE MICROCODE IN THE VT62 THAT PERFORMS THE DIAGNOSTIC CHECKOUT OF THE HARDWARE. (ALSO KNOWN AS "SELF TEST")

SELF TEST - SEE "MICRODIAGNOSTIC".

1.3

DESIGN NOTES

THIS PROGRAM WAS DESIGNED AND IMPLEMENTED USING STRUCTURED TECHNIQUES. THE PROCEDURE CAUSED THE PROGRAM TO HAVE SOME OPERATIONAL CHARACTERISTICS WHICH ARE OF INTEREST TO THE USER. THESE CHARACTERISTICS ARE DISCUSSED IN THIS SECTION. THIS DISCUSSION IS NOT VITAL TO LEARNING HOW TO OPERATE THE PROGRAM HOWEVER.

THE MOST IMPORTANT OPERATIONAL CHARACTERISTIC OF STRUCTURED DESIGN, AS IT AFFECTS THE USER, IS HOW ERRORS ARE REPORTED. THE TOP-DOWN DESIGN OF THE PROGRAM RESULTS IN A BOTTOM-UP REPORT OF ERROR INFORMATION. SEVERAL LINES OF ERROR INFORMATION ARE PRINTED AS EACH PROGRAM SECTION REPORTS ITS LEVEL OF INFORMATION. AN EXAMPLE OF THIS TYPE OF ERROR LOG WOULD BE A DUP ERROR. THE FIRST ERROR MESSAGE WOULD GIVE SPECIFIC DEVICE INFORMATION SUCH AS REGISTER CONTENTS AND THEN RETURN CONTROL TO THE NEXT LEVEL. AT THE NEXT LEVEL, THE LOGICAL DEVICE NUMBER OR DEVICE SERIAL NUMBER WOULD BE REPORTED. AT THE LEVEL ABOVE THAT, STILL MORE GENERAL INFORMATION WOULD BE GIVEN. THE REPORTS CONTINUE UNTIL THE HIGHEST LEVEL IS REACHED.

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  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189

COMMENT FIELDS IN THE LISTING WITH SQUARE BRACKETS ([ ]) REFER TO FUNCTIONAL BLOCKS IN THE DESIGN AND HAVE NO SIGNIFICANCE TO THE USER. THEY ARE RETAINED FOR USE BY PROGRAM MAINTAINERS.

1.4

DDCMP

THIS SECTION GIVES A VERY BRIEF DESCRIPTION OF DDCMP AS USED BY THIS DIAGNOSTIC. THE DESCRIPTION IS BY NO MEANS COMPLETE; IT IS INTENDED TO HELP THE USER UNDERSTAND ERROR REPORTS.

THERE ARE THREE TYPES OF DDCMP MESSAGES: CONTROL, DATA AND BOOTSTRAP. THE BOOTSTRAP MESSAGE IS NOT USED BY THIS DIAGNOSTIC. THE DATA MESSAGES CONSIST OF A 6-BYTE HEADER, A 2-BYTE CRC FOR THE HEADER, THE DATA ITSELF AND A 2-BYTE CRC FOR THE DATA. CONTROL MESSAGES CONSIST OF THE HEADER (AND ITS CRC) ONLY. ALL CONTROL MESSAGES HAVE AN 'ENQ' CHARACTER AS THE FIRST BYTE. ALL DATA MESSAGES ARE NUMBERED SO THAT EACH STATION ON THE LINK CAN KEEP TRACK OF WHICH MESSAGES ARE TRANSMITTED PROPERLY. ALL MESSAGES HAVE A STATION ADDRESS IN THE HEADER SO THAT MESSAGES ARE DIRECTED TO AND FROM THE PROPER STATIONS ON A MULTIDROP LINE.

THERE ARE FOUR CONTROL MESSAGES USED BY THIS DIAGNOSTIC: ACK, NAK, STRT AND STACK. ACK IS AN ACKNOWLEDGEMENT THAT THE LAST MESSAGE SENT WAS PROPERLY RECEIVED. NORMALLY, MESSAGES ARE ACKNOWLEDGED BY SENDING A DATA MESSAGE WITH THE NUMBER OF THE LAST MESSAGE IN THE HEADER. ACK'S ARE USED WHEN THERE IS NO DATA MESSAGE TO SEND BACK. NAK IS A NEGATIVE ACKNOWLEDGEMENT FOR THE LAST MESSAGE SENT AND INDICATES THE REASON FOR THE ERROR. STRT IS USED TO INITIATE COMMUNICATION BETWEEN TWO STATIONS. THE RECEIVING STATION SETS ITSELF UP AND ACKNOWLEDGES THE STRT WITH A STACK, OR START ACKNOWLEDGE. THE STRT-STACK SEQUENCE IS USED BEFORE AND AFTER EACH TEST. ALL MESSAGES ARE REQUIRED TO BE ACKNOWLEDGED WITH EITHER A DATA MESSAGE OR AN ACK.

2.0

EQUIPMENT REQUIREMENTS

THIS PROGRAM REQUIRES A PDP-11 WITH A MINIMUM OF 16K CORE AND A CONSOLE TERMINAL. A CLOCK OPTION IS NOT REQUIRED, BUT ONLY ROUGH TIMING IS AVAILABLE WITHOUT ONE. THE PDP-11 CONSOLE SWITCH REGISTER IS NOT USED. ONE OR MORE DUP-11'S MUST BE CONNECTED TO THE TERMINALS UNDER TEST VIA COMMUNICATION LINES THROUGH EITHER MODEM'S OR MODEM ELIMINATORS. THE VT62'S MUST BE IN SYNCHRONOUS MODE AND ON-LINE. THE TERMINAL ADDRESS SWITCHES ON THE CONTROL BOARD MUST BE SET TO MATCH THE TERMINAL ADDRESS ASSIGNED TO THE LOGICAL UNIT (SEE SECTION 4.0).

3.0

LOADING PROCEDURE

THIS PROGRAM MAY BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER. IT MAY ALSO BE LOADED FROM PAPER TAPE OR DISK USING XXDP PROCEDURES.

4.0

STARTING AND RUNNING THE PROGRAM

THE STARTING ADDRESS FOR THE PROGRAM IS 200. THE PROGRAM WILL FIRST ASK IF A CLOCK (KW11-L OR KW11-P) IS AVAILABLE. IT WILL THEN ASK IF A LINE

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  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243

PRINTER IS AVAILABLE. IF A LINE PRINTER IS SPECIFIED, THE PROGRAM WILL ASK FOR ITS ADDRESS (DEFAULT = 177514). THE PROGRAM ALSO ASKS FOR MEMORY SIZE, Q-BUS OPTION AND LINE FREQUENCY. IF NEITHER CLOCK IS SPECIFIED, THE PROGRAM WILL ASK THE USER TO TYPE TWO CHARACTERS, FIVE SECONDS APART, IN ORDER TO PROVIDE ROUGH TIMING.

THE PROGRAM WILL THEN PROMPT THE USER BY TYPING A ">". NOW THE USER MAY TYPE ONE OF SEVEN COMMANDS: STA(RT), RES(TART), CON(TINUE), ZFL(AG), ADD, DRO(P) OR DIS(PLAY), WHERE THE LETTERS IN PARENTHESIS NEED NOT BE TYPED. IF A TYPING MISTAKE IS MADE, THE RUBOUT KEY IS USED TO DELETE THE LAST CHARACTER(S) TYPED. THE PROGRAM WILL TYPE A MESSAGE IF THE COMMAND IS INVALID. EACH OF THESE COMMANDS IS EXPLAINED IN THE SECTIONS BELOW. SWITCHES ARE USED TO SET SOFTWARE FLAGS SUCH AS "LOOP ON ERROR". THE FLAGS USED BY THIS PROGRAM ARE DESCRIBED IN SECTION 4.8.

4.1

START

THE START COMMAND (STA) ZEROS ALL FLAGS (PRINT ALL ERROR MESSAGES AND DO NOT LOOP WITHIN TESTS) AND INITIATES THE OPERATOR DIALOGUE. AT THIS POINT, THE DEVICE TABLE (PTABLE) IS EMPTY AND THE OPERATOR MUST USE THE DIALOGUE TO DESCRIBE ALL UUT'S AS SHOWN BELOW. THE NUMBERS IN SQUARE BRACKETS, [], RELATE THE TYPEOUT TO THE EXPLANATION BELOW AND ARE NOT TYPED IN ACTUAL OPERATION. THE PROCEDURES FOR HANDLING TYPING ERRORS ARE ALSO DESCRIBED BELOW. CR REFERS TO THE RETURN KEY (CARRIAGE RETURN).

DIALOGUE

- # UNITS ? (DECIMAL) [1]
- DUP ADDRESS: ? (OCTAL) NNNNNN [2]
- DUP VECTOR: ? (OCTAL) NNN [3]
- DUP BUS PRIORITY: ? (OCTAL) N [4]
- TERMINAL ADDRESS: ? (OCTAL) N [5]  
CHANGE S.W. ? (Y OR N) [6]
- HOW MANY INITIATE TRIES BEFORE ABORT ? (OCTAL) 10 [7]
- USE ALL ZERO'S FOR LOOPBACK DATA ? (Y OR N) Y [8]

EXPLANATION

- [1] TYPE THE TOTAL NO. OF UUT'S IN DECIMAL. QUESTIONS 2 THROUGH 5 WILL BE REPEATED UNTIL ALL UUT'S HAVE BEEN DESCRIBED. THE USER DOES NOT HAVE TO ANSWER THESE QUESTIONS FOR EACH DUP HOWEVER, IF THERE ARE MULTIPLE UUT'S PER DUP, THE EXPLANATION FOR QUESTION 5 SHOWS HOW TO ENTER MULTIPLE UUT'S.
- [2] TYPE THE BUS ADDRESS OF THE DUP. THIS MUST BE AN EVEN OCTAL NUMBER NOT GREATER THAN 177776. THE USER MAY TYPE CR TO USE

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  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297

- THE DEFAULT VALUE: NNNNNN.
- [3] TYPE THE VECTOR ADDRESS FOR THE DUP. THIS MUST BE AN EVEN OCTAL NUMBER NOT GREATER THAN 376. THE USER MAY TYPE CR TO USE THE DEFAULT VALUE: NNN.
  - [4] TYPE THE BUS PRIORITY LEVEL OF THE DUP. THIS MUST BE A DIGIT FROM 0 TO 7. AGAIN, THE USER MAY TYPE CR TO GET THE DEFAULT VALUE: N.
  - [5] TYPE THE TERMINAL ADDRESS FOR EACH TERMINAL ATTACHED TO THE DUP. SEPARATE EACH ADDRESS WITH A COMMA. THIS IS THE ADDRESS SELECTED BY THE SWITCHES ON THE I/O BOARD IN THE TERMINAL AND IS AN OCTAL NUMBER BETWEEN 0 AND 377. THE USER MAY TYPE CR TO GET THE DEFAULT VALUE: N.
  - [6] IF THE USER WISHES TO CHANGE EITHER OF THE TWO SOFTWARE PARAMETERS DESCRIBED FOR QUESTIONS 7 OR 8, HE MUST ANSWER THIS QUESTION BY TYPING 'Y'. IF THE USER WISHES TO BYPASS QUESTIONS 7 AND 8, HE TYPES 'N'.
  - [7] TYPE THE NUMBER OF TIMES THE PROGRAM IS TO ATTEMPT TO INITIALIZE A DUP BEFORE ABORTING THE UNIT. THE USER MAY TYPE ANY OCTAL NUMBER BETWEEN 0 AND 177777. TYPE CR TO DEFAULT TO 10 (OCTAL) ATTEMPTS.
  - [8] IF THE USER TYPES 'Y', THE LOOPBACK DATA WILL BE 255 ZERO BYTES. IF THE USER TYPES 'N', THE LOOPBACK DATA WILL BE A PATTERN OF ALTERNATING 1 AND 0 BITS. SEE SECTION 8.2 FOR DETAILS ON THE LOOPBACK TEST.

EXAMPLE OF DIALOGUE

ASSUME THE FOLLOWING HARDWARE CONFIGURATION: A DUP (160000,300,3) WITH TWO TERMINALS (ADDRESSES 0 AND 1), A SECOND DUP (160010,320,3) WITH A SINGLE TERMINAL (ADDRESS 5) AND A THIRD DUP (170000,340,4) WITH TWO TERMINALS (ADDRESSES 1 AND 2). THE USER WOULD USE THE FOLLOWING DIALOGUE TO SET UP THE PROGRAM:

```
>STA
# UNITS ? (DECIMAL) 5
DUP ADDRESS: ? (OCTAL) 160000
DUP VECTOR: ? (OCTAL) 310 300
DUP BUS PRIORITY: ? (OCTAL) 4 3
TERMINAL ADDRESS: ? (OCTAL) 0 0,1
DUP ADDRESS: ? (OCTAL) 160000 160010
DUP VECTOR: ? (OCTAL) 300 320
DUP BUS PRIORITY: ? (OCTAL) 3
TERMINAL ADDRESS: ? (OCTAL) 1 5
```



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  
343  
344  
345  
346  
347  
348  
349  
350  
351

DUP ADDRESS: ? (OCTAL) 160010 170000

DUP VECTOR: ? (OCTAL) 320 340

DUP BUS PRIORITY: ? (OCTAL) 3 5

TERMINAL ADDRESS: ? (OCTAL) 5 1,2

CHANGE S. W. ? (Y OR N) N

(NOTE WHERE CR WAS USED TO ACCEPT THE DEFAULT)

#### TYPING ERRORS

- (1) THE RUBOUT KEY IS USED TO DELETE ANY CHARACTERS WHICH ARE IN ERROR.
- (2) IF THE RESPONSE IS NOT WITHIN LIMITS OR IS IN THE WRONG FORMAT, THE PROGRAM WILL NOTIFY THE USER AND ASK THE QUESTION AGAIN.
- (3) WHEN ENTERING MULTIPLE TERMINAL ADDRESSES FOR A SINGLE DUP, THE USER MAY NEGLECT TO TYPE ONE OR MORE ADDRESSES. IN SUCH A CASE, THE USER MERELY ENTERS THE SAME DUP ADDRESS, VECTOR AND PRIORITY IN THE NEXT SERIES OF QUESTIONS AND THEN ENTERS THE REMAINING TERMINAL ADDRESSES. REMEMBER; THE SERIES OF QUESTIONS (2 THROUGH 5) WILL BE RE-ASKED UNTIL ALL UUT'S HAVE BEEN ACCOUNTED FOR.

#### 4.2

##### RESTART

THE RESTART COMMAND (RES) STARTS THE PROGRAM WITHOUT CLEARING THE TABLE. ALL FLAGS REVERT TO THEIR INITIAL STATE UNLESS THE FLAG SWITCH IS USED TO CHANGE THEM (SECTION 4.8). THE OPERATOR DIALOGUE DESCRIBED IN 4.1 DOES NOT TAKE PLACE UNLESS REQUESTED. WHEN THIS COMMAND IS USED, THE PROGRAM WILL ASK IF THE USER WISHES TO CHANGE THE HARDWARE TABLE AND IF HE WISHES TO CHANGE THE SOFTWARE TABLE. IF THE USER ANSWERS YES (BY TYPING 'Y') TO THE HARDWARE TABLE QUESTION, THAT PORTION OF THE DIALOGUE DESCRIBED IN SECTION 4.1 BEGINNING WITH QUESTION 1 ENDING WITH QUESTION 5 WILL TAKE PLACE. IF THE USER ANSWERS YES TO THE SOFTWARE TABLE QUESTION, HE WILL BE ASKED QUESTIONS 7 AND 8.

THE MOST COMMON USE OF THIS COMMAND IS TO CHANGE RUN-TIME FLAGS OR TO ADD/DROP UNITS. THESE FLAGS ARE DESCRIBED IN SECTION 4.8 - SWITCH OPTIONS. TYPING A '^C' WILL FORCE THE SUPERVISOR TO ISSUE A PROMPT CHARACTER ('>') SO THAT THE USER MAY USE THE RESTART COMMAND FOR THIS PURPOSE. NOTE. IF THE USER USES THE START COMMAND AFTER A ^C, HE WILL HAVE TO REENTER ALL DEVICE INFORMATION!

#### 4.3

##### CONTINUE

THE CONTINUE COMMAND WILL RESTART THE PROGRAM AT THE TEST WHICH WAS INTERRUPTED. UNLIKE THE RESTART COMMAND, THIS COMMAND DOES NOT FORCE FLAGS BACK TO THE INITIAL STATE. THE FLAGS WILL REMAIN AS SET UP DURING THE LAST START OR RESTART. THE USER HAS THE SAME DIALOGUE AND FLAG

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  
401  
402  
403  
404  
405

OPTIONS AS DESCRIBED FOR THE RESTART COMMAND.

4.4 ZERO FLAGS  
THIS COMMAND WILL ZERO ALL FLAGS (SECTION 4.8) AND TYPE A PROMPT. SECTION 4.8 DESCRIBES THE EFFECTS OF CLEARING ALL FLAGS, BUT ESSENTIALLY, ALL LOOPING AND HALTS WILL BE DISABLED AND ALL ERROR INFORMATION WILL BE PRINTED IN THE EVENT OF AN ERROR.

4.5 ADD UNITS  
WHILE TESTING MULTIPLE UNITS, THE USER MAY WISH TO DROP A PARTICULAR UNIT FROM THE TEST PROCEDURE AND, LATER ON, BEGIN TESTING THAT UNIT AGAIN. IN THIS CASE, THE USER NEED NOT STOP THE DIAGNOSTIC AND REBUILD THE PTABLE. AFTER A UNIT HAS BEEN DROPPED (SEE SECTION 4.6), THE USER MAY ADD THE UNIT BY TYPING '^C' AND USING THE ADD COMMAND AS FOLLOWS:

ADD/UNI:N

WHERE N IS A DECIMAL UNIT NUMBER THAT SPECIFIES THE UNIT TO BE ADDED. UNITS ARE NUMBERED FROM 1 TO 32 ACCORDING TO THE ORDER IN WHICH THEY WERE SPECIFIED FOR THE HARDWARE TABLE. MULTIPLE UNITS MAY BE ADDED BY TYPING EACH UNIT NUMBER SEPARATED BY COLONS (I.E., ADD/UNI:2:4:22) OR BY TYPING THE FIRST AND LAST UNIT NUMBERS OF A CONTIGUOUS GROUP OF UNITS SEPARATED BY A DASH (I.E., ADD/UNI:1-10 WILL ADD THE FIRST TEN UNITS). NOTE: AFTER A START COMMAND, ALL UNITS ARE CONSIDERED ADDED. THERE IS NO NEED TO USE THE ADD COMMAND UNLESS THE UNIT WAS DROPPED.

4.6 DROP UNITS  
TO REMOVE A UNIT FROM TESTING, THE DROP COMMAND (DRO) IS USED. TO USE THIS COMMAND, TYPE '^C' AND DROP THE UNIT(S) USING THE DROP COMMAND AS FOLLOWS:

DRO/UNI:N

WHERE 'N' IS A DECIMAL UNIT NUMBER WHICH SPECIFIES THE UNIT TO BE DROPPED. UNITS ARE NUMBERED FROM 1 TO 32 ACCORDING TO THE ORDER IN WHICH THEY WERE SPECIFIED FOR THE HARDWARE TABLE. MULTIPLE UNITS MAY BE DROPPED BY TYPING EACH UNIT NUMBER SEPARATED BY COLONS (I.E., DRO/UNI:1:3:7) OR BY TYPING THE FIRST AND LAST UNIT NUMBERS OF A CONTIGUOUS GROUP SEPARATED BY A DASH (I.E., DRO/UNI:1-9 CAUSES THE FIRST NINE UNITS TO BE DROPPED). AFTER THE PROGRAM IS RESTARTED, THE SPECIFIED UNIT(S) WILL NOT BE INCLUDED FOR TESTING.

4.7 DISPLAY PTABLE  
WHEN THE DISPLAY COMMAND IS TYPED, THE ENTIRE CONTENTS OF THE PTABLE WILL BE PRINTED. UNITS THAT HAVE BEEN DROPPED WILL BE SO MARKED. THIS COMMAND IS USEFUL FOR VERIFYING THE PTABLE CONTENTS.

4.8 SWITCH OPTIONS  
THERE ARE THREE SWITCH OPTIONS AVAILABLE FOR USE WITH THE COMMANDS: /TES:, /PAS: AND /FLA:. THE TEST SWITCH, /TES:, IS USED ONLY WITH THE START COMMAND AND THE RESTART COMMAND. IT IS USED TO SPECIFY THE INDIVIDUAL TEST(S) TO BE RUN IN CASES WHERE ALL TESTS ARE NOT TO BE RUN. FOR EXAMPLE, IF THE USER WISHED TO RUN THE LOOPBACK TEST ONLY, HE WOULD TYPE:

>STA/TES:2

406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459

IF THIS SWITCH IS NOT USED, ALL TESTS WILL BE RUN AS THE DEFAULT CONDITION.

THE PASS SWITCH IS USED TO SPECIFY THE NUMBER OF PASSES TO BE RUN. TO RUN THE LOOPBACK TEST TWENTY TIMES, THE USER WOULD TYPE:

>STA/TES:2/PAS:20

THIS SWITCH MAY BE USED WITH THE START, RESTART AND CONTINUE COMMANDS. THE DEFAULT CONDITION IS ESSENTIALLY AN INFINITE NUMBER OF PASSES.

THE FLAG SWITCH IS USED TO SET FLAGS WHICH CONTROL PROGRAM OPERATION AND ERROR PRINTOUTS. THIS SWITCH MAY BE USED WITH THE CONTINUE COMMAND AS WELL AS THE START COMMAND AND RESTART COMMAND. THE AVAILABLE FLAGS ARE SHOWN IN THE TABLE BELOW.

FLAG	EFFECT IF SET
LOE	LOOP ON ERROR
HLT	HALT ON ERROR
IER	INHIBIT ERROR REPORTS
IBE	INHIBIT BASIC ERROR REPORT*
IXE	INHIBIT EXTENDED ERROR REPORT*
PRI	SEND ALL TYPEOUTS TO THE PRINTER
BOE	BELL ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
PNT	PRINT TEST NUMBER

\*SEE SECTION 5.0

EXAMPLE:

IF THE USER WISHES TO RUN THE PROGRAM WITHOUT ERROR MESSAGES AND HAVE IT LOOP ON ERROR, HE WOULD TYPE:

>RES/FLA:LOE:IER

EXAMPLE:

IF THE USER LATER WANTED TO STOP THE LOOP ON ERROR FEATURE, HE WOULD TYPE:

>CON/FLA:LOE=0

NOTE: THE ERROR REPORTS WOULD STILL BE INHIBITED.

ALL FLAGS ARE DEFAULTED TO ZERO (NOT SET).

5.0

ERROR AND INFORMATIONAL MESSAGES

AS NOTED IN THE DESIGN NOTES (SECTION 1.3), A SINGLE ERROR MAY RESULT IN SEVERAL MESSAGES BEING TYPED. THIS IS DUE TO THE STRUCTURE OF THE PROGRAM. EACH SECTION, IN THE EVENT OF AN ERROR, REPORTS THE INFORMATION AVAILABLE TO IT. THE FINAL MESSAGE FOR EACH ERROR CONTAINS THE UNIT NUMBER (DECIMAL - 1 TO 32 - SEE SECTIONS 4.5 AND 4.6) AND THE PHRASE: "END OF ERROR REPORT".

THERE ARE THREE LEVELS OF ERROR REPORTS. BY USING THE FLAG SWITCHES (SECTION 4.8), ONE OR MORE LEVELS OF REPORT MAY BE INHIBITED. THE FIRST LEVEL CONSISTS OF AN ERROR NUMBER, PC CONTENTS AND ERROR TYPE (HARD, SOFT OR FATAL). THE ERROR NUMBER CAN BE RELATED TO THE TABLE AT THE END OF

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  
513

THIS SECTION. THE PC CONTENTS GIVE THE ADDRESS WHERE THE ERROR REPORT MECHANISM WAS INVOKED. THE NEXT LEVEL (BASIC ERROR REPORT) GIVES AN ENGLISH DESCRIPTION OF THE ERROR. AN EXAMPLE OF THIS TYPE OF MESSAGE IS "HEADER HAS CRC ERROR". THE FINAL LEVEL (EXTENDED ERROR REPORT) GIVES ADDITIONAL DATA CONCERNING THE ERROR, SUCH AS REGISTER CONTENTS.

TO INHIBIT SOME OR ALL ERROR MESSAGES, USE THE FLAGS DESCRIBED IN SECTION 4.8. THE INHIBIT-EXTENDED-ERROR-REPORT FLAG (IXE) INHIBITS THAT LEVEL ONLY. THE INHIBIT-BASIC-ERROR-REPORT FLAG (IBE) INHIBITS BOTH BASIC AND EXTENDED ERROR REPORTS. FINALLY, THE INHIBIT-ERROR-REPORT FLAG (IER)) INHIBITS ALL TYPEOUTS.

TABLE OF ERROR NUMBERS

NO.	ERROR DESCRIPTION
00	CANNOT INITIALIZE DUP
01	REPLY MESSAGE FROM TERM. WAS NOT AN ENQ
02	REPLY FROM TERM. WAS NOT EXPECTED TYPE OF CONTROL MESSAGE
03	TERM. ADDRESS IN RESPONSE WAS NOT CORRECT
04	RESPONSE HAD CRC ERROR IN THE HEADER
20	DUP TRANSMIT ERROR
21	DUP RECEIVE ERROR
22	MESSAGE TO TERMINAL WAS NOT ACK'ED
23	ERROR OCCURED WHILE INITIATING DUP
24	ERROR OCCURED DURING TRANSMISSION
25	DDCMP STRT MESSAGE DID NOT GET A STACK
26	ERROR DURING RECEPTION
30	CANNOT GET CLEAR TO SEND SIGNAL
31	TIMEOUT WHILE WAITING FOR RESPONSE
40	SEE 21
41	SEE 04
42	SEE 03
43	MESSAGE DATA HAD CRC ERROR
50	ERROR OCCURED DURING SELFTEST (TEST #1)
51	ERROR OCCURED DURING LOOPBACK (TEST #2)
52	ERROR IN RECEIVED DATA
70	SELFTEST RESPONSE IN ERROR (NOT RECOGNIZED)
71	TERMINAL MEMORY ERROR
72	TERMINAL CONTROL-ROM ERROR

6.0 RESTRICTIONS  
THIS DIAGNOSTIC DOES NOT ATTEMPT TO TEST THE DUP11. IF A LARGE NUMBER OF COMM ERRORS OCCUR, OR IF ATTEMPTS TO SEND OR RECIEVE DATA TO AND FROM THE TERMINAL FAIL; USE THE SPECIFIC DUP11 DIAGNOSTICS TO VERIFY THE DUP11.

7.0 MISCELLANEOUS  
PLEASE REPORT ALL PROGRAM PROBLEMS VIA THE AIDS REPORTING SYSTEM. REFER ALL COMMENTS REGARDING DOCUMENTATION TO THE DIAGNOSTIC ENGINEERING

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

TERMINAL GROUP IN MAYNARD, MASS.

8.0 TEST DESCRIPTIONS

8.1 MICRODIAGNOSTIC (TEST #1)

THE MICRODIAGNOSTIC, OR SELFTEST, IS ROM RESIDENT CODE IN THE VT62 THAT IS EXECUTED UPON COMMAND FROM THE HOST COMPUTER OR DURING POWER UP. A FULL DESCRIPTION OF THE TEST IS AVAILABLE IN THE HARDWARE MANUAL FOR THE VT62. ESSENTIALLY, THE TEST VERIFIES THE MICROPROCESSOR, CHECKS ROM MEMORY, EXERCISES RAM MEMORY AND EXERCISES THE SCREEN. THE HOST MAY REQUEST THE RESULTS OF THE TEST. IN ANY CASE, THE RESULTS ARE ALWAYS DISPLAYED ON THE LAST LINE OF THE SCREEN. THE FIRST CHARACTERS ARE A CODED ERROR REPORT FOR THE HOST COMPUTER AND THE REMAINDER OF THE LINE IS AN ENGLISH DESCRIPTION OF THE ERROR STATUS.

8.2 LOOPBACK (TEST #2)

IN THIS TEST, THE TERMINAL IS COMMANDED TO ENTER LOOPBACK MODE AND A SINGLE, 256(DECIMAL) BYTE DATA MESSAGE IS SENT TO THE TERMINAL. THIS DATA MESSAGE WILL HAVE A DDCMP PROTOCOL ENVELOPE. LOOPBACK DATA IS DISPLAYED ON THE VT62'S SCREEN. WHEN THE DATA IS RETURNED, THE TERMINAL IS TAKEN OUT OF LOOPBACK MODE USING A DDCMP "STRT" SEQUENCE. THE CRC CHECKSUMS ARE VALIDATED FOR BOTH THE MESSAGE HEADER AND THE MESSAGE DATA. THE DATA IS THEN CHECKED, BYTE BY BYTE. THUS THERE ARE THREE POSSIBLE ERRORS DETECTED: NO TERMINAL RESPONSE, BAD CRC OR BAD DATA.

INTERPRETATION OF ANY ERROR MESSAGES REQUIRES SOME EDUCATED GUESSWORK BECAUSE OF THE NUMBER OF COMPONENTS IN THE COMMUNICATION LINK. INTERMITTENT ERRORS WOULD GENERALLY INDICATE LINE PROBLEMS UNLESS THE LINE IS PHYSICALLY SHORT OR THE ENVIRONMENT IS RELATIVELY NOISE FREE. IF EITHER OF THESE CONDITIONS IS THE CASE OR IF THE ERROR IS NOT INTERMITTENT, THEN THE TERMINAL I/O BOARD, THE MODEM AND THE DUP ARE SUSPECT. IF THERE IS NO TERMINAL RESPONSE, RUN THE TERMINAL SELF TEST OFF-LINE. THIS WILL SHOW WHETHER THE TERMINAL IS COMPLETELY "DEAD" OR THE PORTIONS OF THE COMMUNICATION LINK MENTIONED ABOVE ARE BROKEN.

THERE ARE TWO DATA PATTERNS AVAILABLE FOR LOOPBACK: ALL ZERO'S AND ALTERNATING 1/0 BITS (I.E., BYTES OF 252,125,252,125,...). THE ZERO PATTERN IS NOISE-SUSCEPTIBLE, BUT APPEARS ON THE SCREEN AS NULLS. HOWEVER THE ALTERNATING BITS PATTERN EXERCISES THE DATA PATHS BETTER. SECTION 4.0 SPECIFIES HOW TO SELECT THE PATTERN.



557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
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

9.0

INDEX

ACK . . . . .	5, 11
ADD COMMAND . . . . .	9
CONTINUE COMMAND . . . . .	8, 10
CRC . . . . .	4-5, 11
DDCMP . . . . .	4-5
DISPLAY COMMAND . . . . .	9
DROP COMMAND . . . . .	9
END . . . . .	5, 11
ERROR NUMBER . . . . .	10-11
ERROR REPORT . . . . .	10-11
FLAGS . . . . .	6, 8-10
LOOPBACK . . . . .	7, 11-12
MULTIPLE UNITS . . . . .	9
MULTIPLE UUT'S . . . . .	6
NAK . . . . .	5
PTABLE . . . . .	4, 6, 9
RESTART COMMAND . . . . .	8-10
STACK . . . . .	5, 11
START COMMAND . . . . .	6, 8-10
STRT . . . . .	5, 11
SWITCH OPTIONS . . . . .	8-9
SWITCHES . . . . .	6
UNIT NUMBER . . . . .	9-10

.ENDR

GOO 000000\*

PRINT ;SPRAC: VERSION 1.1

601	002000		.=2000	
602				
603			HEADER - VT62 ACCEPTANCE TEST	
604				
605	002000		POINTER ALL	
606	002000		BGNMOD HEAD	
607	002000		HEADER C7DPG,B,0,0,0,0	
608	002000	103		.ASCII /C/
609	002001	132		.ASCII /Z/
610	002002	104		.ASCII /D/
611	002003	120		.ASCII /P/
612	002004	107		.ASCII /G/
613	002005	000		.BYTE 0
614	002006	000		.BYTE 0
615	002007	000		.BYTE 0
616	002010	102		.ASCII /B/
617	002011	060		.ASCII /O/
618	002012	000000		.WORD 0
619	002014	000000		.WORD 0
620	002016	022016		.WORD L\$HARD
621	002020	022166		.WORD L\$SOFT
622	002022	011656		.WORD L\$HW
623	002024	011670		.WORD L\$SW
624	002026	022424		.WORD L\$LAST
625	002030	000000		.WORD 0
626	002032	000000		.WORD 0
627	002034	000000		.WORD 0
628	002036	000000		.WORD 0
629	002040	011700		.WORD L\$DISPATCH
630	002042	000000		.WORD 0
631	002044	000000		.WORD 0
632	002046	000000		.WORD 0
633	002050	002		.BYTE C\$REVISION
634	002051	002		.BYTE C\$EDIT
635	002052	000000		.WORD 0
636	002054	000000		.WORD 0
637	002056	000000		.WORD 0
638	002060	000000		.WORD 0
639	002062	000000		.WORD 0
640	002064	002212		.WORD L\$DVTYP
641	002066	011704		.WORD L\$RPT
642	002070	002230		.WORD L\$DR
643	002072	002234		.WORD L\$DRST
644	002074	011706		.WORD L\$AU
645	002076	011710		.WORD L\$DU
646	002100	000000		.WORD 0
647	002102	000000		.WORD 0
648	002104	011712		.WORD L\$INIT
649	002106	012256		.WORD L\$CLEAN
650	002110		ENDMOD	

```
651 002110          BGNMOD GLOB
652                :.....
653                : GLOBAL EQUATES
654                :.....
655 002110          EQUALS
656
657                :
658                :
659                :
660                : VT62 DIAGNOSTIC EQUATES
661                :
662                : SOH=201          ;START OF HEADER (DDCMP)
663                : ACK=1          ;ACKNOWLEDGE
664                : NAK=2          ;NEGATIVE ACKNOWLEDGE
665                : REP=3          ;REPLY
666                : ENQ=5          ;ENQUIRY
667                : STRT=6         ;START
668                : STACK=7        ;START ACKNOWLEDGE
669                : SYNC=226       ;SYNCHRONIZATION CHARACTER
670                : DLE=220        ;DATA LINK ESCAPE.
671                : OSOP=24        ;LOOPBACK.
672                : PARITY=200     ;PARITY BIT.
673                : TSTS=2         ;NO. OF TESTS
674                : RCVCSR=0       ;OFFSETS: RECEIVER CSR
675                : RCVBUF=2       ;RECEIVER BUFFER REG.
676                : XMTCSR=4       ;TRANSMITTER CSR
677                : XMTBUF=6       ;TRANSMITTER BUFFER REG.
678                :
679                :
680                :
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                :
738                :
739                :
740                :
741                :
742                :
743                :
744                :
745                :
746                :
747                :
748                :
749                :
750                :
751                :
752                :
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                :
804                :
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                :
854                :
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                :
905                :
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                :
932                :
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                :
965                :
966                :
967                :
968                :
969                :
970                :
971                :
972                :
973                :
974                :
975                :
976                :
977                :
978                :
979                :
980                :
981                :
982                :
983                :
984                :
985                :
986                :
987                :
988                :
989                :
990                :
991                :
992                :
993                :
994                :
995                :
996                :
997                :
998                :
999                :
1000               :
```





```
739
740
741
742
743 002244 040503 020116 047516 MSG00: .ASCIZ /CAN NOT INITIALIZE COMM DEVICE/
744 002252 020124 047111 052111
745 002260 040511 044514 042532
746 002266 041440 046517 020115
747 002274 042504 044526 042503
748 002302 000
749 002303 122 050105 054514 MSG01: .ASCIZ /REPLY WAS NOT AN ENQ/
750 002310 053440 051501 047040
751 002316 052117 040440 020116
752 002324 047105 000121
753 002330 047111 047503 051122 MSG02: .ASCIZ /INCORRECT REPLY TYPE/
754 002336 041505 020124 042522
755 002344 046120 020131 054524
756 002352 042520 000
757 002355 103 051101 044522 MSG05: .ASCIZ /CARRIER NOT DETECTED AFTER 10MSEC/
758 002362 051105 047040 052117
759 002370 042040 052105 041505
760 002376 042524 020104 043101
761 002404 042524 020122 030061
762 002412 051515 041505 000
763 002417
764 002417 122 050105 054514 MSG42:
MSG03: .ASCIZ /REPLY HAS INCORRECT STATION ADDRESS/
765 002424 044040 051501 044440
766 002432 041516 051117 042522
767 002440 052103 051440 040524
768 002446 044524 047117 040440
769 002454 042104 042522 051523
770 002462 000
771 002463
772 002463 110 040505 042504 MSG41:
MSG04: .ASCIZ /HEADER HAS CRC ERROR/
773 002470 020122 040510 020123
774 002476 051103 020103 051105
775 002504 047522 000122
776 002510 051105 047522 020122 MSG10: .ASCIZ /ERROR DURING KEYBOARD TEST ** END OF REPORT **/
777 002516 052504 044522 043516
778 002524 045440 054505 047502
779 002532 051101 020104 042524
780 002540 052123 020040 025052
781 002546 042440 042116 047440
782 002554 020106 042522 047520
783 002562 052122 025040 000052
784 002570 040502 020104 042522 MSG11: .ASCIZ /BAD RESPONSE FROM KEYBOARD/
785 002576 050123 047117 042523
786 002604 043040 047522 020115
787 002612 042513 041131 040517
788 002620 042122 000
789 002623 124 047517 046440 MSG12: .ASCIZ /TOO MANY CHARACTERS SENT FROM KEYBOARD/
790 002630 047101 020131 044103
791 002636 051101 041501 042524
792 002644 051522 051440 047105
793 002652 020124 051106 046517
794 002660 045440 054505 047502
```

795	002666	051101	000104		
796	002672	051124	047101	046523	MSG20: .ASCIZ /TRANSMIT ERROR: NO TRANSMIT INTERRUPTS/
797	002700	052111	042440	051122	
798	002706	051117	020072	047516	
799	002714	052040	040522	051516	
800	002722	044515	020124	047111	
801	002730	042524	051122	050125	
802	002736	051524	000		
803	002741				MSG40:
804	002741	122	041505	050105	MSG21: .ASCIZ /RECEPTION ERROR/
805	002746	044524	047117	042440	
806	002754	051122	051117	000	
807	002761	116	020117	041501	MSG22: .ASCIZ /NO ACK RECEIVED AFTER TRANSMISSION/
808	002766	020113	042522	042503	
809	002774	053111	042105	040440	
810	003002	052106	051105	052040	
811	003010	040522	051516	044515	
812	003016	051523	047511	000116	
813	003024	051105	047522	020122	MSG23: .ASCIZ /ERROR WHILE INITIALIZING COMM DEVICE/
814	003032	044127	046111	020105	
815	003040	047111	052111	040511	
816	003046	044514	044532	043516	
817	003054	041440	046517	020115	
818	003062	042504	044526	042503	
819	003070	000			
820	003071	105	051122	051117	MSG24: .ASCIZ /ERROR DURING COMM DEVICE TRANSMIT/
821	003076	042040	051125	047111	
822	003104	020107	047503	046515	
823	003112	042040	053105	041511	
824	003120	020105	051124	047101	
825	003126	046523	052111	000	
826	003133	116	020117	052123	MSG25: .ASCIZ /NO STACK IN RESPONSE TO STRT/
827	003140	041501	020113	047111	
828	003146	051040	051505	047520	
829	003154	051516	020105	047524	
830	003162	051440	051124	000124	
831	003170	051105	047522	020122	MSG26: .ASCIZ /ERROR DURING COMM DEVICE RECEIVE/
832	003176	052504	044522	043516	
833	003204	041440	046517	020115	
834	003212	042504	044526	042503	
835	003220	051040	041505	044505	
836	003226	042526	000		
837	003231	103	047101	052047	MSG30: .ASCIZ /CAN'T GET CLEAR TO SEND/
838	003236	043440	052105	041440	
839	003244	042514	051101	052040	
840	003252	020117	042523	042116	
841	003260	000			
842	003261	124	046511	020105	MSG31: .ASCIZ /TIME OUT: NO CHARACTERS FROM TERMINAL/
843	003266	052517	035124	047040	
844	003274	020117	044103	051101	
845	003302	041501	042524	051522	
846	003310	043040	047522	020115	
847	003316	042524	046522	047111	
848	003324	046101	000		
849	003327	124	046511	020105	MSG32: .ASCIZ /TIME OUT: MESSAGE FROM TERMINAL INCOMPLETE/
850	003334	052517	035124	046440	

851 003342 051505 040523 042507  
852 003350 043040 047522 020115  
853 003356 042524 046522 047111  
854 003364 046101 044440 041516  
855 003372 046517 046120 052105  
856 003400 000105  
857 003402 040504 040524 041440  
858 003410 041522 042440 051122  
859 003416 051117 000  
860 003421 116 044517 042523  
861 003426 050040 052101 042524  
862 003434 047122 051440 041125  
863 003442 042524 052123 000  
864 003447 105 051122 051117  
865 003454 042040 051125 047111  
866 003462 020107 042523 043114  
867 003470 042524 052123 000  
868 003475 105 051122 051117  
869 003502 042040 051125 047111  
870 003510 020107 047514 050117  
871 003516 040502 045503 000  
872 003523 104 052101 020101  
873 003530 051105 047522 000122  
874 003536 046111 042514 040507  
875 003544 020114 042523 043114  
876 003552 052040 051505 020124  
877 003560 042522 050123 047117  
878 003566 042523 000  
879 003571 115 046505 051117  
880 003576 020131 051105 047522  
881 003604 020122 020040 054524  
882 003612 042520 040  
883 003615 000 000  
884 003617 115 026440 041040  
885 003624 047101 020113 043  
886 003631 000 000  
887 003634  
888 003634 051103 046517 042440  
889 003642 051122 051117 000  
890 003650  
891  
892 003650 005  
893 003651 006  
894 003652 300  
895 003653 000  
896 003654 000  
897 003655 000  
898 003656 000000  
899  
900  
901  
902  
903 003660  
904 003660  
905 003660 016746 176240  
906 003664 012746 003706

MSG43: .ASCIZ /DATA CRC ERROR/  
MSG45: .ASCIZ /NOISE PATTERN SUBTEST/  
MSG50: .ASCIZ /ERROR DURING SELFTEST/  
MSG51: .ASCIZ /ERROR DURING LOOPBACK/  
MSG52: .ASCIZ /DATA ERROR/  
MSG70: .ASCIZ /ILLEGAL SELF TEST RESPONSE/  
MSG71: .ASCII /MEMORY ERROR TYPE /  
MTY: .BYTE 0,0  
.ASCII /M - BANK #/  
BNO: .BYTE 0,0  
.EVEN  
MSG72: .ASCIZ /CROM ERROR/  
.EVEN  
STRTB: .BYTE ENQ  
.BYTE STRT  
.BYTE 300  
.BYTE 0  
.BYTE 0  
STRTSA: .BYTE 0  
STRTC: .WORD 0

\*\*\*\*\*  
: GLOBAL ERROR REPORT \*  
:\*\*\*\*\*  
BGNMSG AREA00  
PRINTX #FORM00,ILIMIT

MOV ILIMIT, -(SP)  
MOV #FORM00, -(SP)

907	003670	012746	000002					MOV	#2,-(SP)
908	003674	010600						MOV	SP,RO
909	003676	104015						EMI	(SPNTX
910	003700	062706	000006					ADD	#6,SP
911	003704			ENDMSG					
912	003704						L10000:		
913	003704	104023						EMI	(SMSG
914	003706	040445	043101	042524	FORM00: .ASCIZ /%AFTER %06% ATTEMPTS%/				
915	003714	020122	047445	022466					
916	003722	020101	052101	042524					
917	003730	050115	051524	047045					
918	003736	000							
919	003740			.EVEN					
920									
921	003740			BGNMSG AREA01					
922	003740			ENDMSG					
923	003740						L10001:		
924	003740	104023						EMI	(SMSG
925									
926	003742			BGNMSG AREA02					
927	003742			PRINTX #FORM02,A02A,A02B					
928	003742	016746	000030					MOV	A02B,-(SP)
929	003746	016746	000022					MOV	A02A,-(SP)
930	003752	012746	004000					MOV	#FORM02,-(SP)
931	003756	012746	000003					MOV	#3,-(SP)
932	003762	010600						MOV	SP,RO
933	003764	104015						EMI	(SPNTX
934	003766	062706	000010					ADD	#10,SP
935	003772			ENDMSG					
936	003772						L10002:		
937	003772	104023						EMI	(SMSG
938	003774	000000		A02A: 0					
939	003776	000000		A02B: 0					
940	004000	040445	042522	042503	FORM02: .ASCIZ /%RECEIVED %03% -- EXPECTED %03%/				
941	004006	053111	042105	022440					
942	004014	031517	040445	026440					
943	004022	020055	054105	042520					
944	004030	052103	042105	022440					
945	004036	031517	047045	000					
946		004044		.EVEN					
947									
948	004044			BGNMSG AREA03					
949	004044			AREA42:					
950	004044			PRINTX #FORM02,TEMP,SCRICH					
951	004044	016746	176042					MOV	SCRICH,-(SP)
952	004050	016746	176040					MOV	TEMP,-(SP)
953	004054	012746	004000					MOV	#FORM02,-(SP)
954	004060	012746	000003					MOV	#3,-(SP)
955	004064	010600						MOV	SP,RO
956	004066	104015						EMI	(SPNTX
957	004070	062706	000010					ADD	#10,SP
958	004074			ENDMSG					
959	004074						L10003:		
960	004074	104023						EMI	(SMSG
961									
962	004076			BGNMSG AREA04					

963	004076			AREA41:				
964	004076			AREA43:				
965	004076			PRINTX #FORM04,A04RC,A04EX				
966	004076	016746	000030			MOV	A04EX,-(SP)	
967	004102	016746	000022			MOV	A04RC,-(SP)	
968	004106	012746	004134			MOV	#FORM04,-(SP)	
969	004112	012746	000003			MOV	#3,-(SP)	
970	004116	010600				MOV	SP,R0	
971	004120	104015				EMT	C\$PNTX	
972	004122	062706	000010			ADD	#10,SP	
973	004126			ENDMSG				
974	004126					L10004:		
975	004126	104023				EMT	C\$MSG	
976	004130	000000		A04RC: 0				
977	004132	000000		A04EX: 0				
978	004134	040445	042522	042503	FORM04: .ASCIZ /%RECEIVED %06%A -- EXPECTED %06%N/			
979	004142	053111	042105	022440				
980	004150	033117	040445	026440				
981	004156	020055	054105	042520				
982	004164	052103	042105	022440				
983	004172	033117	047045	000				
984		004200		.EVEN				
985								
986	004200			BGNMSG AREA11				
987	004200			PRINTX #FORM04,KRC,KEX				
988	004200	016746	000030			MOV	KEX,-(SP)	
989	004204	016746	000022			MOV	KRC,-(SP)	
990	004210	012746	004134			MOV	#FORM04,-(SP)	
991	004214	012746	000003			MOV	#3,-(SP)	
992	004220	010600				MOV	SP,R0	
993	004222	104015				EMT	C\$PNTX	
994	004224	062706	000010			ADD	#10,SP	
995	004230			ENDMSG				
996	004230					L10005:		
997	004230	104023				EMT	C\$MSG	
998	004232	000000		KRC: 0				
999	004234	000000		KEX: 0				
1000	004236			BGNMSG AREA21				
1001	004236			ENDMSG				
1002	004236					L10006:		
1003	004236	104023				EMT	C\$MSG	
1004	004240			BGNMSG AREA22				
1005	004240			ENDMSG				
1006	004240					L10007:		
1007	004240	104023				EMT	C\$MSG	
1008	004242			BGNMSG AREA23				
1009	004242							
1010	004242			AREA24:				
1011	004242			AREA40:				
1012	004242	016767	175670	000064	LET TEMP2 := LOGDEV	MOV	LOGDEV,TEMP2	
1013	004250				IF TEMP2 LT #0 THEN			
1014	004250	005767	000060			TST	TEMP2	
1015	004254	002002				BGE	500008	
1016	004256				LET TEMP2 := #0			
1017	004256	005067	000052			CLR	TEMP2	
1018	004262			ENDIF				





1075	004516	012701	002234				MOV	#DUPREG,R1
1076	004522			PRINTX #FORM30,RCVCSR(R1)				
1077	004522	016146	000000				MOV	RCVCSR(R1),-(SP)
1078	004526	012746	004646				MOV	#FORM30,-(SP)
1079	004532	012746	000002				MOV	#2,-(SP)
1080	004536	010600					MOV	SP,R0
1081	004540	104015					EMT	C\$PNTX
1082	004542	062706	000006				ADD	#6,SP
1083	004546			PRINTX #FRM30A,RCVBUF(R1)				
1084	004546	016146	000002				MOV	RCVBUF(R1),-(SP)
1085	004552	012746	004670				MOV	#FRM30A,-(SP)
1086	004556	012746	000002				MOV	#2,-(SP)
1087	004562	010600					MOV	SP,R0
1088	004564	104015					EMT	C\$PNTX
1089	004566	062706	000006				ADD	#6,SP
1090	004572			PRINTX #FRM30B,XMTCSR(R1)				
1091	004572	016146	000004				MOV	XMTCSR(R1),-(SP)
1092	004576	012746	004713				MOV	#FRM30B,-(SP)
1093	004602	012746	000002				MOV	#2,-(SP)
1094	004606	010600					MOV	SP,R0
1095	004610	104015					EMT	C\$PNTX
1096	004612	062706	000006				ADD	#6,SP
1097	004616			PRINTX #FRM30C,XMTBUF(R1)				
1098	004616	016146	000006				MOV	XMTBUF(R1),-(SP)
1099	004622	012746	004735				MOV	#FRM30C,-(SP)
1100	004626	012746	000002				MOV	#2,-(SP)
1101	004632	010600					MOV	SP,R0
1102	004634	104015					EMT	C\$PNTX
1103	004636	062706	000006				ADD	#6,SP
1104	004642			POP R1				
1105	004642	012601					MOV	(SP)+,R1
1106	004644			ENDMSG				
1107	004644					L10012:		
1108	004644	104023					EMT	C\$MSG
1109	004646	040445	052504	020120	FORM30: .ASCIZ /%ADUP RCSR: %06%N/			
1110	004654	041522	051123	020072				
1111	004662	047445	022466	030116				
1112	004670	040445	052504	020120	FRM30A: .ASCIZ /%ADUP RBUF: %06%N/			
1113	004676	041122	043125	035106				
1114	004704	022440	033117	047045				
1115	004712	000						
1116	004713	045	042101	050125	FRM30B: .ASCIZ /%ADUP TCSR: %06%N/			
1117	004720	052040	051503	035122				
1118	004726	022440	033117	047045				
1119	004734	000						
1120	004735	045	042101	050125	FRM30C: .ASCIZ /%ADUP TBUF: %06%N/			
1121	004742	052040	052502	043106				
1122	004750	020072	047445	022466				
1123	004756	000116						
1124					.EVEN			
1125								
1126	004760				BGNMSG AREA60			
1127	004760				PRINTX #FORM60			
1128	004760	012746	005002				MOV	#FORM60,-(SP)
1129	004764	012746	000001				MOV	#1,-(SP)
1130	004770	010600					MOV	SP,R0

1131	004772	104015					EMT	C\$PNTX
1132	004774	062706	000004				ADD	#4,SP
1133	005000				ENDMSG			
1134	005000					L10013:		
1135	005000	104023					EMT	C\$MSG
1136	005002	040445	040503	023516	FORM60: .ASCIZ /%ACAN'T GET INTO LOOPBACK%N/			
1137	005010	020124	042507	020124				
1138	005016	047111	047524	046040				
1139	005024	047517	041120	041501				
1140	005032	022513	000116					
1141					.EVEN			
1142	005036				BGNMSG AREA61			
1143	005036				PRINTX #FORM61			
1144	005036	012746	005130				MOV	#FORM61,-(SP)
1145	005042	012746	000001				MOV	#1,-(SP)
1146	005046	010690					MOV	SP,RO
1147	005050	104015					EMT	C\$PNTX
1148	005052	062706	000004				ADD	#4,SP
1149	005056				LET TEMP2 := LOGDEV			
1150	005056	016767	175054	177250			MOV	LOGDEV,TEMP2
1151	005064				IF TEMP2 LT #0 THEN			
1152	005064	005767	177244				TST	TEMP2
1153	005070	002002					BGE	50001\$
1154	005072				LET TEMP2 := #0			
1155	005072	005067	177236				CLR	TEMP2
1156	005076				ENDIF			
1157	005076					50001\$:		
1158	005076				LET TEMP2 := TEMP2 + #1			
1159	005076	005267	177232				INC	TEMP2
1160	005102				PRINTX #FORM23,TEMP2			
1161	005102	016746	177226				MOV	TEMP2,-(SP)
1162	005106	012746	004336				MOV	#FORM23,-(SP)
1163	005112	012746	000002				MOV	#2,-(SP)
1164	005116	010600					MOV	SP,RO
1165	005120	104015					EMT	C\$PNTX
1166	005122	062706	000006				ADD	#6,SP
1167	005126				ENDMSG			
1168	005126					L10014:		
1169	005126	104023					EMT	C\$MSG
1170	005130	040445	040503	023516	FORM61: .ASCIZ /%ACAN'T GET OUT OF LOOPBACK%N/			
1171	005136	020124	042507	020124				
1172	005144	052517	020124	043117				
1173	005152	046040	047517	041120				
1174	005160	041501	022513	000116				
1175					.EVEN			
1176	005166				BGNMSG AREA52			
1177	005166				PRINTX #FORM52,A52A,A52B			
1178	005166	016746	000074				MOV	A52B,-(SP)
1179	005172	016746	000066				MOV	A52A,-(SP)
1180	005176	012746	005220				MOV	#FORM52,-(SP)
1181	005202	012746	000003				MOV	#3,-(SP)
1182	005206	010600					MOV	SP,RO
1183	005210	104015					EMT	C\$PNTX
1184	005212	062706	000010				ADD	#10,SP
1185	005216				ENDMSG			
1186	005216					L10015:		

```
1187 005216 104023
1188 005220 040445 054105 042520 FORM52: .ASCIZ /%AEXPECTED: %03%A -RECEIVED: %03%N/
1189 005226 052103 042105 020072
1190 005234 047445 022463 020101
1191 005242 026440 042522 042503
1192 005250 053111 042105 020072
1193 005256 047445 022463 000116
1194
1195 005264 000000          A52A:  0      .EVEN
1196 005266 000000          A52B:  0
1197
1198 005270                BGNMSG AREA70
1199 005270                ENDMSG
1200 005270
1201 005270 104023          L10016:
1202                                EMT      CSMSG
1203 005272                BGNMSG AREA71
1204 005272                ENDMSG
1205 005272
1206 005272 104023          L10017:
1207                                EMT      CSMSG
1208 005274
1209 005274 001452          TEMBUF:
1210 010420                .BLKW 810.
1211 010420 000416          TMPBUF:
1212 011454 000100          .BLKW 270.
1213 011654                .BLKW 100
1214                                SSTACK:          ;STACK FOR SUBROUTINE LINKAGE.
1215 011654                ENDMOD
```

```
1216 011654
1217
1218
1219
1220 011654
1221 011654 000004
1222 011656 160010
1223 011660 000320
1224 011662 000640
1225 011664 007030
1226 011666
1227 011666
1228 011666
1229 011666 000003
1230 011670 000010
1231 011672 000001
1232 011674 020000
1233 011676
1234 011676
1235
1236
1237
1238
1239 011676
1240 011676 000002
1241 011700 017464
1242 011702 020514
1243 011704
```

```

      BGNMOD TABLE
      .....
      DEFAULT HARDWARE P-TABLE AND SOFTWARE P-TABLE
      .....
      BGNHW
      160010
      320
      640
      7030
      ENDMW
      L10020:
      BGNSW
      000010
      PATTERN:000001
      STWAIT: 20000
      ENDSW
      L10021:
      .....
      DISPATCH TABLE
      .....
      DISPATCH TSTS
      TSTS
      T1
      T2
      ENDMOD
```

```

      .WORD L10020-LSHW/2
      .WORD L10021-LSSW/2
      .WORD TSTS
      .WORD T1
      .WORD T2
```



```
1244 011704          BGNMOD CODE
1245
1246                :*****
1247                :STUB AREAS FOR SUPERVISOR COMPATIBILITY *
1248                :*****
1249 011704          BGNRPT
1250 011704          ENDRPT
1251 011704
1252 011704 104025          L10022:  EMT      CSRPT
1253
1254 011706          BGNAU
1255 011706          ENDAU
1256 011706
1257 011706 104054          L10023:  EMT      CSAU
1258
1259 011710          BGNDU
1260 011710          ENDDU
1261 011710
1262 011710 104055          L10024:  EMT      CSDU
1263
1264                :*****
1265                :      INITIALIZE CODING
1266                :*****
1267 011712          BGNINIT
1268
1269                :SAVE R1
1270
1271                :      PUSH R1
1272 011712 010146          MOV      R1,-(SP)
1273
1274                :CHECK START EVENT FLAG TO SEE IF WE GOT HERE BECAUSE OF A START
1275                :
1276 011714          READEF #EF.START
1277 011714 012700 000040          MOV      #EF.START,R0
1278 011720 104050          EMT      CSREFG
1279
1280                :IF A START, THEN...
1281                :
1282 011722          IF.ERROR THEN
1283 011722 103005          BCC      50002$
1284
1285                :INITIALIZE THE SUBROUTINE LINKAGE STACK
1286                :
1287 011724          LET R5 := #SSTACK
1288 011724 012705 011654          MOV      #SSTACK,R5
1289 011730          LET R5STCK := R5STCK + #1
1290 011730 005267 170154          INC      R5STCK
1291 011734          ELSE
1292 011734 000414          BR      50003$
1293 011736          50002$:
1294
1295                :IF NOT START THEN....
1296                :
1297                :CHECK FOR RESTART AND IF IT IS:
1298                :UPDATE LOGICAL DEVICE AND STAT. TABLE ENTRY POINTER,
1299                :BUT DON'T EXCEED THE NUMBER OF UNITS SPECIFIED
```

```
1300
1301 011736      :
1302 011736 012700 000637      READEF #EF.RESTART
1303 011742 104050      MOV #EF.RESTART,RO
1304 011744      EMT CSREFG
1305 011744 103410      IF.NO.ERROR THEN
1306 011746      LET LOGDEV := LOGDEV + #1
1307 011746 005267 170164      BCS 50004$
1308 011752      IF LOGDEV EQ LSUNIT THEN
1309 011752 026767 170160 170032      INC LOGDEV
1310 011760 001002      CMP LOGDEV,LSUNIT
1311 011762      LET LOGDEV := #0
1312 011762 005067 170150      BNE 50005$
1313 011766      CLR LOGDEV
1314 011766      ENDIF
1315 011766      ENDIF
1316 011766      ENDIF
1317 011766      ENDIF
1318 011766      ENDIF
1319
1320 :CHECK FOR NEW PASS... IF IT IS, FORCE LOGICAL UNIT TO ZERO.
1321 :
1322 011766      READEF #EF.NEW
1323 011766 012700 000035      MOV #EF.NEW,RO
1324 011772 104050      EMT CSREFG
1325 011774      IF.ERROR THEN
1326 011774 103002      BCC 50006$
1327 011776      LET LOGDEV := #0
1328 011776 005067 170134      CLR LOGDEV
1329 012002      ENDIF
1330 012002      ENDIF
1331
1332 :END OF START CODE
1333 :FORCE THE COMM DEVICE TO BE INITIALIZED
1334 :
1335 012002      LET INIT := #0
1336 012002 005067 003556      CLR INIT
1337 012006      LET ABORT := #0
1338 012006 005067 170104      CLR ABORT
1339
1340 :SET UP THE INPUT BUFFERS QUEUE
1341 :
1342 012012      LET RQUE := #RQUET
1343 012012 012767 015610 003566      MOV #RQUET,RQUE
1344 012020      LET RQUEUE := RQUE
1345 012020 016767 003562 003556      MOV RQUE,RQUEUE
1346 012026      LET RBQUE := #RBQUET
1347 012026 012767 015626 003570      MOV #RBQUET,RBQUE
1348 012034      LET R1 := RBQUE
1349 012034 016701 003564      MOV RBQUE,R1
1350 012040      LET SCRATCH := #TEMBUF
1351 012040 012767 005274 170044      MOV #TEMBUF,SCRATCH
1352 012046      REPEAT
1353 012046      LET (R1)+ := SCRATCH
1354 012046      REPEAT
1355 012046 016721 170040      MOV SCRATCH,(R1)+
```

```

1356 012052          LET SCRTCH := SCRTCH + #270.
1357 012052 062767 000416 170032          ADD      #270.,SCRTCH
1358 012060          UNTIL R1 GT #RBQUEB
1359 012060 020127 015632          CMP      R1,#RBQUEB
1360 012064 003770          BLE      50007$
1361
1362          ; GET THE PTABLE ENTRY FOR THIS LOGICAL DEVICE
1363          ;
1364 012066          REPEAT
1365 012066          50010$:
1366 012066          LET R1 := R1 + #1
1367 012066 005201          INC      R1
1368 012070          GPHARD LOGDEV,SCRTCH
1369 012070 016700 170042          MOV      LOGDEV,R0
1370 012074 104042          EMT
1371 012076 010067 170010          MOV      R0,SCRTCH
1372
1373          ; IF UNIT HAS BEEN DROPPED, UPDATE POINTERS ETC. AND TRY AGAIN
1374          ;
1375 012102          IF .NO.ERROR THEN
1376 012102 103411          BCS      50011$
1377 012104          LET LOGDEV := LOGDEV + #1
1378 012104 005267 170026          INC      LOGDEV
1379 012110          IF LOGDEV EQ L$UNIT THEN
1380 012110 026767 170022 167674          CMP      LOGDEV,L$UNIT
1381 012116 001002          BNE      50012$
1382 012120          LET LOGDEV := #0
1383 012120 005067 170012          CLR      LOGDEV
1384 012124          ENDIF
1385 012124          50012$:
1386 012124          LET R1 := #0
1387 012124 005001          CLR      R1
1388 012126          ENDIF
1389 012126          50011$:
1390 012126          UNTIL R1 NE #0
1391 012126 005701          TST      R1
1392 012130 001756          BEQ      50010$
1393
1394          ; NOW GET THE PTABLE DATA FROM THE SUPERVISOR AND STORE WITHIN THE
1395          ; DIAGNOSTIC PROGRAM
1396          ;
1397 012132          LET R1 := #PTABLE
1398 012132 012701 002200          MOV      #PTABLE,R1
1399 012136          REPEAT
1400 012136          50013$.
1401 012136          LET (R1) := @SCRTCH
1402 012136 017711 167750          MOV      @SCRTCH,(R1)
1403 012142          LET R1 := R1 + #2
1404 012142 062701 000002          ADD      #2,R1
1405 012146          LET SCRTCH := SCRTCH + #2
1406 012146 062767 000002 167736          ADD      #2,SCRTCH
1407 012154          UNTIL R1 EQ #PTEND
1408 012154 020127 002210          CMP      R1,#PTEND
1409 012160 001366          BNE      50013$
1410
1411          ; RESTORE R1 AND EXIT INIT CODE

```

```
1412
1413 012162 ; POP R1
1414 012162 012601 MOV (SP)+,R1
1415 012164 ENDINIT
1416 012164 L10025:
1417 012164 104011 EMT CSINIT
1418 012166 047045 040445 052126 VMSG: .ASCIZ /%N%AVT62 ACCEPTANCE TEST -- VERSION 10 TERMINALS%N/
1419 012174 031066 040440 041503
1420 012202 050105 040524 041516
1421 012210 020105 042524 052123
1422 012216 020040 026440 020055
1423 012224 020040 042526 051522
1424 012232 047511 020116 030061
1425 012240 052040 051105 044515
1426 012246 040516 051514 047045
1427 012254 000
1428 012256 .EVEN
1429
1430 :.....
1431 : CLEAN-UP CODING
1432 :.....
1433 012256 BGNCLN
1434
1435 : MAKE SURE DEVICE IS SHUT DOWN
1436 :
1437 012256 PUSH R1
1438 012256 010146 MOV R1,-(SP)
1439 012260 LET R1 := PTABLE MOV PTABLE,R1
1440 012260 016701 167714 LET XMTCSR(R1) := XMTCSR(R1) SET.BY #MASCLR BIS #MASCLR,XMTCSR(R)
1441 012264 LET XMTCSR(R1) := XMTCSR(R1) SET.BY #MASCLR BIS #MASCLR,XMTCSR(R)
1442 012264 052761 000400 000004 POP R1 MOV (SP)+,R1
1443 012272
1444 012272 012601
1445
1446 : UPDATE PASS COUNT
1447 :
1448 012274 LET PASCNT := PASCNT + #1
1449 012274 005267 167632 INC PASCNT
1450
1451 : EXIT CLEAN-UP CODE
1452 :
1453 012300 ENDCLN
1454 012300 L10026:
1455 012300 104012 EMT CS CLEAN
1456 012302 ENDMOD
```

```

1457 012302          BGNMOD GSUB
1458
1459          .....
1460          GLOBAL SUBROUTINES
1461          .....
1462          .....
1463          ROUTINE TO PROCESS TERMINAL MESSAGES FOR ALL TESTS
1464          RETURNS POINTER TO MESSAGE ('OUTMSG') AND ITS SIZE
1465          IN BYTES ('OUTSIZ')
1466
1467          SIZE MAY BE ANY NUMBER OF BYTES
1468
1469          SETS A FLAG ('EXPINP') IF RESPONSE OTHER THAN 'ACK' IS
1470          TO BE EXPECTED
1471
1472          SETS ANOTHER FLAG ('EOMSG') IF THIS IS THE LAST MESSAGE
1473          .....
1474 012302          ROUTINE MSG          ;[3.2.14]
1475 012302
1476
1477 012302          DECLARE LOCALS
1478 012302          LOCAL <A>
1479 012302          ALLOCATE
1480 012302 010446          MOV          R4,-(SP)
1481 012304 010604          MOV          SP,R4
1482 012306 162706 000002  SUB          #2,SP
1483
1484          ;GET MESSAGE POINTER
1485
1486 012312          LET A(R4) := @MSGAP
1487 012312 017764 167644 177776          MOV          @MSGAP,A(R4)
1488
1489          ;IF MESSAGE POINTER IS ODD,
1490          ;SET END_OF_MESSAGE INDICATOR
1491
1492 012320          LET SCRTCH := A(R4) AND #1
1493 012320 016467 177776 167564          MOV          A(R4),SCRTCH
1494 012326 016746 167560          MOV          SCRTCH,-(SP)
1495 012332 042716 000001          BIC          #1,(SP)
1496 012336 042667 167550          BIC          (SP)+,SCRTCH
1497 012342          IF SCRTCH NE #0 THEN
1498 012342 005767 167544          TST          SCRTCH
1499 012346 001405          BEQ          500028
1500 012350          LET EOMSG := EOMSG + #1
1501 012350 005267 167560          INC          EOMSG
1502 012354          LET A(R4) := A(R4) CLR.BY #1
1503 012354 042764 000001 177776          BIC          #1,A(R4)
1504 012362          ENDIF
1505 012362          500028:
1506
1507          ;IF MESSAGE_POINTER IS NEGATIVE,
1508          ;SET EXPECT_INPUT INDICATOR
1509
1510 012362          IF A(R4) LT #0 THEN
1511 012362 005764 177776          TST          A(R4)
1512 012366 002006          BGE          500038

```

```

1513 012370          LET EXPINP := EXPINP + #1
1514 012370 005267 167544          INC      EXPINP
1515 012374          LET A(R4) := A(R4) CLR.BY #100000
1516 012374 042764 100000 177776    BIC      #100000,A(R4)
1517
1518          ; OTHERWISE, CLEAR EXPECT_INPUT
1519          ;
1520 012402          ELSE
1521 012402 000402          BR      500048
1522 012404          500038:
1523 012404          LET EXPINP := #0
1524 012404 005067 167530          CLR      EXPINP
1525 012410          ENDIF
1526 012410          500048:
1527
1528          ; SET MESSAGE_ADDRESS AND
1529          ; MESSAGE_SIZE
1530          ;
1531 012410          LET OUTMSG := A(R4)
1532 012410 016467 177776 167524    MOV      A(R4),OUTMSG
1533 012416          LET MSGAP := MSGAP + #2
1534 012416 062767 000002 167536    ADD      #2,MSGAP
1535 012424          LET OUTMSG+2 := @MSGAP
1536 012424 017767 167532 167512    MOV      @MSGAP,OUTMSG+2
1537
1538          ; UPDATE MESSAGE_AREA_POINTER
1539          ;
1540 012432          LET MSGAP := MSGAP + @MSGAP
1541 012432 067767 167524 167522    ADD      @MSGAP,MSGAP
1542 012440          LET MSGAP := MSGAP + #2
1543 012440 062767 000002 167514    ADD      #2,MSGAP
1544
1545          ; IF MESSAGE_AREA_POINTER IS ODD
1546          ; ADD 1 TO MAKE IT EVEN
1547          ;
1548 012446          LET SCRATCH := MSGAP AND #1
1549 012446 016767 167510 167436    MOV      MSGAP,SCRATCH
1550 012454          MOV      SCRATCH,-(SP)
1551 012460          BIC      #1,(SP)
1552 012464          BIC      (SP)+,SCRATCH
1553 012470          IF SCRATCH NE #0 THEN
1554 012470 005767 167416          TST      SCRATCH
1555 012474 001402          BEQ      500058
1556 012476          LET MSGAP := MSGAP + #1
1557 012476 005267 167460          INC      MSGAP
1558 012502          ENDIF
1559 012502          500058:
1560 012502          ENDRTN
1561 012502          500008:
1562 012502          500018:
1563 012502 010406          MOV      R4,SP
1564 012504 012604          MOV      (SP)+,R4
1565 012506 000207          RTS      PC
1566
1567
1568

```

```

1569
1570
1571
1572
1573
1574
1575
1576 012510
1577 012510
1578 012510
1579 012510 005767 003050
1580 012514 001011
1581 012516
1582 012516 004767 000222
1583 012522
1584 012522 103002
1585 012524
1586 012524 000261
1587 012526 000504
1588 012530
1589 012530
1590 012530
1591 012530 005067 167440
1592 012534
1593 012534 005067 167436
1594 012540
1595 012540
1596
1597
1598
1599 012540
1600 012540 016767 167376 167416
1601 012546
1602 012546 012767 002142 167336
1603 012554 062767 000002 167330
1604 012562
1605 012562 017767 167324 167376
1606
1607 012570
1608 012570 012767 010420 167372
1609 012576
1610 012576 012767 000412 167366
1611
1612
1613
1614 012604
1615 012604 162705 000004
1616 012610 004767 000336
1617 012614 012567 167332
1618 012620 012567 167330
1619
1620
1621
1622
1623
1624 012624

```

```

:*****
: ROUTINE TO HANDLE OUTPUT OF MESSAGES TO THE TERMINAL
: UNDER TEST. MESSAGES MAY BE OF ANY LENGTH SINCE
: ROUTINES USED BY THIS ROUTINE SPLIT THE MESSAGES INTO
: BUFFERS OF APPROPRIATE LENGTH FOR DDCMP.
:*****
ROUTINE TEROUT ;[3.2.20]
IF INIT EQ #0 THEN
    CALL XSTRT
    IF .ERROR THEN
        RETURN ERROR
    ENDIF
    LET MSGNO := #0
    LET RMSGNO := #0
ENDIF
50003$:
CLR MSGNO
CLR RMSGNO
50002$:
: INITIALIZE THE BUFFER ROUTINES [3.2.20.4]
:
LET INITMA := OUTMSG
MOV OUTMSG,INITMA
LET SCRCH := #OUTMSG + #2
MOV #OUTMSG,SCRCH
ADD #2,SCRCH
LET INITMS := @SCRCH
MOV @SCRCH,INITMS
:
BUFFER #1,BAPNT
LET BAPNT := #TMPBUF
MOV #TMPBUF,BAPNT
LET BASIZE := #266.
MOV #266.,BASIZE
:
: GET A BUFFER OF THE MESSAGE
:
CALL GETOB OUT <OUTBUF,OUTSIZ>
SUB #2*2,R5
JSR PC,GETOB
MOV (R5)+,OUTBUF
MOV (R5)+,OUTSIZ
:
: PERFORM THE FOLLOWING LOOP WHILE
: A AND ABORT ARE CLEAR AND OUTSIZ
: IS GREATER THAN 0
:
WHILE OUTSIZ GT #0 AND ABORT EQ #0 DO

```



1625	012624							
1626	012624	005767	167324					
1627	012630	003442						
1628	012632	005767	167260					
1629	012636	001037						
1630								
1631								
1632								
1633	012640							
1634	012640	162705	000004					
1635	012644	010546						
1636	012646	016745	167302					
1637	012652	016745	167274					
1638	012656	004767	000416					
1639	012662	012605						
1640	012664	012567	002700					
1641	012670	012567	002676					
1642	012674							
1643	012674	004767	001674					
1644								
1645								
1646								
1647	012700							
1648	012700	103005						
1649	012702							
1650	012702	104423						
1651	012704	000024						
1652	012706	002672						
1653	012710							
1654	012710	000261						
1655	012712	000412						
1656	012714							
1657	012714							
1658								
1659								
1660								
1661	012714							
1662	012714	162705	000004					
1663	012720	004767	000226					
1664	012724	012567	167222					
1665	012730	012567	167220					
1666								
1667								
1668								
1669	012734							
1670	012734	000733						
1671	012736							
1672								
1673								
1674								
1675	012736							
1676	012736							
1677	012736	000241						
1678	012740							
1679	012740	000207						
1680	012742	000000						

500048:

TST OUTSIZ  
BLE 500058  
TST ABORT  
BNE 500058

ADD PROTOCOL ENVELOPE AND TRANSMIT

CALL PROBUF IN <OUTBUF,OUTSIZ> OUT <XBUFFER,XSIZE>

SUB #2\*2,R5  
MOV R5,-(SP)  
MOV OUTSIZ,-(R5)  
MOV OUTBUF,-(R5)  
JSR PC,PROBUF  
MOV (SP)+,R5  
MOV (R5)+,XBUFFER  
MOV (R5)+,XSIZE

CALL XMIT

JSR PC,XMIT

IF TRANSMIT ERROR, REPORT IT AND SET ERRXMT

IF.ERROR THEN

ERRHRD 20,MSG20 ;[3.2.20.10]

BCC 500068  
TRAP T\$ERCODE  
.WORD 20  
.WORD MSG20

RETURN ERROR

SEC  
BR 500018

ENDIF

500068:

GET THE NEXT BUFFER

CALL GETOB OUT <OUTBUF,OUTSIZ>

SUB #2\*2,R5  
JSR PC,GETOB  
MOV (R5)+,OUTBUF  
MOV (R5)+,OUTSIZ

END OF LOOP

ENDDO

BR 500048

500058:

IF ERROR OCCURRED, DO AN ERROR RETURN

ENDRTN

500008:

500018:

CLC  
RTS PI

TYPE: 0 ;TYPE OF MESSAGE EXPECTED

```

1681
1682
1683
1684
1685
1686
1687
1688
1689
1690 012744
1691 012744
1692 012744
1693 012744
1694 012744
1695 012744 010446
1696 012746 010604
1697 012750 162706 000002
1698
1699
1700
1701
1702
1703 G12754
1704 012754 012767 000010 002610
1705 012762
1706 012762 012767 003650 002600
1707
1708
1709
1710 012770
1711 012770 012764 002200 177776
1712 012776 062764 000005 177776
1713 013004
1714 013004 117467 177776 170643
1715
1716
1717
1718
1719 013012
1720 013012 162705 000002
1721 013016 010546
1722 013020 012745 000006
1723 013024 016745 002540
1724 013030 004767 004054
1725 013034 012605
1726 013036 012567 170614
1727
1728
1729
1730 013042
1731 013042 004767 001526
1732
1733
1734
1735 013046
1736 013046 103005
    
```

```

.....
ROUTINE TO SEND A "STRT" MESSAGE TO THE TERMINAL
.....
A "STACK" MUST BE RETURNED BY THE TERMINAL
.....
ROUTINE XSTRT                                ;[3.2.20.1]
XSTRT:
    DECLARE LOCALS
        LOCAL <A>
    ALLOCATE

                                MOV     R4, -(SP)
                                MOV     SP, R4
                                SUB     #2, SP

; THE DDCMP STRT MESSAGE IS 10 BYTES LONG (INCLUDING CRC) AND IS IN
; A BUFFER STARTING AT "STRTB" -- PUT THIS INFORMATION
; IN THE TRANSMIT ROUTINE'S POINTERS
    LET XSIZE := #10
                                MOV     #10, XSIZE
    LET XBUFFER := #STRTB
                                MOV     #STRTB, XBUFFER

; GET THE TERMINAL ADDRESS AND PUT INTO THE STRT MESSAGE
    LET A(R4) := #PTABLE + #5
                                MOV     #PTABLE, A(R4)
                                ADD     #5, A(R4)
    LET STRTSA :B= @A(R4)
                                MOVB   @A(R4), STRTSA

; COMPUTE THE CRC FOR THE STRT MESSAGE AND PUT IN THE LAST TWO BYTES
; OF THE BUFFER ("STRTC")
    CALL CRC IN <XBUFFER, #6> OUT <STRTC>
                                ;[3.2.20.1.2 + 3]
                                SUB     #1*2, R5
                                MOV     R5, -(SP)
                                MOV     #6, -(R5)
                                MOV     XBUFFER, -(R5)
                                JSR     PC, CRC
                                MOV     (SP)+, R5
                                MOV     (R5)+, STRTC

; TRANSMIT THE MESSAGE
    CALL XMIT
                                JSR     PC, XMIT

; IF AN ERROR OCCURS, REPORT IT AND EXIT WITH THE ERROR FLAG SET.
    IF .ERROR THEN
                                BCC     500028
    
```

```

1737 013050          ERRHRD 20,MSG20          ;[3.2.20.10]
1738 013050 104423
1739 013052 000024          TRAP          T$ERCODE
1740 013054 002672          .WORD          20
1741 013056          RETURN ERROR          .WORD          MSG20
1742 013056 000261          SEC
1743 013060 000430          BR          50001$
1744 013062          ENDIF
1745 013062          50002$:
1746
1747          :OTHERWISE, GET THE TERMINAL'S RESPONSE
1748          :
1749 013062          CALL RCVE
1750 013062 004767 003540          JSR          PC,RCVE
1751
1752          :IF RECEPTION ERROR, REPORT IT AND EXIT WITH THE ERROR FLAG SET
1753          :
1754 013066          IF.ERROR THEN
1755 013066 103005
1756 013070          ERRHRD 21,MSG21          ;[3.2.20.10]
1757 013070 104423          TRAP          T$ERCODE
1758 013072 000025          .WORD          21
1759 013074 002741          .WORD          MSG21
1760 013076          RETURN ERROR
1761 013076 000261          SEC
1762 013100 000420          BR          50001$
1763 013102          ENDIF
1764 013102          50003$:
1765
1766          :OTHERWISE, SET TYPE TO "STACK" AND CHECK TO SEE THAT IT WAS THE REPLY
1767          :
1768 013102          LET TYPE := #STACK
1769 013102 012767 000007 177632          MOV          #STACK,TYPE
1770 013110          CALL CHKREP IN <TYPE>
1771 013110 010546          MOV          R5,-(SP)
1772 013112 016745 177624          MOV          TYPE,-(R5)
1773 013116 004767 001062          JSR          PC,CHKREP
1774 013122 012605          MOV          (SP)+,R5
1775
1776          :IF REPLY WASN'T A "STACK", REPORT THE ERROR AND EXIT WITH THE ERROR FLAG SET
1777          :
1778 013124          !IF.ERROR THEN
1779 013124 103005
1780 013126          ERRHRD 25,MSG25          ;[3.2.20.10]
1781 013126 104423          TRAP          T$ERCODE
1782 013130 000031          .WORD          25
1783 013132 003133          .WORD          MSG25
1784 013134          RETURN ERROR
1785 013134 000261          SEC
1786 013136 000401          BR          50001$
1787 013140          ENDIF
1788 013140          50004$.
1789
1790          :END OF XSTRT ROUTINE
1791          :
1792 013140          ENDRTN
  
```

```

1793 013140
1794 013140 000241
1795 013142
1796 013142 010406
1797 013144 012604
1798 013146 000207
1799 013150 0000C0          DDSTRY: 0
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811 013152
1812 013152
1813 013152
1814 013152
1815 013152
1816 013152 010446
1817 013154 010604
1818 013156 162706 000002
1819
1820
1821
1822 013162
1823 013162 016765 167002 000000
1824 013170 062765 000010 000000
1825
1826
1827
1828 013176
1829 013176 005065 000002
1830
1831
1832
1833 013202
1834 013202 016767 166764 166702
1835 013210 162767 000012 166674
1836
1837
1838
1839 013216
1840 013216 016564 000000 177776
1841
1842
1843
1844 013224
1845 013224
1846 013224 026567 000002 166660
1847 013232 002017
1848 013234 005767 166726

```

```

50000$: CLC
50001$: MOV R4,SP
          MOV (SP)+,R4
          RTS PC
          ;[3.2.20.5]
          GETOB:
          DECLARE LOCALS
          LOCAL <C>
          ALLOCATE
          MOV R4,-(SP)
          MOV SP,R4
          SUB #2,SP
          MOV BAPNT,A(R5)
          ADD #10,A(R5)
          CLR B(R5)
          MOV BASIZE,SCRTCH
          SUB #12,SCRTCH
          MOV A(R5),C(R4)
          50002$: CMP B(R5),SCRTCH
                BGE 50003$
                IST INITMS

```

```

*****
: THIS ROUTINE SPLITS A MESSAGE OF ANY LENGTH INTO *
: SEGMENTS OF A SIZE SUITABLE FOR TRANSMISSION OVER A DATA *
: LINK -- RETURNS BUFFER ADDRESS, BUFFER SIZE AND END_OF_MESSAGE *
: (BUFFER SIZE = 0 IS END_OF_MESSAGE) *
*****
ROUTINE GETOB <A,B>
LEAVE ENOUGH ROOM FOR A DDCMP HEADER
LET A(R5) := BAPNT + #10
INIT SIZE TO ZERO
LET B(R5) := #0
SCRTCH HAS MAXIMUM DDCMP MESSAGE SIZE
LET SCRTCH := BASIZE - #12
GET A COPY OF THE BUFFER POINTER
LET C(R4) := A(R5)
REPEAT LOOP UNTIL MAX SIZE OR END OF DATA
WHILE B(R5) LT SCRTCH AND INITMS GT #0 DO

```

```

1849 013240 003414                                BLE      50003$
1850
1851          :MOVE DATA INTO MESSAGE BUFFER
1852          :
1853 013242                                LET @C(R4) := @INITMA           ;[3.2.20.5.1 + 2]
1854 013242 117774 166716 177776                MOV      @INITMA,@C(R4)
1855 013250                                LET INITMS := INITMS - #1     ;[3.2.20.5.1]
1856 013250 005367 166712                DEC      INITMS
1857 013254                                LET INITMA := INITMA + #1
1858 013254 005267 166704                INC      INITMA
1859 013260                                LET B(R5) := B(R5) + #1     ;[3.2.20.5.2]
1860 013260 005265 000002                INC      B(R5)
1861 013264                                LET C(R4) := C(R4) + #1
1862 013264 005264 177776                INC      C(R4)
1863 013270                                ENDDO
1864 013270 000755                                BR       50002$
1865 013272                                50003$:
1866          :
1867          :                               END OF LOOP
1868          :
1869 013272                                ENDRTN
1870 013272                                50000$:
1871 013272                                50001$:
1872 013272 010406                                MOV      R4,SP
1873 013274 012604                                MOV      (SP)+,R4
1874 013276 000207                                RTS      PC
1875
1876
1877
1878
1879
1880          :*****
1881          :ROUTINE TO PROVIDE LINE PROTOCOL HEADER AND CHECKSUMS  +
1882          :CALL WITH DATA AREA ADDRESS AND SIZE (IN BYTES)      +
1883          :RETURN WITH MODIFIED ADDRESS AND SIZE                   +
1884          :*****
1885 013300                                ROUTINE PROBUF <A,B,C,D>     ;[3.2.20.6]
1886 013300                                PROBUF:
1887 013300                                PUSH R1
1888 013300 010146                                MOV      R1,-(SP)
1889 013302                                PUSH R2
1890 013302 010246                                MOV      R2,-(SP)
1891
1892          :CHANGE POINTER TO INCLUDE DDCMP HEADER
1893          :
1894 013304                                LET R2 := A(R5) - #10        ;[3.2.20.6.1]
1895 013304 016502 000000                MOV      A(R5),R2
1896 013310 162702 000010                SUB      #10,R2
1897
1898          :FIRST BYTE IS SOM (START OF HEADER)
1899          :
1900 013314                                LET (R2)+ := #SOM
1901 013314 112722 000201                MOV      #SOM,(R2)+
1902
1903          :NEXT TWO BYTES GET MESSAGE SIZE
1904          :

```

```

1905 013320          LET R1 := B(R5)
1906 013320 016501 000002          MOV     B(R5),R1
1907 013324          LET (R2)+ :B= R1
1908 013324 110122          MOVB   R1,(R2)+
1909 013326          LET R1 := SWAP R1
1910 013326 000301          SWAB   R1
1911 013330          LET (R2) :B= R1
1912 013330 110112          MOVB   R1,(R2)
1913
1914          ;SET SELECT BIT
1915
1916 013332          LET (R2) :B= (R2) SET.BY #200
1917 013332 152712 000200          BISB   #200,(R2)
1918 013336          LET R2 := R2 + #1
1919 013336 005202          INC    R2
1920
1921          ;NEXT BYTE IS LAST GOOD RESPONSE
1922
1923 013340          LET (R2)+ :B= RMSGNO
1924 013340 116722 166632          MOVB   RMSGNO,(R2)+
1925
1926          ;NEXT BYTE GETS MESSAGE NUMBER --- POP NUMBER FOR NEXT MESSAGE (IF ANY)
1927
1928 013344          LET MSGNO := MSGNO + #1
1929 013344 005267 166624          INC    MSGNO
1930 013350          LET (R2)+ :B= MSGNO
1931 013350 116722 166620          MOVB   MSGNO,(R2)+
1932
1933          ;GET THE TERMINAL ADDRESS FOR THE NEXT HEADER BYTE
1934
1935 013354          LET R1 := #PTABLE + #5
1936 013354 012701 002200          MOV    #PTABLE,R1
1937 013360 062701 000005          ADD    #5,R1
1938 013364          LET (R2)+ :B= (R1)
1939 013364 111122          MOVB   (R1),(R2)+
1940
1941          ;GET HEADER START ADDRESS AND SIZE IN BYTES
1942
1943 013366          LET SCRATCH := A(R5) - #10
1944 013366 016567 000000 166516          MOV    A(R5),SCRATCH
1945 013374 162767 000010 166510          SUB    #10,SCRATCH
1946 013402          LET TEMP := #6
1947 013402 012767 000006 166504          MOV    #6,TEMP
1948
1949          ;COMPUTE HEADER CRC
1950
1951 013410          CALL CRC IN <SCRATCH,TEMP> OUT <TEMP>          ;[3.2.20.6.2]
1952 013410 162705 000002          SUB    #1*2,R5
1953 013414 010546          MOV    R5,-(SP)
1954 013416 016745 166472          MOV    TEMP,-(R5)
1955 013422 016745 166464          MOV    SCRATCH,-(R5)
1956 013426 004767 003456          JSR   PC,CRC
1957 013432 012605          MOV    (SP)+,R5
1958 013434 012567 166454          MOV    (R5)+,TEMP
1959
1960          ;PUT CRC INTO BUFFER

```

```

1961
1962 013440      ;      LET (R2) := TEMP      ;[3.2.20.6.3]
1963 013440 016712 166450      MOV      TEMP,(R2)
1964
1965      ;GET ADDRESS OF MESSAGE AND MESSAGE SIZE IN BYTES
1966      ;
1967 013444      ;      LET SCRTCH := A(R5)
1968 013444 016567 000000 166440      MOV      A(R5),SCRTCH
1969 013452      ;      LET TEMP := B(R5)
1970 013452 016567 000002 166434      MOV      B(R5),TEMP
1971
1972      ;COMPUTE DATA CRC
1973      ;
1974 013460      ;      CALL CRC IN <SCRTCH,TEMP> PUT <TEMP>      ;[3.2.20.6.4]
1975 013460 162705 000002      SUB      #1*2,R5
1976 013464 010546      MOV      R5,-(SP)
1977 013466 016745 166422      MOV      TEMP,-(R5)
1978 013472 016745 166414      MOV      SCRTCH,-(R5)
1979 013476 004767 003406      JSR      PC,CRC
1980 013502 012605      MOV      (SP)+,R5
1981 013504 012567 166404      MOV      (R5)+,TEMP
1982
1983      ;PUT DATA CRC IN LAST TWO BYTES OF BUFFER
1984      ;
1985 013510      ;      LET R2 := A(R5) + B(R5)      ;[3.2.20.6.5]
1986 013510 016502 000000      MOV      A(R5),R2
1987 013514 066502 000002      ADD      B(R5),R2
1988 013520      ;      LET (R2)+ :B= TEMP
1989 013520 116722 166370      MOV      TEMP,(R2)+
1990 013524      ;      LET TEMP := SWAP TEMP
1991 013524 000367 166364      SWAB    TEMP
1992 013530      ;      LET (R2) :B= TEMP
1993 013530 116712 166360      MOV      TEMP,(R2)
1994
1995      ;RETURN WITH MODIFIED ADDRESS AND SIZE
1996      ;
1997 013534      ;      LET D(R5) := B(R5) + #12
1998 013534 016565 000002 000006      MOV      B(R5),D(R5)
1999 013542 062765 000012 000006      ADD      #12,D(R5)
2000 013550      ;      LET C(R5) := A(R5) - #10
2001 013550 016565 000000 000004      MOV      A(R5),C(R5)
2002 013556 162765 000010 000004      SUB      #10,C(R5)
2003 013564      ;      POP R2
2004 013564 012602      MOV      (SP)+,R2
2005 013566      ;      POP R1
2006 013566 012601      MOV      (SP)+,R1
2007 013570      ;      ENDRTN
2008 013570      ;
2009 013570      ;
2010 013570 000207      ;      50000$:
RTS      PC
2011
2012
2013
2014
2015
2016
;*****

```

```

2017          : ROUTINE TO GET TERMINAL INPUT
2018          : .....
2019 013572   : ROUTINE TERIN <A>                ;[3.2.1]
2020 013572   :                                TERIN:
2021          :
2022          : SAVE R1
2023          :
2024 013572   : PUSH R1
2025 013572 010146 :                                MOV    R1,-(SP)
2026          :
2027          : GET TERMINAL RESPONSE
2028          :
2029 013574   : CALL RCVE
2030 013574 004767 003026 :                                JSR    PC,RCVE
2031          :
2032          : IF RECEIVE ERROR, REPORT IT AND EXIT WITH THE ERROR FLAG SET
2033          :
2034 013600   : IF.ERROR THEN
2035 013600 103007 :                                BCC   50002$
2036 013602   :                                ERRHRD 40,MSG40,AREA40
2037 013602 104463 :                                TRAP  T$ERCODE
2038 013604 000050 :                                .WORD 40
2039 013606 002741 :                                .WORD MSG40
2040 013610 004242 :                                .WORD AREA40
2041          :
2042          : .....RESTORE R1 FIRST
2043          :
2044 013612   : POP R1
2045 013612 012601 :                                MOV   (SP)+,R1
2046 013614   :                                RETURN ERROR
2047 013614 000261 :                                SEC
2048 013616 000571 :                                BR    50001$
2049 013620   :
2050 013620   :                                ENDIF
2051          :                                50002$:
2052          :
2053          : IF MESSAGE IS ONLY 10 BYTES, SEE IF IT IS AN ACK.
2054          : IT'S OK TO BE AN ACK, BUT RECEPTION IS INCOMPLETE (CLEAR FLAG).
2055          :
2056 013620 026727 166334 000010 :                                IF INSIZ EQ #10 THEN
2057 013626 001027 :                                CMP   INSIZ,#10
2058 013630   :                                BNE   50003$
2059 013630 005065 000000 :                                LET A(R5) := #0
2060 013634   :                                CLR   A(R5)
2061 013634 012767 000001 177100 :                                LET TYPE := #ACK
2062 013642   :                                MOV   #ACK,TYPE
2063 013642 010546 :                                CALL CHKREP IN <TYPE>
2064 013644 016745 177072 :                                MOV   R5,-(SP)
2065 013650 004767 000330 :                                MOV   TYPE,-(R5)
2066 013654 012605 :                                JSR   PC,CHKREP
2067 013656   :                                MOV   (SP)+,R5
2068 013656 103003 :                                IF.ERROR THEN
2069 013660   :                                BCC   50004$
2070 013660 012601 :                                POP R'
2071 013662   :                                MOV   (SP)+,R1
2072 013662 000261 :                                RETURN FRROR
2072          :                                SEC

```



2073	013664	000546			BR	50001\$
2074	013666			ENDIF		
2075	013666				50004\$:	
2076	013666			LET R1 := INBUF + #4		
2077	013666	016701	166264		MOV	INBUF,R1
2078	013672	062701	000004		ADD	#4,R1
2079	013676			LET RMSGNO :B= (R1)		
2080	013676	111167	166274		MOVB	(R1),RMSGNO
2081	013702			POP R1		
2082	013702	012601			MOV	(SP)+,R1
2083	013704			RETURN		
2084	013704	000535			BR	50000\$
2085	013706			ENDIF		
2086	013706				50003\$:	
2087				:		
2088				:IF RECEIVE OK, CHECK THE HEADER CRC		
2089				:COMPUTE EXPECTED CRC		
2090				:		
2091	013706			CALL CRC IN <INBUF,#6> OUT <A04EX>		
2092	013706	162705	000002		SUB	#1*2,R5
2093	013712	010546			MOV	R5,-(SP)
2094	013714	012745	000006		MOV	#6,-(R5)
2095	013720	016745	166232		MOV	INBUF,-(R5)
2096	013724	004767	003160		JSR	PC,CRC
2097	013730	012605			MOV	(SP)+,R5
2098	013732	012567	170174		MOV	(R5)+,A04EX
2099				:		
2100				:GET THE CRC RECEIVED		
2101				:		
2102	013736			LET R1 := INBUF + #10		
2103	013736	016701	166214		MOV	INBUF,R1
2104	013742	062701	000010		ADD	#10,R1
2105	013746			LET A04RC :B= -(R1)		
2106	013746	114167	170156		MOVB	-(R1),A04RC
2107	013752			LET A04RC := SWAP A04RC		
2108	013752	000367	170152		SWAB	A04RC
2109	013756			LET A04RC :B= -(R1)		
2110	013756	114167	170146		MOVB	-(R1),A04RC
2111				:		
2112				:IF THEY AREN'T THE SAME, REPORT THE ERROR, RESTORE R1 AND EXIT WITH		
2113				:THE ERROR FLAG SET		
2114				:		
2115	013762			IF A04EX NE A04RC THEN		
2116	013762	026767	170144 170140		CMP	A04EX,A04RC
2117	013770	001407			BEQ	50005\$
2118	013772			ERRHRD 41,MSG41,AREA41		
2119	013772	104463			TRAP	T\$ERCODE
2120	013774	000051			.WORD	41
2121	013776	002463			.WORD	MSG41
2122	014000	004076			.WORD	AREA41
2123	014002			POP R1		
2124	014002	012601			MOV	(SP)+,R1
2125	014004			RETURN ERROR		
2126	014004	000261			SEC	
2127	014006	000475			BR	50001\$
2128	014010			ENDIF		

```

2129 014010                                50005$:
2130
2131      ; COMPARE TERMINAL ADDRESS RECEIVED WITH THAT EXPECTED
2132      ; AND IF THEY AREN'T THE SAME, REPORT THE ERROR, RESTORE R1 AND
2133      ; EXIT WITH THE ERROR FLAG SET
2134
2135      ;FB -(R1) NE STRTSA THEN
2136 014010 124167 167641                    CMPB   -(R1),STRTSA
2137 014014 001407                          BEQ    50006$
2138 014016                                ERRHRD 42,MSG42,AREA42
2139 014016 104463                          TRAP   T$ERCODE
2140 014020 000052                          .WORD 42
2141 014022 002417                          .WORD MSG42
2142 014024 004044                          .WORD AREA42
2143 014026                                POP R1
2144 014026 012601                          MOV    (SP)+,R1
2145 014030                                RETURN ERROR
2146 014030 000261                          SEC
2147 014032 000463                          BR     50001$
2148 014034                                ENDIF
2149 014034                                50006$:
2150
2151      ; MOVE THE POINTERS PAST THE HEADER AND COMPUTE THE DATA CRC
2152
2153 014034                                LET INBUF := INBUF + #10
2154 014034 062767 000010 166114            ADD    #10,INBUF
2155 014042                                LET INSIZ := INSIZ - #12
2156 014042 162767 000012 166110            SUB    #12,INSIZ
2157 014050                                CALL CRC IN <INBUF,INSIZ> OUT <A04EX>
2158 014050 162705 000002                    SUB    #1*2,R5
2159 014054 010546                          MOV    R5,-(SP)
2160 014056 016745 166076                    MOV    INSIZ,-(R5)
2161 014062 016745 166070                    MOV    INBUF,-(R5)
2162 014066 004767 003016                    JSR    PC,CRC
2163 014072 012605                          MOV    (SP)+,R5
2164 014074 012567 170032                    MOV    (R5)+,A04EX
2165
2166      ; GET THE DATA CRC RECEIVED
2167
2168 014100                                LET R1 := INBUF + INSIZ
2169 014100 016701 166052                    MOV    INBUF,R1
2170 014104 066701 166050                    ADD    INSIZ,R1
2171 014110                                LET R1 := R1 + #2
2172 014110 062701 000002                    ADD    #2,R1
2173 014114                                LET A04RC :B= -(R1)
2174 014114 114167 170010                    MOV    -(R1),A04RC
2175 014120                                LET A04RC := SWAP A04RC
2176 014120 000367 170004                    SWAB  A04RC
2177 014124                                LET A04RC :B= -(R1)
2178 014124 114167 170000                    MOV    -(R1),A04RC
2179
2180      ; IF THEY AREN'T THE SAME; REPORT THE ERROR, RESTORE R1 AND
2181      ; EXIT WITH THE ERROR FLAG SET
2182
2183 014130                                IF A04RC NE A04EX THEN
2184 014130 026767 167774 167774            CMP    A04RC,A04EX

```

```
2185 014136 001407                                BEQ      50007$
2186 014140                                ERRHRD 43,MSG43,AREA43
2187 014140 104463                                TRAP    T$ERCODE
2188 014142 000053                                .WORD  43
2189 014144 003402                                .WORD  MSG43
2190 014146 004076                                .WORD  AREA43
2191 014150                                POP R1
2192 014150 012601                                MOV     (SP)+,R1
2193 014152                                RETURN ERROR
2194 014152 000261                                SEC
2195 014154 000412                                BR     50001$
2196 014156                                ENDIF
2197 014156                                50007$:
2198
2199 ; OTHERWISE; SFT COMPLETE FLAG, RESTORE R' AND EXIT NORMALLY
2200 ;
2201 014156                                LET A(R5) := A(R5) + #1
2202 014156 005265 00U00C                                INC     A(R5)
2203 014162                                LET R1 :- INBUF - #4
2204 014162 016701 165770                                MOV     INBUF,R1
2205 014166 162701 000004                                SUB     #4,R1
2206 014172                                LET RMSGNO :B- (R1)
2207 014172 111167 166000                                MOV     (R1),RMSGNO
2208 014176                                POP R1
2209 014176 012601                                MOV     (SP)+,R1
2210 014200                                ENDRTN
2211 014200                                50000$:
2212 014200 000241                                CLC
2213 014202                                50001$:
2214 014202 000207                                RTS     PC
2215
2216
2217
2218
2219
2220
2221
2222 *****
2223 ROUTINE TO CHECK AN INPUT BUFFER FOR A STANDARD *
2224 RESPONSE ("ACK" OR "STACK") *
2225 *
2226 LOCATION "TYPE" CONTAINS THE RESPONSE CODE EXPECTED *
2227 *
2228 INPUT BUFFER POINTER IS IN "INBUF" *
2229 *
2230 INPUT BUFFER SIZE (IN BYTES) IS IN "INSIZ" *
2231 *****
2231 014204 ROUTINE CHKREP <B> ;[3.2.20.3, 3.2.20.9]
2232 014204                                CHKREP:
2233 014204 DECLARE LOCALS
2234 014204 LOCAL <A>
2235 014204 ALLOCATE
2236 014204 010446                                MOV     R4, -(SP)
2237 014206 010604                                MOV     SP,R4
2238 014210 162706 000002                                SUB     #2,SP
2239
2240 ; IF MESSAGE IS NOT 10 BYTES LONG, IT CAN'T BE A CONTROL MESSAGE
```

```

2241      ;TYPE AN ERROR MESSAGE AND EXIT WITH ERROR SET
2242      ;
2243      IF INSIZE NE #10 THEN
2244      014214      026727      165740      000010      CMP      'INSIZ,#10
2245      014222      001406      BEQ      50002$
2246      014224      FRRHRD 01,MSG01,AREA01      ;[3.2.20.3.4, 3.2.20.9.4]
2247      014224      104463      TRAP     T$ERCODE
2248      014226      000001      .WORD   1
2249      014230      002303      .WORD   MSG01
2250      014232      003740      .WORD   AREA01
2251      014234      RETURN ERROR
2252      014234      000261      SEC
2253      014236      000553      BR      50001$
2254      014240
2255      014240      ELSE
2256      50002$:
2257      ;OTHERWISE, COMPUTE THE CRC AND CHECK IT AGAINST THAT SENT
2258      ;
2259      014240      CALL CRC IN <INBUF,#6> OUT <TEMP>      ;[3.2.20.3.1, 3.2.20.9.1]
2260      014240      162705      000002      SUB      #1+2,R5
2261      014244      010546      MOV      R5,-(SP)
2262      014246      012745      000006      MOV      #6,-(R5)
2263      014252      016745      165700      MOV      INBUF,-(R5)
2264      014256      004767      002626      JSR     PC,CRC
2265      014262      012605      MOV      (SP)+,R5
2266      014264      012567      165624      MOV      (R5)+,TEMP
2267      014270      LET A(R4) := INBUF + #7
2268      014270      016764      165662      177776      MOV      INBUF,A(R4)
2269      014276      062764      000007      177776      ADD      #7,A(R4)
2270      014304      LET SCRTCH :B= @A(R4)
2271      014304      117467      177776      165600      MOV     @A(R4),SCRTCH
2272      014312      LET SCRTCH := SWAP SCRTCH
2273      014312      000367      165574      SWAB   SCRTCH
2274      014316      LET A(R4) := A(R4) - #1
2275      014316      005364      177776      DEC    A(R4)
2276      014322      LET SCRTCH :B= @A(R4)
2277      014322      117467      177776      165562      MOV     @A(R4),SCRTCH
2278
2279      ;IF CRC'S MATCH GO ON... OTHERWISE SKIP FOLLOWING CODE
2280      ;
2281      014330      IF TEMP EQ SCRTCH THEN      ;[3.2.20.3.2, 3.2.20.9.2]
2282      014330      026767      165560      165554      CMP     TEMP,SCRTCH
2283      014336      001076      BNE     50004$
2284
2285      ;COMPARE CONTROL MESSAGE TYPE WITH THAT PASSED AS AN ARGUMENT
2286      ;
2287      014340      LET A(R4) := INBUF + #1
2288      014340      016764      165612      177776      MOV     INBUF,A(R4)
2289      014346      005264      177776      INC    A(R4)
2290      014352      LET A(R4) :B= @A(R4)
2291      014352      117464      177776      177776      MOV     @A(R4),A(R4)
2292      014360      LET A(R4) := A(R4) CLR.BY #HYBYTE
2293      014360      042764      177400      177776      BIC    #HYBYTE,A(R4)
2294
2295      ;IF NOT THE EXPECTED TYPE, PRINT ERROR MESSAGE AND EXIT WITH ERROR SET
2296      ;

```

```

2297 014366                                IF A(R4) NE B(R5) THEN                ;[3.2.20.3.3, 3.2.20.9.3]
2298 014366 026465 177776 000000                                CMP      A(R4),B(R5)
2299 014374 001414                                BEQ      500058
2300 014376                                LET A02A := A(R4)
2301 014376 016467 177776 167370                                MOV      A(R4),A02A
2302 014404                                LET A02B := B(R5)
2303 014404 016567 000000 167364                                MOV      B(R5),A02B
2304 014412                                ERRHRD 02,MSG02,AREA02                ;[3.2.20.3.4, 3.2.20.9.4]
2305 014412 104463                                TRAP    T$ERCODE
2306 014414 000002                                .WORD  2
2307 014416 002330                                .WORD  MSG02
2308 014420 003742                                .WORD  AREA02
2309 014422                                RETURN ERROR
2310 014422 000261                                SEC
2311 014424 000460                                BR      500018
2312
2313 ; OTHERWISE, CHECK TERMINAL ADDRESS
2314 ;
2315 014426                                ELSE
2316 014426                                500058:
2317 014426                                LET SCRTCH := #PTABLE + #5
2318 014426 012767 002200 165456                                MOV      #PTABLE,SCRTCH
2319 014434 062767 000005 165450                                ADD      #5,SCRTCH
2320 014442                                LET SCRTCH :B= @SCRTCH
2321 014442 117767 165444 165442                                MOV      @SCRTCH,SCRTCH
2322 014450                                LET SCRTCH := SCRTCH CLR.BY #HYBYTE
2323 014450 042767 177400 165434                                BIC      #HYBYTE,SCRTCH
2324 014456                                LET TEMP := INBUF + #5
2325 014456 016767 165474 165430                                MOV      INBUF,TEMP
2326 014464 062767 000005 165422                                ADD      #5,TEMP
2327 014472                                LET TEMP :B= @TEMP
2328 014472 117767 165416 165414                                MOV      @TEMP,TEMP
2329 014500                                LET TEMP := TEMP CLR.BY #HYBYTE
2330 014500 042767 177400 165406                                BIC      #HYBYTE,TEMP
2331
2332 ; IF NOT THE EXPECTED TERMINAL ADDRESS, THEN REPORT THE ERROR AND
2333 ; EXIT WITH ERROR SET
2334 ;
2335 014506                                IF TEMP NE SCRTCH THEN
2336 014506 026767 165402 165376                                CMP      TEMP,SCRTCH
2337 014514 001406                                BEQ      500078
2338 014516                                ERRHRD 03,MSG03,AREA03                ;[3.2.20.3.4, 3.2.20.9.4]
2339 014516 104463                                TRAP    T$ERCODE
2340 014520 000003                                .WORD  3
2341 014522 002417                                .WORD  MSG03
2342 014524 004044                                .WORD  AREA03
2343 014526                                RETURN ERROR
2344 014526 000261                                SEC
2345 014530 000416                                BR      500018
2346 014532                                ENDIF
2347 014532                                500078:
2348 014532                                ENDIF
2349 014532                                500068:
2350
2351 ; IF CRC WAS IN ERROR, SKIP ABOVE CODE, COME TO HERE, REPORT ERROR AND
2352 ; EXIT WITH ERROR SET

```

```

2353
2354 014532          ;
2355 014532 000414          ELSE
2356 014534          500048: BR      500108
2357 014534          LET A04RC :- SCRICH
2358 014534 016767 165352 167366          MOV      SCRICH,A04RC
2359 014542          LET A04EX :- TEMP
2360 014542 016767 165346 167362          MOV      TEMP,A04EX
2361 014550          ERRHRD 04,MSG04,AREA04          ;[3.2.20.3.4, 3.2.20.9.4]
2362 014550 104463          TRAP   T$ERCODE
2363 014552 000004          .WORD  4
2364 014554 002463          .WORD  MSG04
2365 014556 004076          .WORD  AREA04
2366 014560          RETURN ERROR
2367 014560 000261          SEC
2368 014562 000401          BR      500018
2369 014564          ENDIF
2370 014564          500108:
2371 014564          ENDIF
2372 014564          500038:
2373
2374          ;NORMAL RETURN
2375          ;
2376 014564          ENDRTN
2377 014564          500008:
2378 014564 000241          CLC
2379 014566          500018:
2380 014566 010406          MOV      R4,SP
2381 014570 012604          MOV      (SP)+,R4
2382 014572 000207          RTS      PC
2383
2384
2385
2386
2387
2388
2389          ;*****
2390          ; THIS ROUTINE SETS UP AND CHECKS FOR COMPLETE TRANSMISSION
2391          ; RETURNS AN ERROR FLAG IF ANY ERROR OCCURS DURING TRANSMISSION
2392          ; SET "ABORT" IF INITIALIZATION OF THE DEVICE FAILS
2393          ;*****
2393 014574          ROUTINE XMIT          ;[3.2.20.7 AND 3.2.20.1]
2394 014574          XMIT:
2395 014574          DECLARE LOCALS
2396 014574          LOCAL <B>
2397 014574          ALLOCATE
2398 014574 010446          MOV      R4,-(SP)
2399 014576 010604          MOV      SP,R4
2400 014600 162706 000002          SUB      #2,SP
2401
2402          ;CLEAR ERROR COUNT, TRANSMIT DONE FLAG AND RECEIVE DONE FLAG
2403          ;
2404 014604          LET B(R4) :- #0
2405 014604 005064 177776          CLR      B(R4)
2406 014610          LET XDONE :- #0
2407 014610 005067 000270          CLR      XDONE
2408 014614          LET RDONE :- #0

```

```

2409 014614 005067 000266                                CLR    RDONE
2410
2411                                START OF LOOP
2412                                :
2413                                REPEAT
2414 014620                                50002$:
2415                                :
2416                                TRY TO INIT DEVICE
2417                                :
2418 014620                                CALL DEVINIT
2419 014620 004767 000264                                JSR    PC,DEVINIT
2420                                :
2421                                IF ERROR OCCURRED....
2422                                :
2423 014624                                IF .ERROR THEN
2424 014624 103033                                BCC   50003$
2425 014626                                PUSH R1
2426 014626 010146                                MOV   R1,-(SP)
2427 014630                                PUSH R2
2428 014630 010246                                MOV   R2,-(SP)
2429                                :
2430                                GET A COPY OF THE DEVICE REGISTERS
2431                                :
2432 014632                                LET R1 := PTABLE
2433 014632 016701 165342                                MOV   PTABLE,R1
2434 014636                                LET R2 := #DUPREG
2435 014636 012702 002234                                MOV   #DUPREG,R2
2436 014642                                LET RCVCSR(R2) := RCVCSR(R1)
2437 014642 016162 000000 000000                                MOV   RCVCSR(R1),RCVCS
2438 014650                                LET RCVBUF(R2) := RCVBUF(R1)
2439 014650 016162 000002 000002                                MOV   RCVBUF(R1),RCVBU
2440 014656                                LET XMTCSR(R2) := XMTCSR(R1)
2441 014656 016162 000004 000004                                MOV   XMTCSR(R1),XMTCS
2442 014664                                LET XMTBUF(R2) := XMTBUF(R1)
2443 014664 016162 000006 000006                                MOV   XMTBUF(R1),XMTBU
2444 014672                                POP R2
2445 014672 012602                                MOV   (SP)+,R2
2446 014674                                POP R1
2447 014674 012601                                MOV   (SP)+,R1
2448                                :
2449                                REPORT THE ERROR AND UPDATE THE ERPOR COUNT
2450                                :
2451 014676                                ERRHRD 23,MSG23,AREA23
2452 014676 104463                                TRAP  T$ERCODE
2453 014700 000027                                .WORD 23
2454 014702 003024                                .WORD MSG23
2455 014704 004242                                .WORD AREA23
2456 014706                                LET B(R4) := B(R4) + #1
2457 014706 005264 177776                                INC   B(R4)
2458                                :
2459                                OTHERWISE, CLEAR THE ERROR COUNT
2460                                :
2461 014712                                ELSE
2462 014712 000402                                BR    50004$
2463 014714                                50003$:
2464 014714                                LET B(R4) := #()

```

```

2465 014714 005064 177776                                CLR      B(R4)
2466 014720                                ENDIF
2467 014720                                50004$:
2468
2469      : IF ERROR COUNT HAS EXCEEDED LIMITS, ABORT AND NEGATE THE COUNT
2470      :
2471 014720                                IF B(R4) EQ ILIMIT THEN
2472 014720 026467 177776 165176                (MP      B(R4),ILIMIT
2473 014726 001010                                BNE      50005$
2474 014730                                ERRDF 00,MSG00,AREA00
2475 014730 104462                                TRAP     T$ERCODE
2476 014732 000000                                .WORD   0
2477 014734 002244                                .WORD   MSG00
2478 014736 003660                                .WORD   AREA00
2479 014740                                LET ABORT := ABORT + #1
2480 014740 005267 165152                                INC      ABORT
2481 014744                                LET B(R4) := NEGATE B(R4)
2482 014744 005464 177776                                NEG      B(R4)
2483 014750                                ENDIF
2484 014750                                50005$:
2485
2486      :                               END OF LOOP
2487      : EXIT LOOP WHEN ERROR COUNT IS ZERO OR NEGATIVE (ABORT)
2488      :
2489 014750                                UNTIL B(R4) LE #0
2490 014750 005764 177776                                TST     B(R4)
2491 014754 003321                                BGT     50002$
2492
2493      : IF NOT ABORT THEN EXECUTE THE FOLLOWING CODE
2494      : OTHERWISE SKIP AND RETURN WITH ERROR SET
2495      :
2496 014756                                IF B(R4) EQ #0 THEN
2497 014756 005764 177776                                TST     B(R4)
2498 014762 001042                                BNE     50006$
2499
2500      : WAIT FOR TRANSMIT DONE FLAG TO SET
2501      :
2502 014764                                REPEAT
2503 014764                                BREAK
2504 014764                                50007$:
2505 014764 104022                                EMT     C$BRK
2506 014766                                UNTIL XDONE NE #0
2507 014766 005767 000112                                TST     XDONE
2508 014772 001774                                BEQ     50007$
2509
2510      : IF TRANSMIT FLAG IS NEGATIVE, AN ERROR OCCURRED.
2511      :
2512 014774                                IF XDONE LT #0 THEN
2513 014774 005767 000104                                TST     XDONE
2514 015000 002032                                BGE     50010$
2515 015002                                PUSH R1
2516 015002 010146                                MOV     R1,-(SP)
2517 015004                                PUSH R2
2518 015004 010246                                MOV     R2,-(SP)
2519
2520      : IF ERROR, GET COPY OF DEVICE REGISTERS, PRINT MESSAGE AND EXIT WITH ERROR SET
  
```



```
2521 ; OTHERWISE DO A NORMAL EXIT
2522 ;
2523 015006 LET R1 := PTABLE
2524 015006 016701 165166 MOV PTABLE,R1
2525 015012 LET R2 := #DUPREG
2526 015012 012702 002234 MOV #DUPREG,R2
2527 015016 LET RCVCSR(R2) := RCVCSR(R1)
2528 015016 016162 000000 000000 MOV RCVCSR(R1),RCVCS
2529 015024 LET RCVBUF(R2) := RCVBUF(R1)
2530 015024 016162 000002 000002 MOV RCVBUF(R1),RCVBU
2531 015032 LET XMTCSR(R2) := XMTCSR(R1)
2532 015032 016162 000004 000004 MOV XMTCSR(R1),XMTCS
2533 015040 LET XMTBUF(R2) := XMTBUF(R1)
2534 015040 016162 000006 000006 MOV XMTBUF(R1),XMTBU
2535 015046 POP R2
2536 015046 012602 MOV (SP)+,R2
2537 015050 POP R1
2538 015050 012601 MOV (SP)+,R1
2539 015052 ERRHRD 24,MSG24,AREA24
2540 015052 104463 TRAP I$ERCODE
2541 015054 000030 .WORD 24
2542 015056 003071 .WORD MSG24
2543 015060 004242 .WORD AREA24
2544 015062 RETURN ERROR
2545 015062 000261 SEC
2546 015064 000404 BR 50001$
2547 015066 ENDIF
2548 015066 50010$:
2549 015066 ELSE
2550 015066 000402 BR 50011$
2551 015070 50006$:
2552 015070 RETURN ERROR
2553 015070 000261 SEC
2554 015072 000401 BR 50001$
2555 015074 ENDIF
2556 015074 50011$:
2557 015074 ENDRTN
2558 015074 50000$:
2559 015074 000241 CLC
2560 015076 50001$:
2561 015076 010406 MOV R4,SP
2562 015100 012604 MOV (SP)+,R4
2563 015102 000207 RTS PC
2564 015104 000000
2565 015106 000000
2566
2567
2568
2569
2570
2571 *****
2572 ROUTINE TO INITIALIZE A DEVICE
2573 IF "INIT" IS NON-ZERO, THE ROUTINE WILL EXECUTE AND SET IT
2574 RETURNS WITH AN ERROR IF CTS CANNOT BE SET
2575 *****
2576
```

```

2577
2578
2579
2580
2581
2582 015110
2583 015110
2584 015110
2585 015110
2586 015110
2587 015110 010446
2588 015112 010604
2589 015114 162706 000002
2590 015120
2591 015120 010146
2592
2593
2594
2595 015122
2596 015122 016701 165052
2597
2598
2599
2600
2601 015126
2602 015126 052761 000400 000004
2603 015134
2604 015134 052761 000002 000000
2605 015142
2606 015142 052761 101226 000002
2607
2608
2609
2610 015150
2611 015150 005067 000402
2612
2613
2614
2615 015154
2616 015154
2617
2618
2619
2620 015154
2621 015154 052711 000004
2622 015160
2623 015160 012700 000100
2624 015164 104027
2625
2626
2627
2628 015166
2629 015166 011164 177776
2630 015172 016446 177776
2631 015176 042716 020000
2632 015202 042664 177776

```

RETURNS WITH ERROR AND "ABORT" SET IF THE DEVICE DOES NOT EXIST  
 .....  
 ROUTINE DEVINIT ;[3.2.20.1.4 AND 3.2.20.7.1] DEVINIT:  
 DECLARE LOCALS LOCAL <A>  
 ALLOCATE  
 MOV R4,-(SP)  
 MOV SP,R4  
 SUB #2,SP  
 PUSH R1  
 MOV R1,-(SP)  
 GET THE DEVICE ADDRESS  
 LET R1 := PTABLE  
 MOV PTABLE,R1  
 DO MASTER CLEAR, SET DATA TERMINAL READY, AND SET PARAMETER REGISTER (DEC MODE, AUTO-CRC INHIBIT, SYNC CHARACTER).  
 LET XMTCSR(R1) := XMTCSR(R1) SET.BY #MASCLR  
 BIS #MASCLR,XMTCSR(R1)  
 LET RCVCSR(R1) := RCVCSR(R1) SET.BY #DTR  
 BIS #DTR,RCVCSR(R1)  
 LET RCVBUF(R1) := RCVBUF(R1) SET.BY #PARAM  
 BIS #PARAM,RCVBUF(R1)  
 SET TIMER TO ZERO  
 LET TIMER := #0  
 CLR TIMER  
 LOOP  
 REPEAT 500028:  
 SET REQUEST TO SEND AND WAIT 10000 MICROSECS  
 LET (R1) := (R1) SET.BY #RTS  
 BIS #RTS,(R1)  
 WAITUS #100  
 MOV #100,R0  
 EMT (SWTU)  
 GET THE CLEAR TO SEND BIT AND UPDATE THE TIMER  
 LET A(R4) := (R1) AND #CTS  
 MOV (R1),A(R4)  
 MOV A(R4),-(SP)  
 BIC #CTS,(SP)  
 BIC (SP)+,A(R4)

```

2633 015206          LET TIMER : TIMER + #1
2634 015206 005267 000344          INC      TIMER
2635
2636          END LOOP
2637          LOOP ENDS WHEN TIMER EXCEEDS LIMIT OR CLEAR TO SEND IS UP
2638
2639 015212          UNTIL TIMER EQ TIMLIM OR A(R4) NE #0
2640 015212 026767 000340 164700          CMP      TIMER,TIMLIM
2641 015220 001403          BEQ      50003$
2642 015222 005764 177776          TST     A(R4)
2643 015226 001752          BEQ      50002$
2644 015230
2645
2646          50003$:
2647          IF TIME LIMIT NOT EXCEEDED...
2648 015230          IF TIMER NE TIMLIM THEN
2649 015230 026767 000322 164662          CMP      TIMER,TIMLIM
2650 015236 001507          BEQ      50004$
2651
2652          GET THE DEVICE PRIORITY AND SET UP THE RECEIVE INTERRUPT HANDLER
2653
2654 015240          LET DEVPRI := #PTABLE + #4
2655 015240 012767 002200 164742          MOV     #PTABLE,DEVPRI
2656 015246 062767 000004 164734          ADD     #4,DEVPRI
2657 015254          LET DEVPRI :B= @DEVPRI
2658 015254 117767 164730 164726          MOVB  @DEVPRI,DEVPRI
2659 015262          LET DEVPRI := DEVPRI CLR.BY #HYBYTE
2660 015262 042767 177400 164720          BIC     #HYBYTE,DEVPRI
2661 015270          LET A(R4) := #PTABLE + #2
2662 015270 012764 002200 177776          MOV     #PTABLE,A(R4)
2663 015276 062764 000002 177776          ADD     #2,A(R4)
2664 015304          SETVEC @A(R4),#RINT,DEVPRI
2665 015304 016746 164700          MOV     DEVPRI,-(SP)
2666 015310 012746 016002          MOV     #RINT,-(SP)
2667 015314 017446 177776          MOV     @A(R4),-(SP)
2668 015320 012746 000003          MOV     #3,-(SP)
2669 015324 104037          EMT     C$SVEC
2670 015326 062706 000010          ADD     #10,SP
2671
2672          CLEAR THE NEW MESSAGE FLAG
2673
2674 015332          LET NMSG := #0
2675 015332 005067 000236          CLR     NMSG
2676 015336          LET A(R4) := @A(R4) + #4
2677 015336 017464 177776 177776          MOV     @A(R4),A(R4)
2678 015344 062764 000004 177776          ADD     #4,A(R4)
2679
2680          SET UP THE TRANSMIT INTERRUPT HANDLER
2681
2682 015352          SETVEC A(R4),#XINT,DEVPRI
2683 015352 016746 164632          MOV     DEVPRI,-(SP)
2684 015356 012746 015634          MO:    #XINT,-(SP)
2685 015362 016446 177776          MOV     A(R4),-(SP)
2686 015366 012746 000003          MOV     #3,-(SP)
2687 015372 104037          EMT     C$SVEC
2688 015374 062706 000010          ADD     #10,SP

```

```

2689
2690      :SET UP COPIES OF THE DEVICE BUFFER REGISTER ADDRESSES
2691      :
2692      LET XMTADD := R1 + #6
2693      MOV      R1,XMTADD
2694      015400 010167 000146      ADD      #6,XMTADD
2695      015404 062767 000006 000140
2696      LET RCVADD := R1 + #2
2697      MOV      R1,RCVADD
2698      015412 010167 000136      ADD      #2,RCVADD
2699      015416 062767 000002 000130
2700      :SET COUNTERS FOR FOUR SYNC'S AND FOUR FILL CHARACTERS
2701      :
2702      LET SYNCNT := #4
2703      MOV      #4,SYNCNT
2704      015424 012767 000004 000126
2705      LET FLLCNT := #4
2706      MOV      #4,FLLCNT
2707      015432 012767 000004 000122
2708      :ENABLE TRANSMIT AND RECEIVE INTERRUPTS...SET TO FULL DUPLEX
2709      :
2710      LET XMTCSR(R1) := XMTCSR(R1) SET.BY #TXINTE!SENDFD
2711      BIS      #TXINTE!SENDFD,X
2712      LET RCVCSR(R1) := RCVCSR(R1) SET.BY #RXINTE
2713      BIS      #RXINTE,RCVCSR(R
2714      :IF TIME LIMIT WAS EXCEEDED, GET REGISTER CONTENTS, PRINT MESSAGE,
2715      :AND RETURN WITH ERROR SET
2716      :
2717      ELSE
2718      BR      50005$
2719      50004$:
2720      PUSH R2
2721      MOV      R2,-(SP)
2722      LET R2 := #DUPREG
2723      MOV      #DUPREG,R2
2724      LET RCVCSR(R2) := RCVCSR(R1)
2725      MOV      RCVCSR(R1),RCVCS
2726      LET RCVBUF(R2) := RCVBUF(R1)
2727      MOV      RCVBUF(R1),RCVBU
2728      LET XMTCSR(R2) := XMTCSR(R1)
2729      MOV      XMTCSR(R1),XMTCS
2730      LET XMTBUF(R2) := XMTBUF(R1)
2731      MOV      XMTBUF(R1),XMTBU
2732      POP R2
2733      MOV      (SP)+,R2
2734      POP R1
2735      MOV      (SP)+,R1
2736      ERRHRD 30,MSG30,AREA30
2737      TRAP   T$ERCODE
2738      .WORD 30
2739      .WORD MSG30
2740      .WORD AREA30
2741      RETURN ERROR
2742      SEC
2743      BR      50001$
2744      50005$:

```

ENDIF

50005\$:

```

2745
2746      ; IF NO ERROR, DO NORMAL EXIT
2747      ;
2748      POP R1
2749      015534 012601      MOV      (SP)+,R1
2750      015536      ;
2751      015536 005267 000022      LET INIT := INIT + #1      INC      INIT
2752      015542      ;
2753      015542      ;
2754      015542 000241      50000$:      CLC
2755      015544      ;
2756      015544 010406      50001$:      MOV      R4,SP
2757      015546 012604      MOV      (SP)+,R4
2758      015550 000207      RTS      PC
2759      ;
2760      ;
2761      015552 000000      XMTADD: 0      ; TRANSMIT BUFFER REGISTER ADDRESS
2762      015554 000000      RCVADD: 0      ; RECEIVE BUFFER REGISTER ADDRESS
2763      015556 000000      TIMER: 0      ; TIMER FOR CLEAR TO SEND OPERATION
2764      015560 000000      SYNCNT: 0      ; NUMBER OF SYNC CHARACTERS INSERTED BEFORE DDCMP MESSAGE
2765      015562 000000      FLLCNT: 0      ; NUMBER OF FILL CHARACTERS AFTER MESSAGE
2766      015564 000000      INIT: 0      ; INITIALIZE COMPLETE FLAG
2767      015566 000000      FIL: 0      ; FILL CHARACTER
2768      015570 000000      XBUFFER: 0      ; CURRENT TRANSMIT BUFFER POINTER
2769      015572 000000      XSIZE: 0      ; CURRENT TRANSMIT BUFFER SIZE
2770      015574 000000      NMSG: 0      ; NEW MESSAGE EXPECTED FLAG
2771      015576 000000      RTEMP1: 0      ; CURRENT RECEIVE BUFFER POINTER
2772      015600 000000      RTEMP2: 0      ; CURRENT RECEIVE BUFFER SIZE
2773      015602 000000      CNTFLG: 0      ; CONTROL FLAG FOR INPUTTING DDCMP MESSAGES
2774      ;
2775      ;
2776      ;
2777      015604 000000      ; RECEIVER QUEUES
2778      015606 000000      RQUEUE: 0      ; QUEUE POINTER FOR BUFFER PROCESSING
2779      015610 000000      RQUE: 0      ; QUEUE POINTER FOR NEXT AVAILABLE BUFFER
2780      015612 000000      RQUET: 0      ; QUEUE OF BUFFER ADDRESSES AND SIZES
2781      015614 000000      0
2782      015616 000000      0
2783      015620 000000      0
2784      015622 000000      RQUEB: 0
2785      015624 000000      RBQUE: 0      ; QUEUE POINTER FOR NEXT AVAILABLE BUFFER AREA
2786      015626 000000      RBQUET: 0      ; QUEUE OF BUFFER AREAS
2787      015630 000000      0
2788      015632 000000      RBQUEB: 0

```

```

2789
2790
2791
2792 015634
2793
2794
2795
2796 015634
2797 015634 016700 164350
2798 015640 104041
2799
2800
2801
2802 015642
2803 015642 005767 177712
2804 015646 001047
2805
2806
2807
2808
2809 015650
2810 015650 010146
2811 015652
2812 015652 005767 177714
2813 015656 003416
2814
2815
2816
2817 015660
2818 015660 117701 177704
2819 015664
2820 015664 042701 177400
2821 015670
2822 015670 042777 000400 177654
2823 015676
2824 015676 110177 177650
2825 015702
2826 015702 005267 177662
2827 015706
2828 015706 005367 177660
2829
2830
2831
2832 015712
2833 015712 000423
2834 015714
2835 015714
2836 015714 005767 177642
2837 015720 001406
2838
2839
2840
2841 015722
2842 015722 056777 177640 177622
2843 015730
2844 015730 005367 177626

```

```

:////////////////////
:TRANSMISSION INTERRUPT HANDLER
:////////////////////
BGNSRV XINT ;[3.2.20.1.7 AND 3.2.20.7.4]

:SET PROCESSOR PRIORITY LEVEL
:
SFTPRI DEVPRI
MOV EMT DEVPRI,RO
CSSPRI

:IF SYNC COUNT EQUAL TO ZERO...
:
IF SYNCNT EQ #0 THEN
TST SYNCNT
BNE 50006$

:
CHECK BUFFER SIZE.
IF GREATER THAN ZERO, THEN....
:
PUSH R1
MOV R1,-(SP)
IF XSIZE GT #0 THEN
TST XSIZE
BLE 50007$

:
SEND THE NEXT CHARACTER AND UPDATE POINTER AND COUNT
:
LET R1 :B= @XBUFFER
MOV B @XBUFFER,R1
LET R1 := R1 CLR.BY #HYBYTE
BIC #HYBYTE,R1
LET @XMTADD := @XMTADD CLR.BY #TSOM
BIC #TSOM,@XMTADD
LET @XMTADD :B= R1
MOV B R1,@XMTADD
LET XBUFFER := XBUFFER + #1
INC XBUFFER
LET XSIZE := XSIZE - #1
DEC XSIZE

:
IF ALL CHARACTERS SENT, CHECK FILL COUNT.
:
ELSE
BR 50010$
50007$:
IF FLLCNT NE #0 THEN
TST FLLCNT
BEQ 50011$

:
IF FILL COUNT ISN'T ZERO SEND A FILL CHAR. AND DECREMENT COUNT
:
LET @XMTADD := @XMTADD SET.BY FIL
BIS FIL,@XMTADD
LET FLLCNT := FLLCNT - #1
DEC FLLCNT

```

```

2845 015734
2846 015734 000412
2847 015736
2848
2849
2850
2851 015736
2852 015736 016701 177610
2853 015742 162701 000006
2854 015746
2855 015746 005267 177132
2856 015752
2857 015752 104021
2858 015754
2859 015754 042761 000120 000004
2860 015762
2861 015762
2862 015762
2863 015762
2864 015762
2865 015762 012601
2866
2867
2868
2869 015764
2870 015764 000405
2871 015766
2872 015766
2873 015766 052777 000626 177556
2874 015774
2875 015774 005367 177560
2876 016000
2877 016000
2878
2879
2880
2881 016000
2882 016000
2883 016000 000002
2884
2885
2886
2887
2888
2889
2890
2891
2892 016002
2893
2894
2895
2896 016002
2897 016002 016700 164202
2898 016006 104041
2899 016010
2900 016010 010146

```

```

ELSE
50011$: BR 50012$
IF ALL TRANSMIT DONE, DISABLE TRANSMISSION AND SET DONE FLAG
LET R1 := XMTADD - #6
MOV XMTADD,R1
SUB #6,R1
LET XDONE := XDONE + #1
INC XDONE
ABORTWAIT
EMT CSABRT
LET XMTCSR(R1) := XMTCSR(R1) CLR.BY #TXINTE!SENDFD
BIC #TXINTE!SENDFD,X
ENDIF
50012$:
ENDIF
50010$:
POP R1
MOV (SP)+,R1
...OTHERWISE, SEND A SYNC AND DECREMENT THE SYNC COUNT
ELSE
LET @XMTADD := @XMTADD SET.BY #SYNC!TSOM
50006$: BR 50013$
LET SYNCNT := SYNCNT - #1
BIS #SYNC!TSOM,@XMTA
DEC SYNCNT
ENDIF
50013$:
;DO A SUPERVISOR-TYPE RII
ENDSRV
L10027: RTI
;////////////////////////////////////
; RECEPTION INTERRUPT ROUTINE
;////////////////////////////////////
BGNSRV RINT ;[3.2.1.1.1, 3.2.20.2.1 AND 3.2.20.9.1]
;SET PROCESSOR PRIORITY
SETPRI DEVPRI
MOV DEVPRI,R0
EMT CSAPRI
MOV R1,-(SP)

```

```

2901
2902      ; IF NEW MESSAGE FLAG IS RESET...
2903
2904
2905      ; GET THE BYTE
2906
2907      LET R1 := @RCVADD
2908      016012 017701 177536      MOV      @RCVADD,R1
2909
2910      ; IF ERROR, REPORT IT AND EXIT IMMEDIATELY (FORCE RESTART)
2911
2912      ; IF R1 LT #0 THEN
2913      016016 005701
2914      016020 002017      TST      R1
2915      016022      BGE      50014$
2916      016022 010167 177060      LET RDONE := R1
2917      MOV      R1,RDONE
2918
2919      ; RESTART THE LINE
2920
2921      LET R1 := RCVADD - #2
2922      016026 016701 177522      MOV      RCVADD,R1
2923      016032 162701 000002      SUB      #2,R1
2924      016036 042711 000020      LET (R1) := (R1) CLR.BY #RCENBL
2925      016042      BIC      #RCENBL,(R1)
2926      016042 052711 000120      LET (R1) := (R1) SET.BY #RXINTE
2927      016046      BIS      #RXINTE,(R1)
2928      016046 005067 177522      LET NMSG := #0
2929      016052      CLR      NMSG
2930      016052 012601      POP R1
2931      016054 000167 000544      MOV      (SP)+,R1
2932      016060      JMP REXIT
2933      016060      ENDIF
2934      016060      50014$:
2935      016060 042701 177400      LET R1 := R1 CLR.BY #HYBYTE
2936      016064      BIC      #HYBYTE,R1
2937      016064 005767 177504      IF NMSG EQ #0 THEN
2938      016070 001142      TST      NMSG
2939      BNE      50015$
2940
2941      ; CHECK FOR SYNC/FILL CHARACTER
2942      ; (LOOKING FOR FIRST NON-SYNC/NON-FILL)
2943
2944      ; IFB R1 NE #SYNC ANDB R1 NE FIL THEN
2945      016072 120127 000226      CMPB    R1,#SYNC
2946      016076 001536      BEQ    50016$
2947      016100 120167 177462      CMPB    R1,FIL
2948      016104 001533      BEQ    50016$
2949
2950      ; NOT FILL OR SYNC. CHECK FOR ENQ (DDCMP CONTROL MESSAGE).
2951
2952      ; IFB R1 EQ #ENQ THEN
2953      016106 120127 000005      CMPB    R1,#ENQ
2954      016112 001053      BNE    50017$
2955
2956      ; IF IT IS AN ENQ, SET EXPECTED BYTE COUNT TO 7, SET MESSAGE SIZE
2957      ; TO 10 (OCTAL) AND UPDATE THE BUFFER QUEUES

```



```

2957
2958 016114 ;
2959 016114 017767 177504 177454 LET RTEMP1 := @RBQUE MOV @RBQUE,RTEMP1
2960 016122 LET @RQUE := RTEMP1 MOV RTEMP1,@RQUE
2961 016122 016777 177450 177456 LET @RTEMP1 :B- R1 MOV R1,@RTEMP1
2962 016130 LET RTEMP1 := RTEMP1 + #1 MOV R1,@RTEMP1
2963 016130 110177 177442 LET RTEMP1 := RTEMP1 + #1 INC RTEMP1
2964 016134 LET RBQUE := RBQUE + #2 ADD #2,RBQUE
2965 016134 005267 177436 IF RBQUE GT #RBQUEB THEN
2966 016140 IF RBQUE GT #RBQUEB THEN
2967 016140 062767 000002 177456 IF RBQUE GT #RBQUEB THEN
2968 016146 IF RBQUE GT #RBQUEB THEN
2969 016146 026727 177452 015632 IF RBQUE GT #RBQUEB THEN
2970 016154 003403 IF RBQUE GT #RBQUEB THEN
2971 016156 LET RBQUE := #RBQUET
2972 016156 012767 015626 177440 MOV #RBQUET,RBQUE
2973 016164 ENDF
2974 016164 50020$:
2975 016164 LET RTEMP2 := #7 MOV #7,RTEMP2
2976 016164 012767 000007 177406 LET RQUE := RQUE + #2 ADD #2,RQUE
2977 016172 LET @RQUE := #10 MOV #10,@RQUE
2978 016172 062767 000002 177406 LET RQUE := RQUE + #2 ADD #2,RQUE
2979 016200 LET @RQUE := #10 MOV #10,@RQUE
2980 016200 012777 000010 177400 LET RQUE := RQUE + #2 ADD #2,RQUE
2981 016206 LET RQUE := RQUE + #2 ADD #2,RQUE
2982 016206 062767 000002 177372 IF RQUE GT #RQUEB THEN
2983 016214 IF RQUE GT #RQUEB THEN
2984 016214 026727 177366 015622 IF RQUE GT #RQUEB THEN
2985 016222 003403 IF RQUE GT #RQUEB THEN
2986 016224 LET RQUE := #RQUET
2987 016224 012767 015610 177354 MOV #RQUET,RQUE
2988 016232 ENDF
2989 016232 50021$:
2990
2991 : SET COUNTER FLAG TO -1 (MESSAGE SIZE COUNT COMPLETE)
2992 :
2993 : LET CNTFLG := #-1
2994 016232 012767 177777 177342 MOV #-1,CNTFLG
2995 016240 ELSE
2996 016240 000453 BR 50022$
2997 016242 50017$:
2998
2999 : NOT FILL, SYNC OR ENQ. CHECK FOR SOM OR DLE (DDCMP DATA MESSAGE)
3000 :
3001 : IFB R1 EQ #SOM ORB R1 EQ #DLE THEN
3002 016242 120127 ' 000201 CMPB R1,#SOM
3003 016246 001403 BEQ 50023$
3004 016250 120127 000220 CMPB R1,#DLE
3005 016254 001033 BNE 50024$
3006 016256 50023$:
3007
3008 : MOVE QUEUE POINTER TO SIZE ENTRY AND SET COUNTER FLAG TO 1 (EXPECT
3009 : FIRST COUNT BYTE)
3010 :
3011 : LET RTEMP1 := @RBQUE
3012 016256 017767 177342 177312 MOV @RBQUE,RTEMP1

```

```

3013 016264          LET @RQUE := RTEMP1
3014 016264 016777 177306 177314          MOV      PTEMP1,@RQUE
3015 016272          LET @RTEMP1 :=B= R1
3016 016272 110177 177300          MOV      R1,@RTEMP1
3017 016276          LET RTEMP1 := RTEMP1 + #1
3018 016276 005267 177274          INC      RTEMP1
3019 016302          LET RBQUE := RBQUE + #2
3020 016302 062767 000002 177314          ADD      #2,RBQUE
3021 016310          IF RBQUE GT #RBQUEB THEN
3022 016310 026727 177310 015632          CMP      RBQUE,#RBQUEB
3023 016316 003403          BLE      50025$
3024 016320          LET RBQUE := #RBQUEB
3025 016320 012767 015626 177276          MOV      #RBQUEB,RBQUE
3026 016326          ENDIF
3027 016326          50025$:
3028 016326          LET RQUE := RQUE + #2
3029 016326 062767 000002 177252          ADD      #2,RQUE
3030 016334          LET CNTFLG := #1
3031 016334 012767 000001 177240          MOV      #1,CNTFLG
3032 016342          ELSE
3033 016342 000412          BR      50026$
3034 016344          50024$:
3035
3036          :
3037          :   BYTE WASN'T ANY OF THE LEGAL CHARACTERS... FORCE
3038          :   A RESTART BY FORCING THE NEW MESSAGE FLAG TO RESET ON EXIT.
3039          :
3039 016344          LET NMSG := NMSG - #1
3040 016344 005367 177224          DEC      NMSG
3041
3042          :   FORCE THE DUP TO RESYNC
3043          :
3044 016350          LET R1 := RCVADD - #2
3045 016350 016701 177200          MOV      RCVADD,R1
3046 016354 162701 000002          SUB      #2,R1
3047 016360          LET (R1) := (R1) CLR.BY #RCENBL
3048 016360 042711 000020          BIC      #RCENBL,(R1)
3049 016364          LET (R1) := (R1) SET.BY #RXINTE
3050 016364 052711 000120          BIS      #RXINTE,(R1)
3051 016370          ENDIF
3052 016370          50026$:
3053 016370          ENDIF
3054 016370          50022$:
3055
3056          :
3057          :   SET THE NEW MESSAGE FLAG
3058          :   EXIT ROUTINE
3059          :
3059 016370          LET NMSG := NMSG + #1
3060 016370 005267 177200          INC      NMSG
3061 016374          ENDIF
3062 016374          50016$:
3063
3064          :   NEW MESSAGE FLAG IS SET... CONTINUE BUILDING MESSAGE
3065          :
3066          :   ELSE
3067 016374 000512          BR      50027$
3068 016376          50015$:

```

```
3069
3070      :NOT A NEW MESSAGE...GET BYTE AND STORE IT
3071
3072 016376      LET @RTEMP1 :B= R1
3073 016376 110177 177174      MOVB R1,@RTEMP1
3074
3075      :
3076      :IF COUNTER FLAG IS GREATER THAN ZERO, BYTE IS ALSO THE FIRST COUNT BYTE
3077      :
3078 016402      IF CNTFLG GT #0 THEN
3079 016406 005767 177174      TST CNTFLG
3080      :
3081      :
3082      :DECREMENT THE COUNTER FLAG (SET TO GET SECOND COUNT BYTE) AND
3083      :GO DO AN RTI
3084      :
3085 016410      LET RTEMP2 :B- @RTEMP1
3086 016410 117767 177162 177162      MOVB @RTEMP1,RTEMP2
3087 016416      LET RTEMP2 := RTEMP2 CLR.BY #HYBYTE
3088 016416 042767 177400 177154      BIC #HYBYTE,RTEMP2
3089 016424      LET RTEMP1 := RTEMP1 + #1
3090 016424 005267 177146      INC RTEMP1
3091 016430      LET CNTFLG := CNTFLG - #1
3092 016430 005367 177146      DEC CNTFLG
3093 016434      ELSE
3094 016434 000472      BR 50030$
3095 016436      :
3096      :
3097      :COUNTER FLAG NOT GREATER THAN ZERO, IS IT ZERO?
3098      :
3099 016436      IF CNTFLG EQ #0 THEN
3100 016436 005767 177140      TST CNTFLG
3101 016442 001041      BNE 50032$
3102      :
3103      :IF ZERO, BYTE IS ALSO SECOND BYTE OF COUNT (6 HIGH ORDER BITS)
3104      :GET IT, MASK IT AND PUT IT INTO RTEMP2 (EXPECTED COUNT)
3105      :
3106 016444      PUSH R1
3107 016444 010146      MOV R1,-(SP)
3108 016446      LET R1 :B= @RTEMP1
3109 016446 117701 177124      MOVB @RTEMP1,R1
3110 016452      LET RTEMP1 := RTEMP1 + #1
3111 016452 005267 177120      INC RTEMP1
3112 016456      LET R1 := SWAP R1
3113 016456 000301      SWAB R1
3114 016460      LET R1 := R1 CLR.BY #140377
3115 016460 042701 140377      BIC #140377,R1
3116 016464      LET RTEMP2 := RTEMP2 + R1
3117 016464 060167 177110      ADD R1,RTEMP2
3118 016470      POP R1
3119 016470 012601      MOV (SP)+,R1
3120
3121      :
3122      :PUT THE COMPLETE SIZE IN THE QUEUE (EXPECTED COUNT +
3123      :12 (OCTAL) BYTES OF HEADER AND (RC)
3124 016472      LET @RQUE := RTEMP2
```

3125	016472	016777	177102	177106		MOV	RTEMP2,@RQUE
3126	016500				LET @RQUE := @RQUE + #12		
3127	016500	062777	000612	177100		ADD	#12,@RQUE
3128							
3129					ADJUST EXPECTED COUNT TO INCLUDE THE REMAINDER OF THE HEADER AND THE DATA CRC		
3130							
3131							
3132	016506				LET RTEMP2 := RTEMP2 + #7		
3133	016506	062767	000007	177064		ADD	#7,RTEMP2
3134							
3135					UPDATE THE QUEUE AND DECREMENT THE COUNTER FLAG (COUNT COMPLETE)		
3136							
3137	016514				LET RQUE := RQUE + #2		
3138	016514	062767	000002	177064		ADD	#2,RQUE
3139	016522				IF RQUE GT #RQUEB THEN		
3140	016522	026727	177060	015622		CMP	RQUE,#RQUEB
3141	016530	003403				BLE	50033\$
3142	016532				LET RQUE := #RQUET		
3143	016532	012767	015610	177046		MOV	#RQUET,RQUE
3144	016540				ENDIF		
3145	016540						50033\$:
3146	016540				LET CNTFLG := CNTFLG - #1		
3147	016540	005367	177036			DEC	CNTFLG
3148							
3149					GO DO AN RTI		
3150							
3151	016544				ELSE		
3152	016544	000426				BR	50034\$
3153	016546						50032\$:
3154							
3155					COUNTER FLAG IS LESS THAN ZERO... THIS IS A DATA BYTE		
3156					UPDATE POINTER AND COUNT		
3157							
3158	016546				LET RTEMP1 := RTEMP1 + #1		
3159	016546	005267	177024			INC	RTEMP1
3160	016552				LET RTEMP2 := RTEMP2 - #1		
3161	016552	005367	177022			DEC	RTEMP2
3162							
3163					IF COUNT HAS REACHED ZERO, END OF MESSAGE. SET DONE FLAG.		
3164	016556				IF RTEMP2 EQ #0 THEN		
3165	016556	005767	177016			TST	RTEMP2
3166	016562	001017				BNE	50035\$
3167	016564				LET RDONE := RDONE + #1		
3168	016564	005267	176316			INC	RDONE
3169							
3170					FORCE DUP TO RESTART RECEPTION		
3171							
3172	016570				LET R1 := RCVADD - #2		
3173	016570	016701	176760			MOV	RCVADD,R1
3174	016574	162701	000002			SUB	#2,R1
3175	016600				LET RCVCSR(R1) := RCVCSR(R1) CLR.BY #RCENBL		
3176	016600	042761	000020	000000		BIC	#RCENBL,RCVCSR(R
3177	016606				LET RCVCSR(R1) := RCVCSR(R1) SET.BY #PCENBL		
3178	016606	052761	000020	000000		BIS	#PCENBL,RCVCSR(R
3179							
3180					ABORT ANY WAITS AND RESET NEW MESSAGE FLAG		

```

3181
3182 016614          :
3183 016614 104021   :          ABORTWAIT
3184 016616          :          LET NMSG := #0
3185 016616 005067 176752 :          EMT      CSABRT
3186 016622          :          CLR      NMSG
3187 016622          :          ENDIF
3188 016622          :          ENDIF
3189 016622          :          50035$:
3190 016622          :          ENDIF
3191 016622          :          50034$:
3192 016622          :          ENDIF
3193 016622          :          50031$:
3194          :          50027$:
3195          :          :DO RTI
3196          :
3197 016622          :          POP R1
3198 016622 012601   :          MOV      (SP)+,R1
3199 016624          :          REXIT: ENDSRV
3200 016624          :          L10030:
3201 016624 000002   :          RTI
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217 016626          :          *****
3218 016626          :          ROUTINE RCVE          ;[3.2.1.1, 3.2.20.2 AND 3.2.20.8]
3219          :          RCVE:
3220          :          :RESET TIMER
3221          :
3222 016626          :          LET TIMER := #0
3223 016626 005067 176724 :          CLR      TIMER
3224          :
3225          :          :GO TO SUPERVISOR FOR AWHILE AND INCREMENT TIMER
3226          :
3227 016632          :          REPEAT
3228 016632          :          WAITUS #100          ;[3.2.1.1.4, 3.2.20.2.4, 3.2.20.8.4]
3229 016632          :          50002$:
3230 016632 012700 000100 :          MOV      #100,R0
3231 016636 104027   :          EMT      CSWTU
3232 016640          :          LET TIMER := TIMER + #1
3233 016640 005267 176712 :          INC      TIMER
3234          :
3235          :          :LOOP UNTIL TIME LIMIT EXCEEDED OR RECEIVER DONE FLAG IS SET
3236          :

```

```

3237 016644 UNTIL TIMER GT TIMLIM OR RDONE NE #0
3238 016644 026767 176706 163246 CMP TIMER,TIMLIM
3239 016652 003003 BGT 50003$
3240 016654 005767 176226 TST RDONE
3241 016660 001764 BEQ 50002$
3242 016662 50003$:
3243
3244 ;RESET THE NEW MESSAGE FLAG
3245
3246 016662 LET NMSG := #0
3247 016662 005067 176706 CLR NMSG
3248
3249 ;IF THE TIME LIMIT WASN'T EXCEEDED...
3250
3251 016666 IF TIMER LT TIMLIM THEN
3252 016666 026767 176664 163224 CMP TIMER,TIMLIM
3253 016674 002044 BGE 50004$
3254
3255 ;RESET THE TIMER
3256 ;CHECK FOR ERROR (RDONE LESS THAN ZERO)
3257
3258 016676 LET TIMER := #0
3259 016676 005067 176654 CLR TIMER
3260 016702 IF RDONE GT #0 THEN
3261 016702 005767 176200 TST RDONE
3262 016706 003426 BLE 50005$
3263
3264 ; NOT ERROR, GET BUFFER ADDRESS AND SIZE FROM QUEUE AND UPDATE QUEUE
3265
3266 016710 LET INBUF := @RQUEUE
3267 016710 017767 176670 163240 MOV @RQUEUE,INBUF
3268 016716 LET RQUEUE := RQUEUE + #2
3269 016716 062767 000002 176660 ADD #2,RQUEUE
3270 016724 LET INSIZ := @RQUEUE
3271 016724 017767 176654 163226 MOV @RQUEUE,INSIZ
3272 016732 LET RQUEUE := RQUEUE + #2
3273 016732 062767 000002 176644 ADD #2,RQUEUE
3274 016740 IF RQUEUE GT #RQUEB THEN
3275 016740 026727 176640 015622 CMP RQUEUE,#RQUEB
3276 016746 003403 BLE 50006$
3277 016750 LET RQUEUE := #RQUET
3278 016750 012767 015610 176626 MOV #RQUET,RQUEUE
3279 016756 ENDIF
3280 016756 50006$:
3281
3282 ;RESET DONE FLAG AND LEAVE
3283
3284 016756 LET RDONE := #0
3285 016756 005067 176124 CLR RDONE
3286 016762 ELSE
3287 016762 000410 BR 50007$
3288 016764 50005$:
3289
3290 ;ERROR DID OCCUR...RESET THE DONE FLAG, REPORT ERROR, RETURN WITH ERROR
3291
3292 016764 ERRHRD 26,MSG26,AREA26
  
```

3293	016764	104463				TRAP	T\$ERCODE
3294	016766	000032				.WORD	26
3295	016770	003170				.WORD	MSG26
3296	016772	004434				.WORD	AREA26
3297	016774				LET RDONE := #0		
3298	016774	005067	176106		RETURN ERROR	CLR	RDONE
3299	017000						
3300	017000	000261				SEC	
3301	017002	000440				BR	50001\$
3302	017004				ENDIF		
3303	017004						50007\$:
3304	017004				ELSE		
3305	017004	000436				BR	50010\$
3306	017006						50004\$:
3307							
3308					: TIME LIMIT EXCEEDED...RESET TIMER, GET REGISTER CONTENTS		
3309							
3310	017006				LET TIMER := #0		
3311	017006	005067	176544			CLR	TIMER
3312	017012				PUSH R1		
3313	017012	010146				MOV	R1, -(SP)
3314	017014				LET R1 := RCVADD - #2		
3315	017014	016701	176534			MOV	RCVADD, R1
3316	017020	162701	000002			SUB	#2, R1
3317	017024				PUSH R2		
3318	017024	010246				MOV	R2, -(SP)
3319	017026				LET R2 := #DUPREG		
3320	017026	012702	002234			MOV	#DUPREG, R2
3321	017032				LET RCVCSR(R2) := RCVCSR(R1)		
3322	017032	016162	000000	000000		MOV	RCVCSR(R1), RCVCS
3323	017040				LET RCVBUF(R2) := RCVBUF(R1)		
3324	017040	016162	000002	000002		MOV	RCVBUF(R1), RCVBU
3325	017046				LET XMTCSR(R2) := XMTCSR(R1)		
3326	017046	016162	000004	000004		MOV	XMTCSR(R1), XMTCS
3327	017054				LET XMTBUF(R2) := XMTBUF(R1)		
3328	017054	016162	000006	000006		MOV	XMTBUF(R1), XMTBU
3329	017062				POP R2		
3330	017062	012602				MOV	(SP)+, R2
3331	017064				POP R1		
3332	017064	012601				MOV	(SP)+, R1
3333							
3334					: PRINT ERROR MESSAGE		
3335							
3336	017066				ERRHRD 31, MSG31, AREA31		
3337	017066	104463				TRAP	T\$ERCODE
3338	017070	000037				.WORD	31
3339	017072	003261				.WORD	MSG31
3340	017074	004514				.WORD	AREA31
3341							
3342					: RETURN WITH ERROR IN ANY CASE		
3343							
3344	017076				RETURN ERROR		
3345	017076	000261				SEC	
3346	017100	000401				BR	50001\$
3347	017102				ENDIF		
3348	017102						50010\$:

```
3349 017102          ENDRTN
3350 017102
3351 017102 000241          50000$:      C.C
3352 017104          50001$:      RTS      PC
3353 017104 000207
3354
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364 017106          *****
3365 017106          DUMMY ROUTINE FOR COMPATIBILITY
3366 017106          *****
3367 017106          ROUTINE INSERR <A>
3368 017106          ENDRTN          INSERR:
3369 017106 000207          50000$:
3370          50001$:      RTS      PC
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387 017110          *****
3388 017110          THIS ROUTINE COMPUTES CRC-16 AND REQUIRES THREE ARGUMENTS:
3389 017110          DATA ADDRESS, DATA SIZE (IN BYTES) AND A WORD LOCATION
3390 017110          FOR THE COMPUTED CRC
3391 017110          [3.2.1.2.1, 3.2.1.2.4, 3.2.20.1.2, 3.2.20.3.1,
3392 017110          3.2.20.6.2, 3.2.20.6.4 AND 3.2.20.10.1]
3393 017112 010446          *****
3394 017114 162706 000004          THIS IS A STANDARD ROUTINE FROM DEC'S NETWORK GROUP
3395 017120          *****
3396 017120 010146          ROUTINE CRC <A,B,C>
3397 017122          DECLARE LOCALS          CRC:
3398 017122 010246          LOCAL<D,E>
3399 017124          ALLOCATE
3400 017124 010346          MOV      R4,-(SP)
3401 017126          LET R1 := A(R5)          MOV      SP,R4
3402 017126 016501 000000          SUB      #4,SP
3403 017132          PUSH R1          MOV      R1,-(SP)
3404 017132 016502 000002          PUSH R2          MOV      R2,-(SP)
          PUSH R3          MOV      R3,-(SP)
          LET R1 := A(R5)          MOV      A(R5),R1
          LET R2 := B(R5)          MOV      B(R5),R2
```



```

3405 017136
3406 017136 005003
3407 017140
3408 017140
3409 017140 005702
3410 017142 003500
3411 017144
3412 017144 112164 177776
3413 017150
3414 017150 042764 177400 177776
3415 017156
3416 017156 005302
3417 017160
3418 017160 010346
3419 017162 046416 177776
3420 017166 040364 177776
3421 017172 052664 177776
3422 017176
3423 017176 016464 177776 177774
3424 017204
3425 017204 042764 177760 177776
3426 017212
3427 017212 006364 177776
3428 017216
3429 017216 062764 017364 177776
3430 017224
3431 017224 017464 177776 177776
3432 017232
3433 017232 042764 177417 177774
3434 017240
3435 017240 006264 177774
3436 017244 006264 177774
3437 017250 006264 177774
3438 017254
3439 017254 062764 017364 177774
3440 017262
3441 017262 062764 000040 177774
3442 017270
3443 017270 017464 177774 177774
3444 017276
3445 017276 016446 177774
3446 017302 046416 177776
3447 017306 046464 177774 177776
3448 017314 052664 177776
3449 017320
3450 017320 042703 000377
3451 017324
3452 017324 000303
3453 017326
3454 017326 016446 177776
3455 017332 040316
3456 017334 046403 177776
3457 017340 052603
3458 017342
3459 017342 000676
3460 017344
    
```

```

LET R3 := #0
WHILE R2 GT #0 DO
    LET D(R4) := B(R1)+
    LET D(R4) := D(R4) CLR.BY #HYBYTE
    LET R2 := R2 - #1
    LET D(R4) := D(R4) XOR R3
    LET E(R4) := D(R4)
    LET D(R4) := D(R4) CLR.BY #177760
    LET D(R4) := D(R4) SHIFT #1
    LET D(R4) := D(R4) + #CRCTAB
    LET D(R4) := @D(R4)
    LET E(R4) := E(R4) CLR.BY #177417
    LET E(R4) := E(R4) SHIF1 -3
    LET E(R4) := E(R4) + #CRCTAB
    LET E(R4) := E(R4) + #32.
    LET E(R4) := @E(R4)
    LET D(R4) := D(R4) XOR E(R4)
    LET R3 := R3 CLR.BY #LOBYTE
    LET R3 := SWAP R3
    LET R3 := R3 XOR D(R4)
ENDDO
    
```

```

CLR R3
50002$:
TST R2
BLE 50003$
MOVB (R1)+,D(R4)
BIC #HYBYTE,D(R4)
DEC R2
MOV R3,-(SP)
BIC D(R4),(SP)
BIC R3,D(R4)
BIS (SP)+,D(R4)
MOV D(R4),E(R4)
BIC #177760,D(R4)
ASL D(R4)
ADD #CRCTAB,D(R4)
MOV @D(R4),D(R4)
BIC #177417,E(R4)
ASR E(R4)
ASR E(R4)
ASR E(R4)
ADD #CRCTAB,E(R4)
ADD #32.,E(R4)
MOV @E(R4),E(R4)
MOV E(R4),-(SP)
BIC D(R4),(SP)
BIC E(R4),D(R4)
BIS (SP)+,D(R4)
BIC #LOBYTE,R3
SWAB R3
MOV D(R4),-(SP)
BIC R3,(SP)
BIC D(R4),R3
BIS (SP)+,R3
BR 50002$
50003$:
    
```

```

3461 017344          LET C(R5) := R3
3462 017344 010365 000004
3463 017350          POP R3
3464 017350 012603          POP R2
3465 017352          POP R1
3466 017352 012602          ENDRTN
3467 017354
3468 017354 012601
3469 017356
3470 017356

```

```

MOV R3,C(R5)
MOV (SP)+,R3
MOV (SP)+,R2
MOV (SP)+,R1

```

50000\$:

50001\$:

```

MOV R4,SP
MOV (SP)+,R4
RTS PC

```

```

3471 017356
3472 017356 010406
3473 017360 012604
3474 017362 000207
3475
3476 017364 000000      CRCTAB: 000000
3477 017366 140301      140301
3478 017370 140601      140601
3479 017372 000500      000500
3480 017374 141401      141401
3481 017376 001700      001700
3482 017400 001200      001200
3483 017402 141101      141101
3484 017404 143001      143001
3485 017406 003300      003300
3486 017410 003600      003600
3487 017412 143501      143501
3488 017414 002400      002400
3489 017416 142701      142701
3490 017420 142201      142201
3491 017422 002100      002100
3492 017424 000000      000000
3493 017426 146001      146001
3494 017430 154001      154001
3495 017432 012000      012000
3496 017434 170001      170001
3497 017436 036000      036000
3498 017440 024000      024000
3499 017442 162001      162001
3500 017444 120001      120001
3501 017446 066000      066000
3502 017450 074000      074000
3503 017452 132001      132001
3504 017454 050000      050000
3505 017456 116001      116001
3506 017460 104001      104001
3507 017462 042000      042000
3508 017464          ENDMOD

```

```

3509 017464 BGNMOD TESTS
3510
3511 :   HARDWARE TESTS [3.21]
3512 :
3513 :
3514 : <><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
3515 :
3516 :   SELF TEST [3.2C]
3517 :
3518 : <><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
3519 017464 BGNTEST
3520
3521 : FORCE INIT OF COMM DEVICE AND LINK
3522 :
3523 017464 LET INIT := #0
3524 017464 005067 176074 CLR INIT
3525
3526 : GET MESSAGE ADDRESS AND SIZE
3527 :
3528 017470 LET OUTMSG := MDMSG CLR.BY #100001
3529 017470 016767 000770 162444 MOV MDMSG,OUTMSG
3530 017476 042767 100001 162436 BIC #100001,OUTMSG
3531 017504 LET OUTMSG+2 := MDMSG+2
3532 017504 016767 000756 162432 MOV MDMSG+2,OUTMSG+2
3533
3534 : SEND THE MESSAGE
3535 :
3536 017512 CALL TEROUT
3537 017512 004767 172772 JSR PC,TEROUT
3538
3539 : IF ERROR....
3540 :
3541 017516 IF.ERROR THEN
3542 017516 103015 BCC 50004$
3543
3544 : INSERT ERROR INTO STATISTICS TABLE (MANUFACTURING VERSIONS ONLY),
3545 : REPORT THE ERROR, REQUEST SUPERVISOR TO LOOP
3546 : IF LOOPING ENABLED (EXIT TEST IF NO LOOPING)
3547 :
3548 017520 CALL INSERR IN <#0>
3549 017520 010546 MOV R5,-(SP)
3550 017522 012745 000000 MOV #0,-(R5)
3551 017526 004767 177354 JSR PC,INSERR
3552 017532 012605 MOV (SP)+,R5
3553 017534 ERRHRD 50,MSG50,AREA40
3554 017534 104463 TRAP T$ERCODE
3555 017536 000062 .WORD 50
3556 017540 003447 .WORD MSG50
3557 017542 004242 .WORD AREA40
3558 017544 CKLOOP
3559 017544 104006 EMT C$CLP1
3560 017546 EXIT TST
3561 017546 104032 EMT C$EXIT
3562 017550 000412 .WORD L10031-
3563 017552
3564 017552 ENDEF 50004$:
```

```

3565
3566      :LOOK FOR ACK
3567      :
3568      :      CALL TERIN OUT <MCOMP>
3569      :      017552      162705      000002
3570      :      017556      004767      174010
3571      :      017562      012567      000674
3572
3573      :      :IF ERROR...INSERT ERROR INTO TABLE (MANUFACTURING TEST ONLY)
3574      :      :      REPORT THE ERROR, REQUEST LOOP IF ENABLED (EXIT TEST
3575      :      :      IF NO LOOPING)
3576      :
3577      :      IF.ERROR THEN
3578      :      017566      103015
3579      :      017570
3580      :      017570      010546
3581      :      017572      012745      000000
3582      :      017576      004767      177304
3583      :      017602      012605
3584      :      017604
3585      :      017604      104463
3586      :      017606      000062
3587      :      017610      003447
3588      :      017612      004242
3589      :      017614
3590      :      017614      104006
3591      :      017616
3592      :      017616      104032
3593      :      017620      000342
3594      :      017622
3595      :      017622
3596
3597      :      :IF MESSAGE WAS NOT AN ACK...ERROR
3598      :      :      REPORT ERROR, LOOP IF ENABLED, EXIT TEST IF NOT
3599      :      :
3600      :      IF MCOMP NE #0 THEN
3601      :      017622      005767      000634
3602      :      017626      001407
3603      :      017630
3604      :      017630      104463
3605      :      017632      000062
3606      :      017634      003447
3607      :      017636      004242
3608      :      017640
3609      :      017640      104006
3610      :      017642
3611      :      017642      104032
3612      :      017644      000316
3613      :      017646
3614      :      017646
3615
3616      :      :FORCE A RESTART ON THE LINK AND WAIT FOR SELFTEST TO COMPLETE
3617      :      :
3618      :      LET INIT := #0
3619      :      017646      005067      175712
3620      :      017652
3620

```

SUB #1\*2,R5  
JSR PC,TERIN  
MOV (R5)+,MCOMP

BCC 50005\$

MOV R5,-(SP)  
MOV #0,-(R5)  
JSR PC,INSERR  
MOV (SP)+,R5

TRAP T\$ERCODE  
.WORD 50  
.WORD MSG50  
.WORD AREA40

EMT C\$CLP1

EMT C\$EXIT  
.WORD L10031-

50005\$:

TST MCOMP  
BEQ 50006\$

TRAP T\$ERCODE  
.WORD 50  
.WORD MSG50  
.WORD AREA40

EMT C\$CLP1

EMT C\$EXIT  
.WORD L10031-

50006\$:

CLR INIT

3621	017652	005067	162234			CLR	SCRTCM
3622	017656			REPEAT			
3623	017656				500078:		
3624	017656			WAITUS #100			
3625	017656	012700	000100			MOV	#100,R0
3626	017662	104027				EMT	C\$WTU
3627	017664			LET SCRTCM := SCRTCM + #1			
3628	017664	005267	162222			INC	SCRTCM
3629	017670			UNTIL SCRTCM GT STWAIT			
3630	017670	026767	162216	171776		CMP	SCRTCM,STWAIT
3631	017676	003767				BLE	500078
3632							
3633				SET UP MESSAGE ASKING FOR RESULTS AND SEND IT			
3634							
3635	017700			LET OUTMSG := MDMSG1 CLR.BY #100001			
3636	017700	016767	000570	162234		MOV	MDMSG1,OUTMSG
3637	017706	042767	100001	162226		BIC	#100001,OUTMSG
3638	017714			LET OUTMSG+2 := MDMSG1+2			
3639	017714	016767	000556	162222		MOV	MDMSG1+2,OUTMSG+
3640	017722			CALL TEROUT			
3641	017722	004767	172562			JSR	PC,TEROUT
3642							
3643				IF ERROR....INSERT ERROR IN TABLE (MANUF. TESTS ONLY), REPORT ERROR,			
3644				LOOP IF ENABLED, EXIT TEST IF NO LOOPING			
3645							
3646	017726			IF.ERROR THEN			
3647	017726	103015				BCC	500108
3648	017730			CALL INSERR IN <#0>			
3649	017730	010546				MOV	R5,-(SP)
3650	017732	012745	000000			MOV	#0,-(R5)
3651	017736	004767	177144			JSR	PC,INSERR
3652	017742	012605				MOV	(SP)+,R5
3653	017744			ERRHRD 50,MSG50,AREA40			
3654	017744	104463				TRAP	T\$ERCODE
3655	017746	000062				.WORD	50
3656	017750	003447				.WORD	MSG50
3657	017752	004242				.WORD	AREA40
3658	017754			CKLOOP			
3659	017754	104006				EMT	C\$CLP1
3660	017756			EXIT TST			
3661	017756	104032				EMT	C\$EXIT
3662	017760	000202				.WORD	L10031-
3663	017762			ENDIF			
3664	017762				500108:		
3665							
3666				ACK TERMINAL AND GET ITS RESPONSE			
3667							
3668	017762			REPEAT			
3669	017762				500118:		
3670							
3671				GET THE RESPONSE			
3672							
3673	017762			CALL TERIN OUT <MCOMP>			
3674	017762	162705	000002			SUB	#1+2,R5
3675	017766	004767	173600			JSR	PC,TERIN
3676	017772	012567	000464			MOV	(R5)+,MCOMP

```

3677
3678
3679
3680 017776
3681 017776 103015
3682 020000
3683 020000 010546
3684 020002 012745 000000
3685 020006 004767 177074
3686 020012 012605
3687 020014
3688 020014 104463
3689 020016 000062
3690 020020 003447
3691 020022 004242
3692 020024
3693 020024 104006
3694 020026
3695 020026 104032
3696 020030 000132
3697 020032
3698 020032
3699 020032
3700 020032 005767 000424
3701 020036 001026
3702 020040
3703 020040 004767 000120
3704
3705
3706
3707 020044
3708 020044 103023
3709 020046
3710 020046 010546
3711 020050 012745 000000
3712 020054 004767 177026
3713 020060 012605
3714 020062
3715 020062 010546
3716 020064 012745 000001
3717 020070 004767 177012
3718 020074 012605
3719 020076
3720 020076 104463
3721 020100 000062
3722 020102 003447
3723 020104 004242
3724 020106
3725 020106 104006
3726 020110
3727 020110 104032
3728 020112 000050
3729 020114
3730 020114
3731 020114
3732 020114

```

HANDLE ERROR IF ONE OCCURED.

IF.ERROR THEN

CALL INSERR IN <#0>

ERRHRD 50,MSG50,AREA40

CKLOOP

EXIT TST

ENDIF

IF MCOMP EQ #0 THEN

CALL OUTREP

HANDLE ERROR IF ONE OCCURED WHILE ACK'ING

IF.ERROR THEN

CALL INSERR IN <#0>

CALL INSERR IN <#1>

ERRHRD 50,MSG50,AREA40

CKLOOP

EXIT TST

ENDIF

ENDIF

```

BCC 50012$
MOV R5,-(SP)
MOV #0,-(R5)
JSR PC,INSERR
MOV (SP)+,R5
TRAP T$ERCODE
.WORD 50
.WORD MSG50
.WORD AREA40
EMT C$CLP1
EMT C$EXIT
.WORD L10031-.

```

50012\$:

```

TST MCOMP
BNE 50013$
JSR PC,OUTREP

```

```

BCC 50014$
MOV R5,-(SP)
MOV #0,-(R5)
JSR PC,INSERR
MOV (SP)+,R5
MOV R5,-(SP)
MOV #1,-(R5)
JSR PC,INSERR
MOV (SP)+,R5

```

```

TRAP T$ERCODE
.WORD 50
.WORD MSG50
.WORD AREA40

```

```

EMT C$CLP1
EMT C$EXIT
.WORD L10031-.

```

50014\$:

50013\$:

```
3733
3734      : REPEAT THE LOOP UNTIL A MESSAGE IS RECEIVED
3735      :
3736 020114      : UNTIL MCOMP NE #0
3737 020114 005767 000342      :
3738 020120 001720      : TST MCOMP
3739      : BEQ 50011$
3740      :
3741      : PROCESS THE MESSAGE RECEIVED
3742      :
3743 020122      : CALL STINPT
3744      :
3745      : IF IN ERROR...INSERT ERROR INTO TABLE (MANUF. TESTS ONLY, REPORT
3746      : ERROR, LOOP IF ENABLED, EXIT TEST IF NOT
3747      :
3748 020126      : IF .ERROR THEN
3749 020126 103015      : BCC 50015$
3750 020130      : CALL INSERR IN <#1>
3751 020130 010546      : MOV R5,-(SP)
3752 020132 012745 000001      : MOV #1,-(R5)
3753 020136 004767 176744      : JSR PC,INSERR
3754 020142 012605      : MOV (SP)+,R5
3755 020144      : ERRHRD 50,MSG50,AREA40
3756 020144 104463      : TRAP T$ERCODE
3757 020146 000062      : .WORD 50
3758 020150 003447      : .WORD MSG50
3759 020152 004242      : .WORD AREA40
3760 020154      : CKLOOP
3761 020154 104006      : EMT C$CLP1
3762 020156      : EXIT TST
3763 020156 104032      : EMT C$EXIT
3764 020160 000002      : .WORD L10031-
3765 020162      :
3766 020162      : ENDFIF
3767      :
3768      :
3769      : END OF TEST
3770      :
3771 020162      : ENDTST
3772 020162 104001      :
3773      :
3774      :
3775      :
3776      :
3777      :
3778      :
3779      :
3780      : *****
3781      : ROUTINE TO SEND AND ACK IN ORDER TO POLL A TERMINAL
3782      : *****
3783 020164      : ROUTINE OUTREP
3784 020164      :
3785      :
3786      :
3787      : INSERT MESSAGE NUMBER (LAST MESSAGE SENT) AND RESPONSE NUMBER
3788      : (LAST MESSAGE RECEIVED)...INSERT STATION ADDRESS
```

```

3789 020164          LET MSGN :B= MSGNO
3790 020164 116767 162004 000316          MOVB  MSGNO,MSGN
3791 020172          LET REPN :B= RMSGNO
3792 020172 116767 162000 000307          MOVB  RMSGNO,REPN
3793 020200          LET REPA :B- STRTSA
3794 020200 116767 163451 000303          MOVB  STRTSA,REPA
3795
3796          : COMPUTE AND INSERT CRC
3797          :
3798 020206          CALL CRC IN <#REPM,#6> OUT <REPCRC>
3799 020206 162705 000002          SUB   #1+2,R5
3800 020212 010546          MOV   R5,-(SP)
3801 020214 012745 000006          MOV   #6,-(R5)
3802 020220 012745 020504          MOV   #REPM,-(R5)
3803 020224 004767 176660          JSR  PC,CRC
3804 020230 012605          MOV   (SP)+,R5
3805 020232 012567 000254          MOV   (R5)+,REPCRC
3806
3807          : SET UP BUFFER POINTER AND SIZE AND SEND THE MESSAGE
3808          :
3809 020236          LET XBUFFER :- #REPM
3810 020236 012767 020504 175324          MOV   #REPM,XBUFFER
3811 020244          LET XSIZE :- #10
3812 020244 012767 000010 175320          MOV   #10,XSIZE
3813 020252          CALL XMIT
3814 020252 004767 174316          JSR  PC,XMIT
3815
3816          : IF AN ERROR OCCURRED, RETURN WITH ERROR. OTHERWISE RETURN NORMALLY
3817          :
3818 020256          IF.ERROR THEN
3819 020256 103004          BCC   50002$
3820 020260          LET OUTERR := OUTERR + #1
3821 020260 005267 161644          INC  OUTERR
3822 020264          RETURN ERROR
3823 020264 000261          SEC
3824 020266 000401          BR   50001$
3825 020270          ENDIF
3826 020270          50002$:
3827 020270          ENDRTN
3828 020270          50700$:
3829 020270 000241          CLC
3830 020272          50001$:
3831 020272 000207          RTS  PC
3832
3833
3834
3835
3836
3837          : *****
3838          : ROUTINE TO PROCESS TERMINAL SELFTEST RESULTS
3839          : *****
3840
3841          : *****
3842 020274          ROUTINE STINPT
3843 020274          STINPT:
3844 020274          PUSH R1

```



```

3845 020274 010146                                MOV     R1,-(SP)
3846 020276                                PUSH R2
3847 020276 010246                                MOV     R2,-(SP)
3848 020300                                PUSH R3
3849 020300 010346                                MOV     R3,-(SP)
3850
3851      : SKIP RESPONSE HEADER AND CHECK FIRST DATA BYTE
3852      :
3853 020302                                LET R1 := INBUF + #4
3854 020302 016701 161650                        MOV     INBUF,R1
3855 020306 062701 000004                        ADD     #4,R1
3856
3857      : IF IT IS 'A', GOOD RESPONSE. OTHERWISE...
3858      :
3859 020312                                IFB (R1) NE #101 THEN
3860 020312 121127 000101                        CMPB   (R1),#101
3861 020316 001454                        BEQ    50C02$
3862
3863      : GET THE BYTE AND CHECK TO SEE IF IT IS 'M' (MEMORY ERROR)
3864      :
3865 020320                                LET R3 := (R1)+
3866 020320 112103                                MOVB   (R1)+,R3
3867 020322                                IFB R3 EQ #115 THEN
3868 020322 120327 000115                        CMPB   R3,#115
3869 020326 001025                        BNE    50003$
3870
3871      : MEMORY ERROR...GET THE TYPE AND BANK NUMBER
3872      : INSERT INTO ERROR MESSAGE, REPORT THE ERROR AND
3873      : RETURN WITH ERROR
3874      : (NOTE: ERROR REPORTING PROTOCOL DELIMITERS ARE
3875      : IGNORED. THIS PROGRAM WILL NOT ACCEPT INPUT FROM
3876      : ANY OTHER INTELLIGENT TERMINAL PROPERLY.)
3877      :
3878 020330                                LET R1 := R1 + #2
3879 020330 062701 000002                        ADD     #2,R1
3880 020334                                LET R2 := #MTY
3881 020334 012702 003615                        MOV     #MTY,R2
3882 020340                                LET (R2)+ := (R1)+
3883 020340 112122                                MOVB   (R1)+,(R2)+
3884 020342                                LET (R2)+ := (R1)+
3885 020342 112122                                MOVB   (R1)+,(R2)+
3886 020344                                LET R2 := #BNO
3887 020344 012702 003631                        MOV     #BNO,R2
3888 020350                                LET R1 := R1 + #1
3889 020350 005201                                INC     R1
3890 020352                                LET R3 := (R1)
3891 020352 111103                                MOVB   (R1),R3
3892 020354                                LET R3 := R3 + #60
3893 020354 062703 000060                        ADD     #60,R3
3894 020360                                LET (R2) := R3
3895 020360 110312                                MOVB   R3,(R2)
3896 020362                                ERRHRD 71,MSG71
3897 020362 104423                                TRAP   T$ERCODE
3898 020364 000107                                .WORD 71
3899 020366 003571                                .WORD MSG71
3900 020370                                POP R3

```

```

3901 020370 012603                                MOV      (SP)+,R3
3902 020372                                POP R2
3903 020372 012602                                MOV      (SP)+,R2
3904 020374                                POP R1
3905 020374 012601                                MOV      (SP)+,R1
3906 020376                                RETURN ERROR
3907 020376 000261                                SEC
3908 020400 000427                                BR       50001$
3909 020402                                ELSE
3910 020402                                50003$:
3911
3912      : NOT A MEMORY ERROR OR PASS...CHECK FOR "P" (PROCESSOR ERROR)
3913      :
3914 020402                                IFB R3 EQ #120 THEN
3915 020402 120327 000120                                CMPB    R3,#120
3916 020406 001010                                BNE     50005$
3917
3918      : PROCESSOR ERROR...REPORT IT AND EXIT WITH ERROR
3919      :
3920 020410                                ERRHRD 72,MSG72
3921 020410 104423                                TRAP    T$ERCODE
3922 020412 000110                                .WORD  72
3923 020414 003634                                .WORD  MSG72
3924 020416                                POP R3
3925 020416 012603                                MOV      (SP)+,R3
3926 020420                                POP R2
3927 020420 012602                                MOV      (SP)+,R2
3928 020422                                POP R1
3929 020422 012601                                MOV      (SP)+,R1
3930 020424                                RETURN ERROR
3931 020424 000261                                SEC
3932 020426 000414                                BR       50001$
3933 020430                                ELSE
3934 020430                                50005$:
3935
3936      : RESPONSE WAS NOT LEGAL...REPORT IT AND RETURN WITH ERROR
3937      :
3938 020430                                ERRHRD 70,MSG70
3939 020430 104423                                TRAP    T$ERCODE
3940 020432 000106                                .WORD  70
3941 020434 003536                                .WORD  MSG70
3942 020436                                POP R3
3943 020436 012603                                MOV      (SP)+,R3
3944 020440                                POP R2
3945 020440 012602                                MOV      (SP)+,R2
3946 020442                                POP R1
3947 020442 012601                                MOV      (SP)+,R1
3948 020444                                RETURN ERROR
3949 020444 000261                                SEC
3950 020446 000404                                BR       50001$
3951 020450                                ENDIF
3952 020450                                50006$.
3953 020450                                ENDIF
3954 020450                                50004$:
3955 020450                                ENDIF
3956 020450                                50002$:

```



```

4013 020530 005067 175030          CLR      INIT
4014 020534          CALL XSTRT
4015 020534 004767 172204          JSR      PC,XSTRT
4016
4017          : IF ERROR DURING START-UP...INSERT INTO TABLE (MANUF. TESTS ONLY, REPORT
4018          : IT, LOOP IF ENABLED--EXIT IF NOT
4019          :
4020 020540          IF ERROR THEN
4021 020540 103015          BCC      500078
4022 020542          CALL INSERR IN <#0>
4023 020542 010546          MOV      R5,-(SP)
4024 020544 012745 000000          MOV      #0,-(R5)
4025 020550 004767 176332          JSR      PC,INSERR
4026 020554 012605          MOV      (SP)+,R5
4027 020556          ERRHRD 51,MSG51,AREA40
4028 020556 104463          TRAP    T$ERCODE
4029 020560 000063          .WORD  51
4030 020562 003475          .WORD  MSG51
4031 020564 004242          .WORD  ARFA40
4032 020566          CKLOOP
4033 020566 104006          EMT     C$CLP1
4034 020570          EXIT TST
4035 020570 104032          EMT     C$EXIT
4036 020572 000460          .WORD  L10032-
4037 020574          ENDIF
4038 020574          500078:
4039
4040          : FILL MESSAGE AREA WITH 256 BYTES OF SELECTED PATTERN
4041          :
4042 020574          PUSH R1
4043 020574 010146          MOV      R1,-(SP)
4044 020576          LET R1 := #LOOPB
4045 020576 012701 021412          MOV      #LOOPB,R1
4046 020602          IF PATTERN NE #0 THEN
4047 020602 005767 171064          TST     PATTERN
4048 020606 001403          BEQ     500108
4049 020610          LET SCRTCH := #0
4050 020610 005067 161276          CLR     SCRTCH
4051 020614          ELSE
4052 020614 000403          BR     500118
4053 020616          500108:
4054 020616          LET SCRTCH := #125125
4055 020616 012767 125125 161266          MOV     #125125,SCRTCH
4056 020624          ENDIF
4057 020624          500118:
4058 020624          REPEAT
4059 020624          500128:
4060 020624          LET (R1)+ := SCRTCH
4061 020624 016721 161262          MOV     SCRTCH,(R1)+
4062 020630          UNTIL R1 EQ #ELOOPB
4063 020630 020127 022012          CMP     R1,#ELOOPB
4064 020634 001373          BNE     500128
4065
4066          : PUT A 1 AT THE END TO ASSURE THAT THE FINAL BYTE IS NOT NULL (0)
4067          :
4068 020636          LET R1 := #ELOOPB - #1

```

4069	020636	012701	022012				
4070	020642	005301				MOV	#LOOPB,R1
4071	020644			LET (R1) :B= #1		DEC	R1
4072	020644	112711	00000i			MOVB	#1,(R1)
4073							
4074				GET THE STATION ADDRESS AND INSERT IT IN THE HEADER			
4075							
4076	020650			LET R1 := #PTABLE + #5			
4077	020650	012701	002200			MOV	#PTABLE,R1
4078	020654	062701	000005			ADD	#5,R1
4079	020660			LET LOOSTA :B- (R1)			
4080	020660	111167	000523			MOVB	(R1),LOOSTA
4081	020664			POP R1			
4082	020664	012601				MOV	(SP)+,R1
4083							
4084				INSERT LOOP DATA TYPE			
4085							
4086	020666			LET LOOPB :B- #OSOP			
4087	020666	112767	000030 0005.16			MOVB	#OSOP,LOOPB
4088							
4089				SET MESSAGE NUMBER			
4090							
4091	020674			LET LOOPN :B- #1			
4092	020674	112767	000001 000504			MOVB	#1,LOOPN
4093							
4094				COMPUTE AND INSERT HEADER CRC			
4095							
4096	020702			CALL CRC IN <#LOOPH,#6> OUT <LCRCH>			
4097	020702	162705	000002			SUB	#1*2,R5
4098	020706	010546				MOV	R5,-(SP)
4099	020710	012745	000006			MOV	#6,-(R5)
4100	020714	012745	021402			MOV	#LOOPH,-(R5)
4101	020720	004767	176164			JSR	PC,CRC
4102	020724	012605				MOV	(SP)+,R5
4103	020726	012567	000456			MOV	(R5)+,LCRCH
4104							
4105				COMPUTE AND INSERT DATA CRC			
4106							
4107	020732			CALL CRC IN <#LOOPB,#256.> OUT <LCRCD>			
4108	020732	162705	000002			SUB	#1*2,R5
4109	020736	010546				MOV	R5,-(SP)
4110	020740	012745	000400			MOV	#256,-(R5)
4111	020744	012745	021412			MOV	#LOOPB,-(R5)
4112	020750	004767	176134			JSR	PC,CRC
4113	020754	012605				MOV	(SP)+,R5
4114	020756	012567	001030			MOV	(R5)+,LCRCD
4115							
4116				SET UP BUFFER POINTER AND SIZE....SEND MESSAGE			
4117							
4118	020762			LET XBUFFER := #LOOPH			
4119	020762	012767	021402 174600			MOV	#LOOPH,XBUFFER
4120	020770			LET XSIZE := #266.			
4121	020770	012767	000412 174574			MOV	#266.,XSIZE
4122	020776			CALL XMIT			
4123	020776	004767	173572			JSR	PC,XMIT
4124							

```

4125 ;IF ERROR OCCURRED...INSERT THE ERROR INTO TABLE (MANUF. TESTS ONLY),
4126 ;REPORT THE ERROR, LOOP IF ENABLED
4127 ;
4128 IF.ERROR THEN
4129     021002 103012
4130     021004
4131     021004 010546
4132     021006 012745 000000
4133     021012 004767 176070
4134     021016 012605
4135     021020
4136     021020 104463
4137     021022 000063
4138     021024 003475
4139     021026 004242
4140
4141     021030
4142     021030
4143     021030 104006
4144 ;
4145 ;SEND ACK'S TO TERMINAL UNTIL IT RESPONDS
4146 ;
4147     021032
4148     021032
4149 ;
4150 ;GET RESPONSE
4151 ;
4152     021032
4153     021032 162705 000002
4154     021036 004767 172530
4155     021042 012567 000326
4156 ;
4157 ;IF ERROR OCCURS...INSERT IN TABLE (MAN. ONLY), REPORT IT, LOOP IF
4158 ;ENABLED, EXIT TEST IF NOT
4159 ;
4160 IF.ERROR THEN
4161     021046 103012
4162     021050
4163     021050 010546
4164     021052 012745 000000
4165     021056 004767 176024
4166     021062 012605
4167     021064
4168     021064 104463
4169     021066 000063
4170     021070 003475
4171     021072 004242
4172
4173     021074
4174     021074
4175     021074 104006
4176     021076
4177     021076 005767 000272
4178     021102 001026
4179     021104
4180     021104 004767 177054

```

IF.ERROR THEN

CALL INSERR IN <#0>

ERRHRD 51,MSG51,AREA40

ENDIF

CKLOOP

SEND ACK'S TO TERMINAL UNTIL IT RESPONDS

REPEAT

GET RESPONSE

CALL TERIN OUT <LCOMP>

IF ERROR OCCURS...INSERT IN TABLE (MAN. ONLY), REPORT IT, LOOP IF  
ENABLED, EXIT TEST IF NOT

IF.ERROR THEN

CALL INSERR IN <#0>

ERRHRD 51,MSG51,AREA40

ENDIF

CKLOOP

IF LCOMP EQ #0 THEN

CALL OUTREP

BCC 500138

MOV R5,-(SP)  
MOV #0,-(R5)  
JSR PC,INSERR  
MOV (SP)+,R5

TRAP T\$ERCODE  
.WORD 51  
.WORD MSG51  
.WORD AREA40

500138:

EMT C\$CLP1

500148:

SUB #1\*2,R5  
JSR PC,TERIN  
MOV (R5)+,LCOMP

BCC 500158

MOV R5,-(SP)  
MOV #0,-(R5)  
JSR PC,INSERR  
MOV (SP)+,R5

TRAP T\$ERCODE  
.WORD 51  
.WORD MSG51  
.WORD AREA40

500158:

EMT C\$CLP1

IST LCOMP  
BNE 500168

JSR PC,OUTREP

```

4181
4182      : IF AN ERROR OCCURS...INSERT IN TABLE (MAN. ONLY), REPORT IT, LOOP
4183      : IF ENABLED - EXIT TEST IF NOT
4184      :
4185      IF.ERROR THEN
4186      021110 103023      CALL INSERR IN <#0>      BCC      50017$
4187      021112
4188      021112 010546      CALL INSERR IN <#2>      MOV      R5,-(SP)
4189      021114 012745 000000      MOV      #0,-(R5)
4190      021120 004767 175762      JSR      PC,INSERR
4191      021124 012605      MOV      (SP)+,R5
4192      021126
4193      021126 010546      CALL INSERR IN <#2>      MOV      R5,-(SP)
4194      021130 012745 000002      MOV      #2,-(R5)
4195      021134 004767 175746      JSR      PC,INSERR
4196      021140 012605      MOV      (SP)+,R5
4197      021142
4198      021142 104463      ERRHRD 51,MSG51,AREA40
4199      021144 000063      TRAP     T$ERCODE
4200      021146 003475      .WORD   51
4201      021150 004242      .WORD   MSG51
4202      021152
4203      021152 104006      .WORD   AREA40
4204      021154
4205      021154 104032      CKLOOP
4206      021156 000074      EXIT TST
4207      021160
4208      021160
4209      021160
4210      021160
4211
4212      : REPEAT LOOP UNTIL TERMINAL SENDS LOOPBACK DATA
4213      :
4214      021160
4215      021160 005767 000210      UNTIL LCOMP NE #0
4216      021164 001722
4217
4218      : PROCESS THE LOOPBACK DATA
4219      :
4220      021166
4221      021166 004767 000062      CALL PROLOP
4222      021172
4223      021172 104004      BGNSEG
4224
4225      : REINIT THE LINK
4226      :
4227      021174
4228      021174 005067 174364      LET INIT := #0
4229      021200
4230      021200 004767 171540      CALL XSTR1
4231
4232      : IF ERROR OCCURS, INSERT ERROR INTO TABLE (MANUF. ONLY), REPORT EPROR,
4233      : LOOP ON REINIT IF ERROR LOOPING ENABLED...EXIT TEST IF NO LOOPING
4234      :
4235      021204
4236      021204 103020      IF.ERROR THEN      BCC      50020$

```

50017\$:

50016\$:

ENDIF

: REPEAT LOOP UNTIL TERMINAL SENDS LOOPBACK DATA

UNTIL LCOMP NE #0

: PROCESS THE LOOPBACK DATA

CALL PROLOP

BGNSEG

: REINIT THE LINK

LET INIT := #0

CALL XSTR1

: IF ERROR OCCURS, INSERT ERROR INTO TABLE (MANUF. ONLY), REPORT EPROR,  
: LOOP ON REINIT IF ERROR LOOPING ENABLED...EXIT TEST IF NO LOOPING

IF.ERROR THEN

```

4237 021206                CALL INSERR IN <#0>
4238 021206 010546                MOV     R5,-(SP)
4239 021210 012745 000600        MOV     #0,-(R5)
4240 021214 004767 175666        JSR    PC,INSERR
4241 021220 012605                MOV     (SP)+,R5
4242 021222                CALL INSERR IN <#2>
4243 021222 010546                MOV     R5,-(SP)
4244 021224 012745 000002        MOV     #2,-(R5)
4245 021230 004767 175652        JSR    PC,INSERR
4246 021234 012605                MOV     (SP)+,R5
4247 021236                ERRHRD 51,MSG51,AREA61
4248 021236 104463                TRAP   T$ERCODE
4249 021240 000063                .WORD 51
4250 021242 003475                .WORD MSG51
4251 021244 005036                .WORD AREA61
4252 021246                ENDIF
4253 021246                CKLOOP
4254 021246                50020$:
4255 021246 104006                EMT    C$CLP1
4256 021250                ENDSEG
4257 021250                10000$:
4258 021250 104005                EMT    C$ESEG
4259
4260                :END OF TEST
4261                :
4262 021252                ENDTST
4263 021252                L10032:
4264 021252 104001                EMT    C$ETST
4265
4266
4267
4268
4269
4270                :-----:
4271                :
4272                : ROUTINE TO PROCESS LOOPBACK DATA
4273                :
4274                :-----:
4275 021254                ROUTINE PROLOP
4276 021254                PROLOP:
4277 021254                PUSH R1
4278 021254 010146                MOV     R1,-(SP)
4279 021256                PUSH R2
4280 021256 010246                MOV     R2,-(SP)
4281 021260                PUSH R3
4282 021260 010346                MOV     R3,-(SP)
4283
4284                :R1 = POINTER TO EXPECTED DATA
4285                :R2 = COUNT OF DATA EXPECTED
4286                :R3 = POINTER TO DATA FROM TERMINAL
4287                :
4288 021262                LET R1 := #LOOPB
4289 021262 012701 021412                MOV     #LOOPB,R1
4290 021266                LET R2 := #128.
4291 021266 012702 000200                MOV     #128.,R2
4292 021272                LET R3 := INBUF

```



```

4293 021272 016703 160660                                MOV     INBUF,R3
4294
4295                ;CHECK DATA BYTE-BY-BYTE...
4296                ;
4297 021276                REPEAT
4298 021276                                50002$:
4299
4300                ;IF ERROR...INSERT ERROR INTO TABLE (MANUF. ONLY), REPORT ERROR
4301                ;(INCLUDING RECEIVED/EXPECTED BYTES), RETURN WITH ERROR
4302                ;
4303 021276                IFB (R1) NE (R3) THEN
4304 021276 121113                                CMPB   (R1),(R3)
4305 021300 001423                                BEQ    50003$
4306 021302
4307 021302 011167 163756                LET A52A := (R1)
4308 021306                LET A52B := (R3)
4309 021306 011367 163754                MOV    (R1),A52A
4310 021312                MOV    (R3),A52B
4311 021312 010546                                MOV    R5,-(SP)
4312 021314 012745 000002                MOV    #2,-(R5)
4313 021320 004767 175562                JSR    PC,INSERR
4314 021324 012605                                MOV    (SP)+,R5
4315 021326                ERRSOF 52,MSG52,AREA52
4316 021326 104464                                TRAP  T$ERCODE
4317 021330 000064                                .WORD 52
4318 021332 003523                                .WORD MSG52
4319 021334 005166                                .WORD AREA52
4320 021336                POP R3
4321 021336 012603                                MOV    (SP)+,R3
4322 021340                POP R2
4323 021340 012602                                MOV    (SP)+,R2
4324 021342                POP R1
4325 021342 012601                                MOV    (SP)+,R1
4326 021344                RETURN ERROR
4327 021344 000261
4328 021346 000411
4329 021350                ENDIF
4330 021350                                50003$:
4331
4332                ;UPDATE COUNT AND POINTERS
4333                ;
4334 021350                LET R1 := R1 + #1
4335 021350 005201                                INC    R1
4336 021352                LET R2 := R2 - #1
4337 021352 005302                                DEC    R2
4338 021354                LET R3 := R3 + #1
4339 021354 005203                                INC    R3
4340
4341                ;REPEAT UNTIL ALL DATA HAS BFEN CHECKED
4342                ;
4343 021356                UNTIL R2 EQ #0
4344 021356 005702                                TST   R2
4345 021360 001346                                BNE   50002$
4346 021362                POP R3
4347 021362 012603                                MOV    (SP)+,R3
4348 021364                POP R2

```

4349	021364	012602					MOV	(SP)+,R2
4350	021366		POP R1					
4351	021366	012601					MOV	(SP)+,R1
4352	021370		ENDRTN					
4353	021370					50000\$:		
4354	021370	000241					CLC	
4355	021372					50001\$:		
4356	021372	000207					RTS	PC
4357								
4358								
4359			.. LOOPBACK MESSAGE AREA					
4360			..					
4361	021374	000000	LCOMP: 0					
4362	021376	121403	LOOPMG: .+100005					
4363	021400	0004'0	ELOOPB-2-					
4364	021402	220	000 201 LOOPM: .BYTE DLE,0,201,0					
4365	021405	000						
4366	021406	000	LOOPN: .BYTE 0					
4367	021407	000	LOOSTA: .BYTE 0					
4368	021410	000000	LCRCH: 0					
4369	021412	000200	LOOPB: .BLKW 128.					
4370	022012		ELOOPB:					
4371	022012	000000	LCRCD: 0					
4372								
4373	022014		ENDMOD					

```

4374 022014          BGNMOD PTCODE
4375                :
4376                :
4377                :
4378 022014          BGNHRD
4379 022014 000022          .WORD L10033-L$HARD/2
4380 022016          GPRMA  DEVICE,0,0,0,177776,YES
4381 022016 000031          .WORD  TSCODE
4382 022020 022062          .WORD  DEVICE
4383 022022 000000          .WORD  T$LOLIM
4384 022024 177776          .WORD  T$MILIM
4385 022026          GPRMA  VECTOR,2,0,0,400,YES
4386 022026 001031          .WORD  TSCODE
4387 022030 022100          .WORD  VECTOR
4388 022032 000000          .WORD  T$LOLIM
4389 022034 000400          .WORD  T$MILIM
4390 022036          GPRMD  PRIOR,4,0,340,0,7,YES
4391 022036 002032          .WORD  TSCODE
4392 022040 022115          .WORD  PRIOR
4393 022042 000340          .WORD  340
4394 022044 000000          .WORD  T$LOLIM
4395 022046 000007          .WORD  T$MILIM
4396 022050          GPRMD  STATION,4,0,177400,0,377,YES
4397 022050 002032          .WORD  TSCODE
4398 022052 022140          .WORD  STATION
4399 022054 177400          .WORD  177400
4400 022056 000000          .WORD  T$LOLIM
4401 022060 000377          .WORD  T$MILIM
4402 022062          ENDHRD
4403                :
4404 022062          .EVEN
4405                :
4406 022062 052504 020120 042101  DEVICE: .ASCIZ /DUP ADDRESS: /
4407 022070 051104 051505 035123
4408 022076 000040
4409 022100 052504 020120 042526  VECTOR: .ASCIZ /DUP VECTOR: /
4410 022106 052103 051117 020072
4411 022114      000
4412 022115      104 050125 041040  PRIOR: .ASCIZ /DUP BUS PRIORITY: /
4413 022122 051525 050040 044522
4414 022130 051117 052111 035131
4415 022136 000040
4416 022140 042524 046522 047111  STATION:.ASCIZ /TERMINAL ADDRESS: /
4417 022146 046101 040440 042104
4418 022154 042522 051523 020072
4419 022162      000
4420          022164          .EVEN
4421                :
4422                :
4423                :
4424                :
4425 022164          BGNSFT
4426 022164 000015          .WORD L10034-L$SOFT/2
4427 022166          GPRMD  ALIMIT,0,0,177777,0,77777,YES
4428 022166 000032          .WORD  TSCODE
4429 022170 022220          .WORD  ALIMIT

```

```
4430 022172 177777 .WORD 177777
4431 022174 000000 .WORD T$LOLIM
4432 022176 077777 .WORD T$HILIM
4433 022200 GPRML PATRN,2,177777,YES
4434 022200 001130 .WORD T$CODE
4435 022202 022267 .WORD PATRN
4436 022204 177777 .WORD 177777
4437 022206 GPRMD SWAIT,4,0,177777,0,77777,YES
4438 022206 002032 .WORD T$CODE
4439 022210 022332 .WORD SWAIT
4440 022212 177777 .WORD 177777
4441 022214 000000 .WORD T$LOLIM
4442 022216 077777 .WORD T$HILIM
4443 022220 ENDSFT
4444
4445 022220 .EVEN
4446 022220 047510 020127 040515 ALIMIT: .ASCIZ /HOW MANY INITIATE TRIES BEFORE ABORT? / L10034:
4447 022226 054516 044440 044516
4448 022234 044524 052101 020105
4449 022242 051124 042511 020123
4450 022250 042502 047506 042522
4451 022256 040440 047502 052122
4452 022264 020077 000
4453 022267 125 042523 040440 PATRN: .ASCIZ /USE ALL ZERO'S FOR LOOPBACK DATA? /
4454 022274 046114 055040 051105
4455 022302 023517 020123 047506
4456 022310 020122 047514 050117
4457 022316 040502 045503 042040
4458 022324 052101 037501 000040
4459 022332 040527 052111 043040 SWAIT: .ASCIZ @WAIT FACTOR FOR SELFTEST (2000 = 11/70 - 4000 = 11/10)?@
4460 022340 041501 047524 020122
4461 022346 047506 020122 042523
4462 022354 043114 042524 052123
4463 022362 024040 030062 030060
4464 022370 020060 020075 030461
4465 022376 033457 020060 020055
4466 022404 030064 030060 036440
4467 022412 030440 027461 030061
4468 022420 037451 000
4469 022424 .EVEN
4470 022424 ENDMOD
4471 022424 LASTAD
4472
4473 022424 .EVEN
4474 L$LAST:: .REQ DOCTOR.P11[400,1066]
```

4475  
4476 053220 000000  
053222 000000  
053224 000000  
053226 000000  
053232  
060200

.SBTTL DIAGNOSTIC SUPERVISOR -- LOW CORE SET UP  
.WORD 0 ;SPACE FOR USER POOL POINTER  
.WORD 0 ;SIZE  
.WORD 0 ;CHECKSUM (NOT CURRENTLY USED)  
.WORD 0 ;SIZE OF H.W. PTAB. ALLOCATION  
END.SJPV=+.+?  
.END 200

















LSDISP	011700	G	629	1241#		
LSDR	002230	G	642	736#		
LSDRCT	002070	G	642#			
LSDRS	002072	G	643#			
LSDRST	002234	G	643	738#		
LSDTP	002040	G	629#			
LSDU	011710	G	645	1260#		
LSDUT	002076	G	645#			
LSDVTY	002212	G	640	731#		
LSEF	002056	G	637#			
LSEFLG	002034	G	627#			
LSEXP1	002042	G	630#			
LSEXP2	002044	G	631#			
LSEXP3	002046	G	632#			
LSHARD	022016	G	620	4379	4390#	
LSHPCP	002016	G	620#			
LSHPTP	002022	G	622#			
LSHW	011656	G	622	1221	1222#	
LSICP	002104	G	648#			
LSINIT	011712	G	648	1268#		
LSLADP	002026	G	624#			
LSLAST	022424	G	624	4473#		
LSMREV	002050	G	633#			
LSNAME	002000	G	608#			
LSREPP	002066	G	641#			
LSREV	002010	G	616#			
LSRPT	011704	G	641	1250#		
LSSOFT	022166	G	621	4426	4427#	
LSSPC	002062	G	639#			
LSSPCP	002020	G	621#			
LSSPTP	002024	G	623#			
LSSTA	002030	G	625#			
LSSW	011670	G	623	1229	1230#	
LSTIML	002014	G	619#			
LSTIMU	002054	G	636#			
LSTIM1	002052	G	635#			
LSTST1	002100	G	646#			
LSUNIT	002012	G	618#	1309	1380	
L.CLK.	027022		4476#			
L10000	003704		912#			
L10001	003740		923#			
L10002	003772		936#			
L10003	004074		959#			
L10004	004126		974#			
L10005	004230		996#			
L10006	004236		1002#			
L10007	004240		1006#			
L10010	004332		1036#			
L10011	004460		1061#			
L10012	004644		1107#			
L10013	005000		1134#			
L10014	005126		1168#			
L10015	005216		1186#			
L10016	005270		1200#			
L10017	005272		1205#			
L10020	011666		1221	1227#		









SET.MA 027602  
SHIFT 052706 G  
SOM = 000201  
SPEC.U 027316  
SPV.SE 000400  
SSTACK 011654  
STACK = 000007  
STARTC 051576 G  
STATIO 022140  
STIME 002122  
STINPT 020274  
STRCHR 042610  
STRT = 000006  
STRIB 003650  
STRIC 003656  
STRISA 003655  
STRT.T 027374  
STWAIT 011674  
ST.SET 024226  
SUNIT. 027400  
SUPERV 025262  
SUPFLA 022670 G  
SUPV.T 023042 G  
SUP.PR 024000  
SVCGBL = 000000

4476#														
4476#														
660#	1901	3002												
4476#														
4476#														
1213#	1288													
666#	1769													
4476#														
4398	4416#													
698#														
3743	3843#													
4476#														
665#	893													
892#	1706													
898#	1726#													
897#	1714#	2136	3794											
4476#														
1232#	3630													
4476#														
4476#														
4476#														
4476#														
600#	607	608	616	617	618	619	620	621	622	623	624	625		
626	627	628	629	630	631	632	633	635	636	637	639	640		
641	642	643	644	645	646	647	648	649	652	731	736	738		
904	922	927	949	963	987	1001	1005	1009	1053	1071	1127	1143		
1177	1199	1204	1217	1222	1230	1241	1245	1250	1255	1260	1268	1434		
1458	2793	2893	3510	4375	4380	4427	4473#	4474						
4476#														
600#	601#	608	609	610	611	612	613	614	615	616	617	618		
619	620	621	622	623	624	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	731	733	735	736	737	738	905	906		
907	908	909	910	913	924	928	929	930	931	932	933	934		
937	951	952	953	954	955	956	957	960	966	967	968	969		
970	971	972	975	988	989	990	991	992	993	994	997	1003		
1007	1023	1024	1025	1026	1027	1028	1030	1031	1032	1033	1034	1037		
1054	1055	1056	1057	1058	1059	1062	1077	1078	1079	1080	1081	1082		
1084	1085	1086	1087	1088	1089	1091	1092	1093	1094	1095	1096	1098		
1099	1100	1101	1102	1103	1108	1128	1129	1130	1131	1132	1135	1144		
1145	1146	1147	1148	1161	1162	1163	1164	1165	1166	1169	1178	1179		
1180	1181	1182	1183	1184	1187	1201	1206	1221	1229	1240	1241	1242		
1252	1257	1262	1277	1278	1302	1303	1323	1324	1369	1370	1371	1417		
1455	1650	1651	1652	1738	1739	1740	1757	1758	1759	1781	1782	1783		
2037	2038	2039	2040	2119	2120	2121	2122	2139	2140	2141	2142	2187		
2188	2189	2190	2247	2248	2249	2250	2305	2306	2307	2308	2339	2340		
2341	2342	2362	2363	2364	2365	2452	2453	2454	2455	2475	2476	2477		
2478	2505	2540	2541	2542	2543	2623	2624	2665	2666	2667	2668	2669		
2670	2683	2684	2685	2686	2687	2688	2736	2737	2738	2739	2797	2798		
2857	2883	2897	2898	3183	3201	3230	3231	3293	3294	3295	3296	3337		
3338	3339	3340	3554	3555	3556	3557	3559	3561	3562	3585	3586	3587		
3588	3590	3592	3593	3604	3605	3606	3607	3609	3611	3612	3625	3626		
3654	3655	3656	3657	3659	3661	3662	3688	3689	3690	3691	3693	3695		
3696	3720	3721	3722	3723	3725	3727	3728	3756	3757	3758	3759	3761		

SVCNAM 031716  
SVCINS- 000001





TSSCLE= 010026	1434#	1454																		
TSSDU = 010024	1260#	1261																		
TSSHAR= 010033	4379#	4404																		
TSSHW = 010020	1221#	1227																		
TSSINI= 010025	1268#	1416																		
TSSMSG- 010017	904#	912	922#	923	927#	936	949#	959	963#	974	987#	996	1001#							
	1002	1005#	1006	1009#	1036	1053#	1061	1071#	1107	1127#	1134	1143#	1168							
	1177#	1186	1199#	1200	1204#	1205														
	1250#	1251																		
TSSRPT= 010022	4223#	4257#																		
TSSSEG= 010000	4426#	4445																		
TSSSOF= 010034	2793#	2882	2893#	3200																
TSSSRV= 010030	1229#	1234																		
TSSSW = 010021	3520#	3561	3592	3611	3661	3695	3727	3763	3771	4000#	4035	4205	4263							
TBSTES= 010032	1241	3520#																		
T1	017464	G	1242	4000#																
T2	020514	G	4476#																	
UNIT.D	022442	G	4476#																	
UNI.MA	027320		4476#																	
USER.P	022704	G	4476#																	
USER.T	022706	G	4476#																	
VALID.	023144		4476#																	
VAL.LA	023764		4476#																	
VAL.SW	027430		4476#																	
VECTOR	022100		4387	4409#																
VMSG	012166		1418#																	
WIDTH	036712		4476#																	
XBUFFE	015570		1640*	1706*	1723	2768#	2818	2826*	3810*	4119*										
XDONE	015104		2407*	2507	2513	2564#	2855*													
XEQDIA	051734	G	4476#																	
XEQSUB	051722	G	4476#																	
XEQ.CL	031444		4476#																	
XEQ.CM	026754		4476#																	
XEQ.IN	031126		4476#																	
XEQ.LA	025216		4476#																	
XEQ.OP	031220		4476#																	
XEQ.PR	024420		4476#																	
XEQ.TE	031264		4476#																	
XINT	015634	G	2684	2793#																
XMIT	014574		1643	1731	2394#	3814	4123													
XMTADD	015552		2693*	2694*	2761#	2822*	2824*	2842*	2852	2873*										
XMTBUF=	000006		675#	1098	2443*	2534*	2730*	3328*												
XMTCSR=	000004		674#	1091	1442*	2441*	2532*	2602*	2709*	2728*	2859*	3326*								
XSIZE	015572		1641*	1704*	2769#	2812	2828*	3812*	4121*											
XSTRT	012744		1582	1691#	4015	4230														
XTIME	050606	G	4476#																	
XTIMEN	051432		4476#																	
XTIMST	050630		4476#																	
XXDP.D	026774		4476#																	
X\$ALWA=	000000		600#																	
X\$FALS=	000040		600#																	
X\$OFFS=	000400		600#																	
X\$TRUF=	000020		600#																	
\$BGNLE=	177777		601#																	
\$BREG	027470		4476#																	
\$ENDAD	051706	G	4476#																	
\$ERFLG=	000400		601#	1012#	1013	1017#	1018	1021#	1022	1075#	1076	1150#	1151	1155#	1156					

1159#	1160	1288#	1289	1290#	1291	1307#	1308	1312#	1313	1328#	1329	1336#
1337	1338#	1339	1343#	1344	1345#	1346	1347#	1348	1349#	1350	1351#	1352
1355#	1356	1357#	1358	1367#	1368	1378#	1379	1383#	1384	1387#	1388	1398#
1399	1402#	1403	1404#	1405	1406#	1407	1440#	1441	1442#	1443	1449#	1450
1487#	1488	1493#	1494#	1497	1501#	1502	1503#	1504	1514#	1515	1516#	1517
1524#	1525	1532#	1533	1534#	1535	1536#	1537	1541#	1542	1543#	1544	1549#
1550#	1553	1557#	1558	1582#	1586#	1588#	1591#	1592	1593#	1594	1600#	1601
1602#	1603#	1604	1605#	1606	1608#	1609	1610#	1611	1615#	1634#	1643#	1654#
1656#	1662#	1704#	1705	1706#	1707	1711#	1712#	1713	1714#	1715	1720#	1731#
1742#	1744#	1750#	1761#	1763#	1769#	1770	1771#	1785#	1787#	1823#	1824#	1825
1829#	1830	1834#	1835#	1836	1840#	1841	1854#	1855	1856#	1857	1858#	1859
1860#	1861	1862#	1863	1895#	1896#	1897	1901#	1902	1906#	1907	1908#	1909
1910#	1911	1912#	1913	1917#	1918	1919#	1920	1924#	1925	1929#	1930	1931#
1932	1936#	1937#	1938	1939#	1940	1944#	1945#	1946	1947#	1948	1952#	1963#
1964	1968#	1969	1970#	1971	1975#	1986#	1987#	1988	1989#	1990	1991#	1992
1993#	1994	1998#	1999#	2000	2001#	2002#	2003	2030#	2047#	2049#	2059#	2060
2061#	2062	2063#	2072#	2074#	2077#	2078#	2079	2080#	2081	2092#	2103#	2104#
2105	2106#	2107	2108#	2109	2110#	2111	2126#	2128#	2146#	2148#	2154#	2155
2156#	2157	2158#	2169#	2170#	2171	2172#	2173	2174#	2175	2176#	2177	2178#
2179	2194#	2196#	2202#	2203	2204#	2205#	2206	2207#	2208	2252#	2254#	2260#
2268#	2269#	2270	2271#	2272	2273#	2274	2275#	2276	2277#	2278	2288#	2289#
2290	2291#	2292	2293#	2294	2301#	2302	2303#	2304	2310#	2312#	2318#	2319#
2320	2321#	2322	2323#	2324	2325#	2326#	2327	2328#	2329	2330#	2331	2344#
2346#	2358#	2359	2360#	2361	2367#	2369#	2405#	2406	2407#	2408	2409#	2410
2419#	2433#	2434	2435#	2436	2437#	2438	2439#	2440	2441#	2442	2443#	2444
2457#	2458	2465#	2466	2480#	2481	2482#	2483	2524#	2525	2526#	2527	2528#
2529	2530#	2531	2532#	2533	2534#	2535	2545#	2547#	2553#	2555#	2596#	2597
2602#	2603	2604#	2605	2606#	2607	2611#	2612	2621#	2622	2629#	2630#	2633
2634#	2635	2655#	2656#	2657	2658#	2659	2660#	2661	2662#	2663#	2664	2675#
2676	2677#	2678#	2679	2693#	2694#	2695	2696#	2697#	2698	2702#	2703	2704#
2705	2709#	2710	2711#	2712	2722#	2723	2724#	2725	2726#	2727	2728#	2729
2730#	2731	2741#	2743#	2751#	2752	2818#	2819	2820#	2821	2822#	2823	2824#
2825	2826#	2827	2828#	2829	2842#	2843	2844#	2845	2852#	2853#	2854	2855#
2856	2859#	2860	2873#	2874	2875#	2876	2908#	2909	2916#	2917	2921#	2922#
2923	2924#	2925	2926#	2927	2928#	2929	2935#	2936	2959#	2960	2961#	2962
2963#	2964	2965#	2966	2967#	2968	2972#	2973	2976#	2977	2978#	2979	2980#
2981	2982#	2983	2987#	2988	2994#	2995	3012#	3013	3014#	3015	3016#	3017
3018#	3019	3020#	3021	3025#	3026	3029#	3030	3031#	3032	3040#	3041	3045#
3046#	3047	3048#	3049	3050#	3051	3060#	3061	3073#	3074	3086#	3087	3088#
3089	3090#	3091	3092#	3093	3109#	3110	3111#	3112	3113#	3114	3115#	3116
3117#	3118	3125#	3126	3127#	3128	3133#	3134	3138#	3139	3143#	3144	3147#
3148	3159#	3160	3161#	3162	3168#	3169	3173#	3174#	3175	3176#	3177	3178#
3179	3185#	3186	3223#	3224	3233#	3234	3247#	3248	3259#	3260	3267#	3268
3269#	3270	3271#	3272	3273#	3274	3278#	3279	3285#	3286	3298#	3299	3300#
3302#	3311#	3312	3315#	3316#	3317	3320#	3321	3322#	3323	3324#	3325	3326#
3327	3328#	3329	3345#	3347#	3402#	3403	3404#	3405	3406#	3407	3412#	3413
3414#	3415	3416#	3417	3418#	3422	3423#	3424	3425#	3426	3427#	3428	3429#
3430	3431#	3432	3433#	3434	3435#	3438	3439#	3440	3441#	3442	3443#	3444
3445#	3449	3450#	3451	3452#	3453	3454#	3455	3462#	3463	3524#	3525	3529#
3530#	3531	3532#	3533	3537#	3549#	3569#	3590#	3619#	3620	3621#	3622	3628#
3629	3636#	3637#	3638	3639#	3640	3641#	3649#	3674#	3683#	3703#	3710#	3715#
3743#	3751#	3790#	3791	3792#	3793	3794#	3795	3799#	3810#	3811	3812#	3813
3814#	3821#	3822	3823#	3825#	3854#	3855#	3856	3866#	3867	3879#	3880	3881#
3882	3883#	3884	3885#	3886	3887#	3888	3889#	3890	3891#	3892	3893#	3894
3895#	3896	3907#	3909#	3931#	3933#	3949#	3951#	4004#	4005	4006#	4007	4008#
4009	4013#	4014	4015#	4023#	4045#	4046	4050#	4051	4055#	4056	4061#	4062

SFSBAND- 000310

SFSBAD= 000401

4069#	4070#	4071	4072#	4073	4077#	4078#	4079	4080#	4081	4087#	4088	4092#
4093	4097#	4108#	4119#	4120	4121#	4122	4123#	4131#	4153#	4163#	4180#	4188#
4193#	4221#	4228#	4229	4230#	4238#	4243#	4289#	4290	4291#	4292	4293#	4294
4307#	4308	4309#	4310	4311#	4327#	4329#	4335#	4336	4337#	4338	4339#	4340
601#	1016	1154	1311	1382	1500	1513	1556	1581	1628	1848	2058	2118
2138	2186	2246	2284	2300	2338	2474	2499	2515	2651	2805	2814	2838
2915	2939	2946	2954	2971	2986	3004	3024	3080	3102	3142	3167	3254
3263	3277	3411	3603	3702	3862	3870	3917	4049	4179	4306		
601#	1012	1016	1017	1021	1075	1150	1154	1155	1159	1288	1290	1307
1311	1312	1328	1336	1338	1343	1345	1347	1349	1351	1355	1357	1367
1378	1382	1383	1387	1398	1402	1404	1406	1440	1442	1449	1487	1493
1500	1501	1503	1513	1514	1516	1524	1532	1534	1536	1541	1543	1549
1556	1557	1581	1591	1593	1600	1602	1605	1608	1610	1630	1704	1706
1711	1714	1769	1823	1829	1834	1840	1850	1854	1856	1858	1860	1862
1895	1901	1906	1908	1910	1912	1917	1919	1924	1929	1931	1936	1939
1944	1947	1963	1968	1970	1986	1989	1991	1993	1998	2001	2058	2059
2061	2077	2080	2103	2106	2108	2110	2118	2138	2154	2156	2169	2172
2174	2176	2178	2186	2202	2204	2207	2246	2268	2271	2273	2275	2277
2284	2288	2291	2293	2300	2301	2303	2318	2321	2323	2325	2328	2330
2338	2358	2360	2405	2407	2409	2433	2435	2437	2439	2441	2443	2457
2465	2474	2480	2482	2499	2515	2524	2526	2528	2530	2532	2534	2596
2602	2604	2606	2611	2621	2629	2634	2651	2655	2658	2660	2662	2675
2677	2693	2696	2702	2704	2709	2711	2722	2724	2726	2728	2730	2751
2805	2814	2818	2820	2822	2824	2826	2828	2838	2842	2844	2852	2855
2859	2873	2875	2908	2915	2916	2921	2924	2926	2928	2935	2939	2948
2954	2959	2961	2963	2965	2967	2971	2972	2976	2978	2980	2982	2986
2987	2994	3007	3012	3014	3016	3018	3020	3024	3025	3029	3031	3040
3045	3048	3050	3060	3073	3080	3086	3088	3090	3092	3102	3109	3111
3113	3115	3117	3125	3127	3133	3138	3142	3143	3147	3159	3161	3167
3168	3173	3176	3178	3185	3223	3233	3247	3254	3259	3263	3267	3269
3271	3273	3277	3278	3285	3298	3311	3315	3320	3322	3324	3326	3328
3402	3404	3406	3411	3412	3414	3416	3418	3423	3425	3427	3429	3431
3433	3435	3439	3441	3443	3445	3450	3452	3454	3462	3524	3529	3532
3603	3619	3621	3628	3636	3639	3702	3790	3792	3794	3810	3812	3821
3854	3862	3866	3870	3879	3881	3883	3885	3887	3889	3891	3893	3895
3917	4004	4006	4008	4013	4045	4049	4050	4055	4061	4069	4072	4077
4080	4087	4092	4119	4121	4119	4228	4289	4291	4293	4306	4307	4309
4335	4337	4339										

SFSBLA= 000170  
 SFSBCLAS= 000150  
 SFSBDEC= 000220  
 SFSBDO = 000340  
 SFSBFAL= 000405  
 SFSBGDO= 000400

601#												
601#												
601#												
601#	1630	1850	3411									
601#	1012	1013	1014	1017	1018	1021	1022	1075	1076	1150	1151	1152
1155	1156	1159	1160	1283	1288	1289	1290	1291	1305	1307	1308	1309
1312	1313	1326	1328	1329	1336	1337	1338	1339	1343	1344	1345	1346
1347	1348	1349	1350	1351	1352	1355	1356	1357	1359	1367	1368	1376
1378	1379	1380	1383	1384	1387	1388	1398	1399	1402	1403	1404	1405
1406	1407	1440	1441	1442	1443	1449	1450	1487	1488	1493	1494	1497
1498	1501	1502	1503	1504	1511	1514	1515	1516	1517	1524	1525	1532
1533	1534	1535	1536	1537	1541	1542	1543	1544	1549	1550	1553	1554
1557	1558	1579	1582	1584	1591	1592	1593	1594	1600	1601	1602	1603
1604	1605	1606	1608	1609	1610	1611	1615	1625	1626	1634	1643	1648
1662	1704	1705	1706	1707	1711	1712	1713	1714	1715	1720	1731	1736
1750	1755	1769	1770	1771	1779	1823	1824	1825	1829	1830	1834	1835
1836	1840	1841	1845	1846	1854	1855	1856	1857	1858	1859	1860	1861

1862	1863	1895	1896	1897	1901	1902	1906	1907	1908	1909	1910	1911
1912	1913	1917	1918	1919	1920	1924	1925	1929	1930	1931	1932	1936
1937	1938	1939	1940	1944	1945	1946	1947	1948	1952	1963	1964	1968
1964	1970	1971	1975	1986	1987	1988	1989	1990	1991	1992	1993	1994
1998	1999	2000	2001	2002	2003	2030	2035	2056	2059	2060	2061	2062
2063	2068	2077	2078	2079	2080	2081	2092	2103	2104	2105	2106	2107
2108	2109	2110	2111	2116	2136	2154	2155	2156	2157	2158	2169	2170
2171	2172	2173	2174	2175	2176	2177	2178	2179	2184	2202	2203	2204
2205	2206	2207	2208	2244	2260	2268	2269	2270	2271	2272	2273	2274
2275	2276	2277	2278	2282	2288	2289	2290	2291	2292	2293	2294	2298
2301	2302	2303	2304	2318	2319	2320	2321	2322	2323	2324	2325	2326
2327	2328	2329	2330	2331	2336	2358	2359	2360	2361	2405	2406	2407
2408	2409	2410	2419	2424	2433	2434	2435	2436	2437	2438	2439	2440
2441	2442	2443	2444	2457	2458	2465	2466	2472	2480	2481	2482	2483
2497	2513	2524	2525	2526	2527	2528	2529	2530	2531	2532	2533	2534
2535	2596	2597	2602	2603	2604	2605	2606	2607	2611	2612	2621	2622
2629	2630	2633	2634	2635	2649	2655	2656	2657	2658	2659	2660	2661
2662	2663	2664	2675	2676	2677	2678	2679	2693	2694	2695	2696	2697
2698	2702	2703	2704	2705	2709	2710	2711	2712	2722	2723	2724	2725
2726	2727	2728	2729	2730	2731	2751	2752	2803	2812	2818	2819	2820
2821	2822	2823	2824	2825	2826	2827	2828	2829	2836	2842	2843	2844
2845	2852	2853	2854	2855	2856	2859	2860	2873	2874	2875	2876	2908
2909	2913	2916	2917	2921	2922	2923	2924	2925	2926	2927	2928	2929
2935	2936	2937	2944	2952	2959	2960	2961	2962	2963	2964	2965	2966
2967	2968	2969	2972	2973	2976	2977	2978	2979	2980	2981	2982	2983
2984	2987	2988	2994	2995	3002	3012	3013	3014	3015	3016	3017	3018
3019	3020	3021	3022	3025	3026	3029	3030	3031	3032	3040	3041	3045
3046	3047	3048	3049	3050	3051	3060	3061	3073	3074	3078	3086	3087
3088	3089	3090	3091	3092	3093	3100	3109	3110	3111	3112	3113	3114
3115	3116	3117	3118	3125	3126	3127	3128	3133	3134	3138	3139	3140
3143	3144	3147	3148	3159	3160	3161	3162	3165	3168	3169	3173	3174
3175	3176	3177	3178	3179	3185	3186	3223	3224	3233	3234	3247	3248
3252	3259	3260	3261	3267	3268	3269	3270	3271	3272	3273	3274	3275
3278	3279	3285	3286	3298	3299	3311	3312	3315	3316	3317	3320	3321
3322	3323	3324	3325	3326	3327	3328	3329	3402	3403	3404	3405	3406
3407	3408	3409	3412	3413	3414	3415	3416	3417	3418	3422	3423	3424
3425	3426	3427	3428	3429	3430	3431	3432	3433	3434	3435	3438	3439
3440	3441	3442	3443	3444	3445	3449	3450	3451	3452	3453	3454	3458
3462	3463	3524	3525	3529	3530	3531	3532	3533	3537	3542	3549	3569
3578	3580	3601	3619	3620	3621	3622	3628	3629	3636	3637	3638	3639
3640	3641	3647	3649	3674	3681	3683	3700	3703	3708	3710	3715	3743
3749	3751	3790	3791	3792	3793	3794	3795	3799	3810	3811	3812	3813
3814	3819	3821	3822	3854	3855	3856	3860	3866	3867	3868	3879	3880
3881	3882	3883	3884	3885	3886	3887	3888	3889	3890	3891	3892	3893
3894	3895	3896	3915	4004	4005	4006	4007	4008	4009	4013	4014	4015
4021	4023	4045	4046	4047	4050	4051	4055	4056	4061	4062	4069	4070
4071	4072	4073	4077	4078	4079	4080	4081	4087	4088	4092	4093	4097
4108	4119	4120	4121	4122	4123	4129	4131	4153	4161	4163	4177	4180
4186	4188	4193	4221	4228	4229	4230	4236	4238	4243	4289	4290	4291
4292	4293	4294	4304	4307	4308	4309	4310	4311	4335	4336	4337	4338
4339	4340											
601#	1014	1019	1152	1157	1283	1292	1305	1309	1314	1316	1318	1326
1330	1376	1380	1385	1389	1498	1505	1511	1521	1526	1554	1559	1579
1584	1589	1595	1626	1648	1657	1736	1745	1755	1764	1779	1788	1846
2035	2050	2056	2068	2075	2086	2116	2129	2136	2149	2184	2197	2244
2255	2282	2298	2316	2336	2347	2349	2355	2370	2372	2424	2462	2467

2472	2484	2497	2513	2548	2550	2556	2649	2717	2744	2803	2812	2833
2836	2846	2861	2863	2870	2877	2913	2933	2937	2944	2946	2952	2969
2974	2984	2989	2996	3002	3004	3022	3027	3033	3052	3054	3062	3067
3078	3094	3100	3140	3145	3152	3165	3187	3189	3191	3193	3252	3261
3275	3280	3287	3303	3305	3348	3542	3564	3578	3595	3601	3614	3647
3664	3681	3698	3700	3708	3730	3732	3749	3766	3819	3826	3860	3868
3910	3915	3934	3952	3954	3956	4021	4038	4047	4052	4057	4129	4141
4161	4173	4177	4186	4208	4210	4236	4253	4304	4330			
601#												
601#												
601#												
601#	1012	1013	1014	1017	1018	1021	1022	1075	1076	1150	1151	1152
1155	1156	1159	1160	1288	1289	1290	1291	1307	1308	1309	1312	1313
1328	1329	1336	1337	1338	1339	1343	1344	1345	1346	1347	1348	1349
1350	1351	1352	1355	1356	1357	1358	1367	1368	1378	1379	1380	1383
1384	1387	1388	1398	1399	1402	1403	1404	1405	1406	1407	1440	1441
1442	1443	1449	1450	1487	1488	1493	1497	1498	1501	1502	1503	1504
1511	1514	1515	1516	1517	1524	1525	1532	1533	1534	1535	1536	1537
1541	1542	1543	1544	1549	1553	1554	1557	1558	1579	1588	1591	1592
1593	1594	1600	1601	1602	1604	1605	1606	1608	1609	1610	1611	1626
1628	1656	1704	1705	1706	1707	1711	1713	1714	1715	1744	1763	1769
1770	1787	1823	1825	1829	1830	1834	1836	1840	1841	1846	1848	1854
1855	1856	1857	1858	1859	1860	1861	1862	1863	1895	1897	1901	1902
1906	1907	1908	1909	1910	1911	1912	1913	1917	1918	1919	1920	1924
1925	1929	1930	1931	1932	1936	1938	1939	1940	1944	1946	1947	1948
1963	1964	1968	1969	1970	1971	1986	1988	1989	1990	1991	1992	1993
1994	1998	2000	2001	2003	2049	2056	2059	2060	2061	2062	2074	2077
2079	2080	2081	2085	2103	2105	2106	2107	2108	2109	2110	2111	2116
2128	2148	2154	2155	2156	2157	2169	2171	2172	2173	2174	2175	2176
2177	2178	2179	2184	2196	2202	2203	2204	2206	2207	2208	2244	2254
2268	2270	2271	2272	2273	2274	2275	2276	2277	2278	2282	2288	2290
2291	2292	2293	2294	2298	2301	2302	2303	2304	2312	2318	2320	2321
2322	2323	2324	2325	2327	2328	2329	2330	2331	2336	2346	2358	2359
2360	2361	2369	2405	2406	2407	2408	2409	2410	2433	2434	2435	2436
2437	2438	2439	2440	2441	2442	2443	2444	2457	2458	2465	2466	2472
2480	2481	2482	2483	2497	2513	2524	2525	2526	2527	2528	2529	2530
2531	2532	2533	2534	2535	2547	2555	2596	2597	2602	2603	2604	2605
2606	2607	2611	2612	2621	2622	2629	2633	2634	2635	2649	2655	2657
2658	2659	2660	2661	2662	2664	2675	2676	2677	2679	2693	2695	2696
2698	2702	2703	2704	2705	2709	2710	2711	2712	2722	2723	2724	2725
2726	2727	2728	2729	2730	2731	2743	2751	2752	2803	2812	2818	2819
2820	2821	2822	2823	2824	2825	2826	2827	2828	2829	2836	2842	2843
2844	2845	2852	2854	2855	2856	2859	2860	2873	2874	2875	2876	2908
2909	2913	2916	2917	2921	2923	2924	2925	2926	2927	2928	2929	2935
2936	2937	2959	2960	2961	2962	2963	2964	2965	2966	2967	2968	2969
2972	2973	2976	2977	2978	2979	2980	2981	2982	2983	2984	2987	2988
2994	2995	3012	3013	3014	3015	3016	3017	3018	3019	3020	3021	3022
3025	3026	3029	3030	3031	3032	3040	3041	3045	3047	3048	3049	3050
3051	3060	3061	3073	3074	3078	3086	3087	3088	3089	3090	3091	3092
3093	3100	3109	3110	3111	3112	3113	3114	3115	3116	3117	3118	3125
3126	3127	3128	3133	3134	3138	3139	3140	3143	3144	3147	3148	3159
3160	3161	3162	3165	3168	3169	3173	3175	3176	3177	3178	3179	3185
3186	3223	3224	3233	3234	3247	3248	3252	3259	3260	3261	3267	3268
3269	3270	3271	3272	3273	3274	3275	3278	3279	3285	3286	3298	3299
3302	3311	3312	3315	3317	3320	3321	3322	3323	3324	3325	3326	3327
3328	3329	3347	3402	3403	3404	3405	3406	3407	3409	3412	3413	3414

SFSINC= 000210  
SFSLOO= 000200  
SFSNAM= 000160  
SFSNO = 000403



	3415	3416	3417	3418	3422	3423	3424	3425	3426	3427	3428	3429	3430
	3431	3432	3433	3434	3435	3438	3439	3440	3441	3442	3443	3444	3445
	3449	3450	3451	3452	3453	3454	3458	3462	3463	3524	3525	3529	3531
	3532	3533	3601	3619	3620	3621	3622	3628	3629	3636	3638	3639	3640
	3700	3790	3791	3792	3793	3794	3795	3810	3811	3812	3813	3821	3822
	3825	3854	3856	3866	3867	3879	3880	3881	3882	3883	3884	3885	3886
	3887	3888	3889	3890	3891	3892	3893	3894	3895	3896	3909	3933	3951
	4004	4005	4006	4007	4008	4009	4013	4014	4045	4046	4047	4050	4051
	4055	4056	4061	4062	4069	4071	4072	4073	4077	4079	4080	4081	4087
	4088	4092	4093	4119	4120	4121	4122	4177	4228	4229	4289	4290	4291
	4292	4293	4294	4307	4308	4309	4310	4329	4335	4336	4337	4338	4339
	4340												
BFSBR - 000320	601#	1016	1154	1311	1382	1500	1513	1556	1581	1628	1630	1848	1850
	2058	2118	2138	2186	2246	2284	2300	2338	2474	2499	2515	2651	2805
	2814	2838	2915	2939	2946	2948	2954	2971	2986	3004	3024	3080	3102
	3142	3167	3254	3263	3277	3411	3603	3702	3862	3870	3917	4049	4179
	4306												
BFSRTI= 000350	601#												
BFSRTM= 000300	601#	1476	1561	1578	1676	1692	1793	1813	1870	1887	2008	2021	2211
	2233	2377	2395	2558	2584	2753	3219	3350	3366	3367	3389	3470	3785
	3828	3844	3964	4277	4353								
BFSSEL= 000140	601#												
BFSME= 000330	601#	1016	1154	1311	1382	1500	1513	1556	1581	2058	2118	2138	2186
	2246	2284	2300	2338	2474	2499	2515	2651	2805	2814	2838	2915	2939
	2948	2954	2971	2986	3007	3024	3080	3102	3142	3167	3254	3263	3277
	3603	3702	3862	3870	3917	4049	4179	4306					
BFSSTRU= 000404	601#												
BFSUNT= 000130	601#	1353	1359	1365	1391	1400	1408	2414	2490	2503	2507	2616	2640
	3228	3238	3623	3630	3669	3737	4059	4063	4148	4215	4298	4344	
BFSWHI= 000120	601#	1625	1626	1628	1670	1845	1846	1848	1864	2946	3004	3408	3409
	3459												
BFSVES= 000402	601#	1013	1014	1018	1019	1022	1076	1151	1152	1156	1157	1160	1283
	1289	1291	1292	1305	1308	1309	1313	1314	1316	1318	1326	1329	1330
	1337	1339	1344	1346	1348	1350	1352	1356	1358	1368	1376	1379	1380
	1384	1385	1388	1389	1399	1403	1405	1407	1441	1443	1450	1488	1497
	1498	1502	1504	1505	1511	1515	1517	1521	1525	1526	1533	1535	1537
	1542	1544	1553	1554	1558	1559	1579	1584	1589	1592	1594	1595	1601
	1604	1606	1609	1611	1626	1628	1648	1657	1705	1707	1713	1715	1736
	1745	1755	1764	1770	1779	1788	1825	1830	1836	1841	1846	1848	1855
	1857	1859	1861	1863	1897	1902	1907	1909	1911	1913	1918	1920	1925
	1930	1932	1938	1940	1946	1948	1964	1969	1971	1988	1990	1992	1994
	2000	2003	2035	2050	2056	2060	2062	2068	2075	2079	2081	2086	2105
	2107	2109	2111	2116	2129	2136	2149	2155	2157	2171	2173	2175	2177
	2179	2184	2197	2203	2206	2208	2244	2255	2270	2272	2274	2276	2278
	2282	2290	2292	2294	2298	2302	2304	2316	2320	2322	2324	2327	2329
	2331	2336	2347	2349	2355	2359	2361	2370	2372	2406	2408	2410	2424
	2434	2436	2438	2440	2442	2444	2458	2462	2466	2467	2472	2481	2483
	2484	2497	2513	2525	2527	2529	2531	2533	2535	2548	2550	2556	2597
	2603	2605	2607	2612	2622	2633	2635	2649	2657	2659	2661	2664	2676
	2679	2695	2698	2703	2705	2710	2712	2717	2723	2725	2727	2729	2731
	2744	2752	2803	2812	2819	2821	2823	2825	2827	2829	2833	2836	2843
	2845	2846	2854	2856	2860	2861	2863	2870	2874	2876	2877	2909	2913
	2917	2923	2925	2927	2929	2933	2936	2937	2944	2946	2952	2960	2962
	2964	2966	2968	2969	2973	2974	2977	2979	2981	2983	2984	2988	2989
	2995	2996	3002	3004	3013	3015	3017	3019	3021	3022	3026	3027	3030
	3032	3033	3041	3047	3049	3051	3052	3054	3061	3062	3067	3074	3078

3087	3089	3091	3093	3094	3100	3110	3112	3114	3116	3118	3126	3128
3134	3139	3140	3144	3145	3148	3152	3160	3162	3165	3169	3175	3177
3179	3186	3187	3189	3191	3193	3224	3234	3248	3252	3260	3261	3268
3270	3272	3274	3275	3279	3280	3286	3287	3299	3303	3305	3312	3317
3321	3323	3325	3327	3329	3348	3403	3405	3407	3409	3413	3415	3417
3422	3424	3426	3428	3430	3432	3434	3438	3440	3442	3444	3449	3451
3453	3458	3463	3525	3531	3533	3542	3564	3578	3595	3601	3614	3620
3622	3629	3638	3640	3647	3664	3681	3698	3700	3708	3730	3732	3749
3766	3791	3793	3795	3811	3813	3819	3822	3826	3856	3860	3867	3868
3880	3882	3884	3886	3888	3890	3892	3894	3896	3910	3915	3934	3952
3954	3956	4005	4007	4009	4014	4021	4038	4046	4047	4051	4052	4056
4057	4062	4071	4073	4079	4081	4088	4093	4120	4122	4129	4141	4161
4173	4177	4186	4208	4210	4229	4236	4253	4290	4292	4294	4304	4308
4310	4330	4336	4338	4340								
601#	1014#	1019#	1152#	1157#	1283#	1305#	1309#	1314#	1316#	1318#	1326#	1330#
1376#	1380#	1385#	1389#	1498#	1505#	1511#	1526#	1554#	1559#	1579#	1584#	1589#
1595#	1648#	1657#	1736#	1745#	1755#	1764#	1779#	1788#	2035#	2050#	2056#	2068#
2075#	2086#	2116#	2129#	2136#	2149#	2184#	2197#	2244#	2282#	2298#	2336#	2347#
2349#	2370#	2377#	2424#	2467#	2472#	2484#	2497#	2513#	2548#	2556#	2649#	2744#
2803#	2812#	2836#	2861#	2863#	2877#	2913#	2933#	2937#	2944#	2952#	2969#	2974#
2984#	2989#	3002#	3022#	3027#	3052#	3054#	3062#	3078#	3100#	3140#	3145#	3165#
3187#	3189#	3191#	3193#	3252#	3261#	3275#	3280#	3303#	3348#	3542#	3564#	3578#
3595#	3601#	3614#	3647#	3664#	3681#	3698#	3700#	3708#	3730#	3732#	3749#	3766#
3819#	3826#	3860#	3868#	3915#	3952#	3954#	3956#	4021#	4038#	4047#	4057#	4129#
4141#	4161#	4173#	4177#	4186#	4208#	4210#	4236#	4253#	4304#	4330#		
1014#	1019	1152#	1157	1283#	1318	1326#	1330	1376#	1389	1498#	1505	1511#
1526	1554#	1559	1579#	1595	1648#	1657	1736#	1745	1755#	1764	1779#	1788
2035#	2050	2056#	2086	2116#	2129	2136#	2149	2184#	2197	2244#	2372	2424#
2467	2472#	2484	2497#	2556	2649#	2744	2803#	2877	2913#	2933	2937#	3193
3252#	3348	3542#	3564	3578#	3595	3601#	3614	3647#	3664	3681#	3698	3700#
3732	3749#	3766	3819#	3826	3860#	3956	4021#	4038	4047#	4057	4129#	4141
4161#	4173	4177#	4210	4236#	4253	4304#	4330					
1305#	1316	1380#	1385	1584#	1589	2068#	2075	2282#	2370	2513#	2548	2812#
2863	2944#	3062	3078#	3191	3261#	3303	3708#	3730	3868#	3954	4186#	4208
1309#	1314	2298#	2349	2836#	2861	2952#	3054	3100#	3189	3275#	3280	3915#
3952												
2336#	2347	2969#	2974	2984#	2989	3002#	3052	3140#	3145	3165#	3187	
3022#	3027											
601#	1015	1019	1153	1157	1283	1292	1293	1305	1310	1314	1316	1318
1326	1330	1353	1360	1365	1376	1381	1385	1389	1392	1400	1409	1475
1499	1505	1512	1521	1522	1526	1555	1559	1561	1562	1577	1580	1584
1587	1589	1595	1625	1627	1629	1648	1655	1657	1670	1671	1676	1678
1691	1736	1743	1745	1755	1762	1764	1779	1786	1788	1793	1795	1812
1845	1847	1849	1864	1865	1870	1871	1886	2008	2009	2020	2035	2048
2050	2057	2068	2073	2075	2084	2086	2117	2127	2129	2137	2147	2149
2185	2195	2197	2211	2213	2232	2245	2253	2255	2283	2299	2311	2316
2337	2345	2347	2349	2355	2356	2368	2370	2372	2377	2379	2394	2414
2424	2462	2463	2467	2473	2484	2491	2498	2503	2508	2514	2546	2548
2550	2551	2554	2556	2558	2560	2583	2616	2641	2643	2644	2650	2717
2718	2742	2744	2753	2755	2804	2813	2833	2834	2837	2846	2847	2861
2863	2870	2871	2877	2914	2933	2938	2945	2947	2953	2970	2974	2985
2989	2996	2997	3003	3005	3006	3023	3027	3033	3034	3052	3054	3062
3067	3068	3079	3094	3095	3101	3141	3145	3152	3153	3166	3187	3189
3191	3193	3218	3228	3239	3241	3242	3253	3262	3276	3280	3287	3288
3301	3303	3305	3306	3346	3348	3350	3352	3365	3367	3368	3388	3408
3410	3459	3460	3470	3471	3542	3564	3578	3595	3602	3614	3623	3631

\$IFLEV= 177777

\$ISK0 = 000001

\$ISK1 = 000001

\$ISK2 = 000001

\$ISK3 = 000001

\$ISK4 = 000001

\$LOCIA= 177777

Si STIN- 000001

3647	3664	3669	3681	3698	3701	3708	3730	3732	3738	3749	3766	3784
3819	3824	3826	3828	3830	3843	3861	3869	3908	3910	3916	3932	3934
3950	3952	3954	3956	3964	3966	4021	4038	4048	4052	4053	4057	4059
4064	4129	4141	4148	4161	4173	4178	4186	4208	4210	4216	4236	4253
4276	4298	4305	4328	4330	4345	4353	4355					
601#	1012	1014	1015	1017	1021	1073	1075	1105	1150	1152	1153	1155
1159	1272	1283	1288	1290	1292	1305	1307	1309	1310	1312	1326	1328
1336	1338	1343	1345	1347	1349	1351	1355	1357	1359	1360	1367	1376
1378	1380	1381	1383	1387	1391	1392	1398	1402	1404	1406	1408	1409
1414	1438	1440	1442	1444	1449	1480	1481	1482	1487	1493	1494	1495
1496	1498	1499	1501	1503	1511	1512	1514	1516	1521	1524	1532	1534
1536	1541	1543	1549	1550	1551	1552	1554	1555	1557	1563	1564	1565
1579	1580	1582	1584	1586	1587	1591	1593	1600	1602	1603	1605	1608
1610	1615	1616	1617	1618	1626	1627	1628	1629	1634	1635	1636	1637
1638	1639	1640	1641	1643	1648	1654	1655	1662	1663	1664	1665	1670
1677	1679	1695	1696	1697	1704	1706	1711	1712	1714	1720	1721	1722
1723	1724	1725	1726	1731	1736	1742	1743	1750	1755	1761	1762	1769
1771	1772	1773	1774	1779	1785	1786	1794	1796	1797	1798	1816	1817
1818	1823	1824	1829	1834	1835	1840	1846	1847	1848	1849	1854	1856
1858	1860	1862	1864	1872	1873	1874	1888	1890	1895	1896	1901	1906
1908	1910	1912	1917	1919	1924	1929	1931	1936	1937	1939	1944	1945
1947	1952	1953	1954	1955	1956	1957	1958	1963	1968	1970	1975	1976
1977	1978	1979	1980	1981	1986	1987	1989	1991	1993	1998	1999	2001
2002	2004	2006	2010	2025	2030	2035	2045	2047	2048	2056	2057	2059
2061	2063	2064	2065	2066	2068	2070	2072	2073	2077	2078	2080	2082
2084	2092	2093	2094	2095	2096	2097	2098	2103	2104	2106	2108	2110
2116	2117	2124	2126	2127	2136	2137	2144	2146	2147	2154	2156	2158
2159	2160	2161	2162	2163	2164	2169	2170	2172	2174	2176	2178	2184
2185	2192	2194	2195	2202	2204	2205	2207	2209	2212	2214	2236	2237
2238	2244	2245	2252	2253	2260	2261	2262	2265	2264	2265	2266	2268
2269	2271	2273	2275	2277	2282	2283	2288	2289	2291	2293	2298	2299
2301	2303	2310	2311	2318	2319	2321	2323	2325	2326	2328	2330	2336
2337	2344	2345	2355	2358	2360	2367	2368	2378	2380	2381	2382	2398
2399	2400	2405	2407	2409	2419	2424	2426	2428	2433	2435	2437	2439
2441	2443	2445	2447	2457	2462	2465	2472	2473	2480	2482	2490	2491
2497	2498	2507	2508	2513	2514	2516	2518	2524	2526	2528	2530	2532
2534	2536	2538	2545	2546	2550	2553	2554	2559	2561	2562	2563	2587
2588	2589	2591	2596	2602	2604	2606	2611	2621	2629	2630	2631	2632
2634	2640	2641	2642	2643	2649	2650	2655	2656	2658	2660	2662	2663
2675	2677	2678	2693	2694	2696	2697	2702	2704	2709	2711	2717	2720
2722	2724	2726	2728	2730	2732	2734	2741	2742	2749	2751	2754	2756
2757	2758	2803	2804	2810	2812	2813	2818	2820	2822	2824	2826	2828
2833	2836	2837	2842	2844	2846	2852	2853	2855	2859	2865	2870	2873
2875	2900	2908	2913	2914	2916	2921	2922	2924	2926	2928	2930	2935
2937	2938	2944	2945	2946	2947	2952	2953	2959	2961	2963	2965	2967
2969	2970	2972	2976	2978	2980	2982	2984	2985	2987	2994	2996	3002
3003	3004	3005	3012	3014	3016	3018	3020	3022	3023	3025	3029	3031
3033	3040	3045	3046	3048	3050	3060	3067	3073	3078	3079	3086	3088
3090	3092	3094	3100	3101	3107	3109	3111	3113	3115	3117	3119	3125
3127	3133	3138	3140	3141	3143	3147	3152	3159	3161	3165	3166	3168
3173	3174	3176	3178	3185	3198	3223	3233	3238	3239	3240	3241	3247
3252	3253	3259	3261	3262	3267	3269	3271	3273	3275	3276	3278	3285
3287	3298	3300	3301	3305	3311	3313	3315	3316	3318	3320	3322	3324
3326	3328	3330	3332	3345	3346	3351	3353	3369	3392	3393	3394	3396
3398	3400	3402	3404	3406	3409	3410	3412	3414	3416	3418	3419	3420
3421	3423	3425	3427	3429	3431	3433	3435	3436	3437	3439	3441	3443

3445	3446	3447	3448	3450	3452	3454	3455	3456	3457	3459	3462	3464
3466	3468	3472	3473	3474	3524	3529	3530	3532	3537	3542	3549	3550
3551	3552	3569	3570	3571	3578	3580	3581	3582	3583	3601	3602	3619
3621	3628	3630	3631	3636	3637	3639	3641	3647	3649	3650	3651	3652
3674	3675	3676	3681	3683	3684	3685	3686	3700	3701	3703	3708	3710
3711	3712	3713	3715	3716	3717	3718	3737	3738	3743	3749	3751	3752
3753	3754	3790	3792	3794	3799	3800	3801	3802	3803	3804	3805	3810
3812	3814	3819	3821	3823	3824	3829	3831	3845	3847	3849	3854	3855
3860	3861	3866	3868	3869	3879	3881	3883	3885	3887	3889	3891	3893
3895	3901	3903	3905	3907	3908	3915	3916	3925	3927	3929	3931	3932
3943	3945	3947	3949	3950	3958	3960	3962	3965	3967	4004	4006	4008
4013	4015	4021	4023	4024	4025	4026	4043	4045	4047	4048	4050	4052
4055	4061	4063	4064	4069	4070	4072	4077	4078	4080	4082	4087	4092
4097	4098	4099	4100	4101	4102	4103	4108	4109	4110	4111	4112	4113
4114	4119	4121	4123	4129	4131	4132	4133	4134	4153	4154	4155	4161
4163	4164	4165	4166	4177	4178	4180	4186	4188	4189	4190	4191	4193
4194	4195	4196	4215	4216	4221	4228	4230	4236	4238	4239	4240	4241
4243	4244	4245	4246	4278	4280	4282	4289	4291	4293	4304	4305	4307
4309	4311	4312	4313	4314	4321	4323	4325	4327	4328	4335	4337	4339
4344	4345	4347	4349	4351	4354	4356						
601#	1019	1157	1293	1314	1316	1318	1330	1353	1365	1385	1389	1400
1475	1505	1522	1526	1559	1561	1562	1577	1589	1595	1625	1657	1671
1676	1678	1691	1745	1764	1788	1793	1795	1812	1845	1865	1870	1871
1886	2008	2009	2020	2050	2075	2086	2129	2149	2197	2211	2213	2232
2255	2316	2347	2349	2356	2370	2372	2377	2379	2394	2414	2463	2467
2484	2503	2548	2551	2556	2558	2560	2583	2616	2644	2718	2744	2753
2755	2834	2847	2861	2863	2871	2877	2933	2974	2989	2997	3006	3027
3034	3052	3054	3062	3068	3095	3145	3153	3187	3189	3191	3193	3218
3228	3242	3280	3288	3303	3306	3348	3350	3352	3365	3367	3368	3388
3408	3460	3470	3471	3564	3595	3614	3623	3664	3669	3698	3730	3732
3766	3784	3826	3828	3830	3843	3910	3934	3952	3954	3956	3964	3966
4038	4053	4057	4059	4141	4148	4173	4208	4210	4253	4276	4298	4330
4353	4355											
601#	1014#	1019#	1152#	1157#	1283#	1292	1305#	1309#	1314#	1316#	1318#	1326#
1330#	1353#	1359#	1365#	1376#	1380#	1385#	1389#	1391#	1400#	1408#	1476#	1498#
1505#	1511#	1521	1526#	1554#	1559#	1561#	1578#	1579#	1584#	1589#	1595#	1625#
1648#	1657#	1670#	1676#	1692#	1736#	1745#	1755#	1764#	1779#	1788#	1793#	1813#
1845#	1864#	1870#	1887#	2008#	2021#	2035#	2050#	2056#	2068#	2075#	2086#	2116#
2129#	2136#	2149#	2184#	2197#	2211#	2233#	2244#	2255	2282#	2298#	2316	2336#
2347#	2349#	2355	2370#	2372#	2377#	2395#	2414#	2424#	2462	2467#	2472#	2484#
2490#	2497#	2503#	2507#	2513#	2548#	2550	2556#	2558#	2584#	2616#	2640#	2649#
2717	2744#	2753#	2803#	2812#	2833	2836#	2846	2861#	2863#	2870	2877#	2913#
2933#	2937#	2944#	2952#	2969#	2974#	2984#	2989#	2996	3002#	3022#	3027#	3033
3052#	3054#	3062#	3067	3078#	3094	3100#	3140#	3145#	3152	3165#	3187#	3189#
3191#	3193#	3219#	3228#	3238#	3252#	3261#	3275#	3280#	3287	3303#	3305	3348#
3350#	3366#	3367#	3389#	3408#	3459#	3470#	3542#	3564#	3578#	3595#	3601#	3614#
3623#	3630#	3647#	3664#	3669#	3681#	3698#	3700#	3708#	3730#	3732#	3737#	3749#
3766#	3785#	3819#	3826#	3828#	3844#	3860#	3868#	3910	3915#	3934	3952#	3954#
3956#	3964#	4021#	4038#	4047#	4052	4057#	4059#	4063#	4129#	4141#	4148#	4161#
4173#	4177#	4186#	4208#	4210#	4215#	4236#	4253#	4277#	4298#	4304#	4330#	4344#
4353#												
1014#	1019	1152#	1157	1283#	1292	1318	1326#	1330	1353#	1359	1365#	1391
1400#	1408	1476#	1561	1578#	1676	1692#	1793	1813#	1870	1887#	2008	2021#
2211	2233#	2377	2395#	2558	2584#	2753	2803#	2870	2877	2913#	2933	2937#
3067	3193	3219#	3350	3366#	3367	3389#	3470	3542#	3564	3578#	3595	3601#
3614	3623#	3630	3647#	3664	3669#	3737	3749#	3766	3785#	3828	3844#	3964

\$LSTTA= 000001

\$NESTL= 177777

\$NSKO - 000300

	4021#	4038	4047#	4052	4057	4059#	4063	4129#	4141	4148#	4215	4236#	4253
BNSK1 = 000130	4277#	4353											
	1305#	1316	1376#	1389	1498#	1505	1511#	1521	1526	1554#	1559	1579#	1595
	1625#	1670	1736#	1745	1755#	1764	1779#	1788	1845#	1864	2035#	2050	2056#
	2086	2116#	2129	2136#	2149	2184#	2197	2244#	2255	2372	2414#	2490	2497#
	2550	2556	2616#	2640	2649#	2717	2744	2812#	2833	2863	2944#	3062	3078#
	3094	3191	3228#	3238	3252#	3305	3348	3408#	3459	3681#	3698	3700#	3732
BNSK2 = 000110	3819#	3826	3860#	3956	4161#	4173	4177#	4210	4298#	4344			
	1309#	1314	1380#	1385	1584#	1589	1648#	1657	2068#	2075	2282#	2355	2370
	2424#	2462	2467	2472#	2484	2503#	2507	2513#	2548	2836#	2846	2861	2952#
	2996	3054	3100#	3152	3189	3261#	3287	3303	3708#	3730	3868#	3910	3954
BNSK3 = 000110	4186#	4208	4304#	4330									
	2298#	2316	2349	2969#	2974	2984#	2989	3002#	3033	3052	3140#	3145	3165#
	3187	3275#	3280	3915#	3934	3952							
BNSK4 = 000110	2336#	2347	3022#	3027									
BSAVLE = 177777	601#	1670#	1671#	1864#	1865	3459#	3460#						
BSAV2 = 052752 G	4476#												
BSAV3 = 052766 G	4476#												
BSAV4 = 053004 G	4476#												
BSAV5 = 053024 G	4476#												
BSSKO = 050003	1670#	1671	1864#	1865	3459#	3460							
BTAGLE = 177777	601#	1016#	1019#	1154#	1157#	1284#	1293#	1294#	1306#	1311#	1314#	1316#	1318#
	1327#	1330#	1354#	1359#	1366#	1377#	1382#	1385#	1389#	1391#	1401#	1408#	1500#
	1505#	1513#	1522#	1523#	1526#	1556#	1559#	1581#	1585#	1589#	1595#	1626#	1630#
	1649#	1657#	1670#	1737#	1745#	1756#	1764#	1780#	1788#	1846#	1850#	1864#	2036#
	2050#	2058#	2069#	2075#	2086#	2118#	2129#	2138#	2149#	2186#	2197#	2246#	2255#
	2256#	2284#	2300#	2316#	2317#	2338#	2347#	2349#	2356#	2357#	2370#	2372#	2415#
	2425#	2463#	2464#	2467#	2474#	2484#	2490#	2499#	2504#	2507#	2515#	2548#	2551#
	2552#	2556#	2617#	2640#	2651#	2718#	2719#	2744#	2805#	2814#	2834#	2835#	2838#
	2847#	2848#	2861#	2863#	2871#	2872#	2877#	2915#	2933#	2939#	2948#	2954#	2971#
	2974#	2986#	2989#	2997#	2998#	3007#	3024#	3027#	3034#	3035#	3052#	3054#	3062#
	3068#	3069#	3080#	3095#	3096#	3102#	3142#	3145#	3153#	3154#	3167#	3187#	3189#
	3191#	3193#	3229#	3238#	3254#	3263#	3277#	3280#	3288#	3289#	3303#	3306#	3307#
	3348#	3409#	3411#	3459#	3543#	3564#	3579#	3595#	3603#	3614#	3624#	3630#	3648#
	3664#	3670#	3682#	3698#	3702#	3709#	3730#	3732#	3737#	3750#	3766#	3820#	3826#
	3862#	3870#	3910#	3911#	3917#	3934#	3935#	3952#	3954#	3956#	4022#	4038#	4049#
	4053#	4054#	4057#	4060#	4063#	4130#	4141#	4149#	4162#	4173#	4179#	4187#	4208#
	4210#	4215#	4237#	4253#	4299#	4306#	4330#	4344#					
BTAGNU = 050004	601#	1015	1016#	1153	1154#	1283	1284#	1292	1294#	1305	1306#	1310	1311#
	1326	1327#	1353	1354#	1365	1366#	1376	1377#	1381	1382#	1400	1401#	1475#
	1476#	1499	1500#	1512	1513#	1521	1523#	1555	1556#	1577#	1578#	1580	1581#
	1584	1585#	1625	1626#	1627	1629	1630#	1648	1649#	1691#	1692#	1736	1737#
	1755	1756#	1779	1780#	1812#	1813#	1845	1846#	1847	1849	1850#	1886#	1887#
	2020#	2021#	2035	2036#	2057	2058#	2068	2069#	2117	2118#	2137	2138#	2185
	2186#	2232#	2233#	2245	2246#	2256#	2283	2284#	2299	2300#	2317#	2337	2338#
	2355	2357#	2394#	2395#	2414	2415#	2424	2425#	2462	2464#	2473	2474#	2498
	2499#	2503	2504#	2514	2515#	2550	2552#	2583#	2584#	2616	2617#	2641	2644
	2645#	2650	2651#	2717	2719#	2804	2805#	2813	2814#	2833	2835#	2837	2838#
	2846	2848#	2870	2872#	2914	2915#	2938	2939#	2945	2947	2948#	2953	2954#
	2970	2971#	2985	2986#	2996	2998#	3003	3005	3006	3007#	3023	3024#	3033
	3035#	3067	3069#	3079	3080#	3094	3096#	3101	3102#	3141	3142#	3152	3154#
	3166	3167#	3218#	3219#	3228	3229#	3239	3242	3243#	3253	3254#	3262	3263#
	3276	3277#	3287	3289#	3305	3307#	3365#	3366#	3388#	3389#	3408	3409#	3410
	3411#	3542	3543#	3578	3579#	3602	3603#	3623	3624#	3647	3648#	3669	3670#
	3681	3682#	3701	3702#	3708	3709#	3749	3750#	3784#	3785#	3819	3820#	3843#
	3844#	3861	3862#	3869	3870#	3911#	3916	3917#	3935#	4021	4022#	4048	4049#

STEMP 000300

4052	4054#	4059	4060#	4129	4130#	4148	4149#	4161	4162#	4178	4179#	4186
4187#	4236	4237#	4276#	4277#	4298	4299#	4305	4306#				
1012#	1013#	1017#	1018#	1019#	1021#	1022#	1075#	1076#	1150#	1151#	1155#	1156#
1157#	1159#	1160#	1288#	1289#	1290#	1291#	1292#	1293#	1307#	1308#	1312#	1313#
1314#	1316#	1318#	1328#	1329#	1330#	1336#	1337#	1338#	1339#	1343#	1344#	1345#
1346#	1347#	1348#	1349#	1350#	1351#	1352#	1355#	1356#	1357#	1358#	1359#	1360
1367#	1368#	1378#	1379#	1383#	1384#	1385#	1387#	1388#	1389#	1391#	1392	1398#
1399#	1402#	1403#	1404#	1405#	1406#	1407#	1408#	1409	1440#	1441#	1442#	1443#
1449#	1450#	1487#	1488#	1493#	1497#	1501#	1502#	1503#	1504#	1505#	1514#	1515#
1516#	1517#	1521#	1522#	1524#	1525#	1526#	1532#	1533#	1534#	1535#	1536#	1537#
1541#	1542#	1543#	1544#	1549#	1553#	1557#	1558#	1559#	1561#	1589#	1591#	1592#
1593#	1594#	1595#	1600#	1601#	1602#	1604#	1605#	1606#	1608#	1609#	1610#	1611#
1657#	1670#	1671#	1676#	1704#	1705#	1706#	1707#	1711#	1713#	1714#	1715#	1745#
1764#	1769#	1770#	1788#	1793#	1823#	1825#	1829#	1830#	1834#	1836#	1840#	1841#
1854#	1855#	1856#	1857#	1858#	1859#	1860#	1861#	1862#	1863#	1864#	1865#	1870#
1895#	1897#	1901#	1902#	1906#	1907#	1908#	1909#	1910#	1911#	1912#	1913#	1917#
1918#	1919#	1920#	1924#	1925#	1929#	1930#	1931#	1932#	1936#	1938#	1939#	1940#
1944#	1946#	1947#	1948#	1963#	1964#	1968#	1969#	1970#	1971#	1986#	1988#	1989#
1990#	1991#	1992#	1993#	1994#	1998#	2000#	2001#	2003#	2008#	2050#	2059#	2060#
2061#	2062#	2075#	2077#	2079#	2080#	2081#	2086#	2103#	2105#	2106#	2107#	2108#
2109#	2110#	2111#	2129#	2149#	2154#	2155#	2156#	2157#	2169#	2171#	2172#	2173#
2174#	2175#	2176#	2177#	2178#	2179#	2197#	2202#	2203#	2204#	2206#	2207#	2208#
2211#	2255#	2268#	2270#	2271#	2272#	2273#	2274#	2275#	2276#	2277#	2278#	2288#
2290#	2291#	2292#	2293#	2294#	2301#	2302#	2303#	2304#	2316#	2318#	2320#	2321#
2322#	2323#	2324#	2325#	2327#	2328#	2329#	2330#	2331#	2347#	2349#	2355#	2356#
2358#	2359#	2360#	2361#	2370#	2372#	2377#	2405#	2406#	2407#	2408#	2409#	2410#
2433#	2434#	2435#	2436#	2437#	2438#	2439#	2440#	2441#	2442#	2443#	2444#	2457#
2458#	2462#	2463#	2465#	2466#	2467#	2480#	2481#	2482#	2483#	2484#	2490#	2491
2507#	2508	2524#	2525#	2526#	2527#	2528#	2529#	2530#	2531#	2532#	2533#	2534#
2535#	2548#	2550#	2551#	2556#	2558#	2596#	2597#	2602#	2603#	2604#	2605#	2606#
2607#	2611#	2612#	2621#	2622#	2629#	2633#	2634#	2635#	2640#	2643	2655#	2657#
2658#	2659#	2660#	2661#	2662#	2664#	2675#	2676#	2677#	2679#	2693#	2695#	2696#
2698#	2702#	2703#	2704#	2705#	2709#	2710#	2711#	2712#	2717#	2718#	2722#	2723#
2724#	2725#	2726#	2727#	2728#	2729#	2730#	2731#	2744#	2751#	2752#	2753#	2818#
2819#	2820#	2821#	2822#	2823#	2824#	2825#	2826#	2827#	2828#	2829#	2833#	2834#
2842#	2843#	2844#	2845#	2846#	2847#	2852#	2854#	2855#	2856#	2859#	2860#	2861#
2863#	2870#	2871#	2873#	2874#	2875#	2876#	2877#	2908#	2909#	2916#	2917#	2921#
2923#	2924#	2925#	2926#	2927#	2928#	2929#	2933#	2935#	2936#	2959#	2960#	2961#
2962#	2963#	2964#	2965#	2966#	2967#	2968#	2972#	2973#	2974#	2976#	2977#	2978#
2979#	2980#	2981#	2982#	2983#	2987#	2988#	2989#	2994#	2995#	2996#	2997#	3012#
3013#	3014#	3015#	3016#	3017#	3018#	3019#	3020#	3021#	3025#	3026#	3027#	3029#
3030#	3031#	3032#	3033#	3034#	3040#	3041#	3045#	3047#	3048#	3049#	3050#	3051#
3052#	3054#	3060#	3061#	3062#	3067#	3068#	3073#	3074#	3086#	3087#	3088#	3089#
3090#	3091#	3092#	3093#	3094#	3095#	3109#	3110#	3111#	3112#	3113#	3114#	3115#
3116#	3117#	3118#	3125#	3126#	3127#	3128#	3133#	3134#	3138#	3139#	3143#	3144#
3145#	3147#	3148#	3152#	3153#	3159#	3160#	3161#	3162#	3168#	3169#	3173#	3175#
3176#	3177#	3178#	3179#	3185#	3186#	3187#	3189#	3191#	3193#	3223#	3224#	3233#
3234#	3238#	3241	3247#	3248#	3259#	3260#	3267#	3268#	3269#	3270#	3271#	3272#
3273#	3274#	3278#	3279#	3280#	3285#	3286#	3287#	3288#	3298#	3299#	3303#	3305#
3306#	3311#	3312#	3315#	3317#	3320#	3321#	3322#	3323#	3324#	3325#	3326#	3327#
3328#	3329#	3348#	3350#	3367#	3402#	3403#	3404#	3405#	3406#	3407#	3412#	3413#
3414#	3415#	3416#	3417#	3418#	3422#	3423#	3424#	3425#	3426#	3427#	3428#	3429#
3430#	3431#	3432#	3433#	3434#	3435#	3438#	3439#	3440#	3441#	3442#	3443#	3444#
3445#	3449#	3450#	3451#	3452#	3453#	3454#	3458#	3459#	3460#	3462#	3463#	3470#
3524#	3525#	3529#	3531#	3532#	3533#	3564#	3595#	3614#	3619#	3620#	3621#	3622#
3628#	3629#	3630#	3631	3636#	3638#	3639#	3640#	3664#	3698#	3730#	3732#	3737#



	3738	3766#	3790#	3791#	3792#	3793#	3794#	3795#	3810#	3811#	3812#	3813#	3821#
	3822#	3826#	3828#	3854#	3856#	3866#	3867#	3879#	3880#	3881#	3882#	3883#	3884#
	3885#	3886#	3887#	3888#	3889#	3890#	3891#	3892#	3893#	3894#	3895#	3896#	3910#
	3934#	3952#	3954#	3956#	3964#	4004#	4005#	4006#	4007#	4008#	4009#	4013#	4014#
	4038#	4045#	4046#	4050#	4051#	4052#	4053#	4055#	4056#	4057#	4061#	4062#	4063#
	4064	4069#	4071#	4072#	4073#	4077#	4079#	4080#	4081#	4087#	4088#	4092#	4093#
	4119#	4120#	4121#	4122#	4141#	4173#	4208#	4210#	4215#	4216	4228#	4229#	4253#
	4289#	4290#	4291#	4292#	4293#	4294#	4307#	4308#	4309#	4310#	4330#	4335#	4336#
\$TSK0 = 050002	4337#	4338#	4339#	4340#	4344#	4345	4353#						
	1016#	1019	1154#	1157	1284#	1293	1294#	1318	1327#	1330	1354#	1359	1366#
	1391	1401#	1408	1500#	1505	1513#	1522	1523#	1526	1556#	1559	1581#	1595
	1626#	1670	1737#	1745	1756#	1764	1780#	1788	1846#	1864	2036#	2050	2058#
	2086	2118#	2129	2138#	2149	2186#	2197	2246#	2255	2256#	2372	2415#	2490
	2499#	2551	2552#	2556	2617#	2640	2651#	2718	2719#	2744	2805#	2871	2872#
	2877	2915#	2933	2939#	3068	3069#	3193	3229#	3238	3254#	3306	3307#	3348
	3409#	3459	3543#	3564	3579#	3595	3603#	3614	3624#	3630	3648#	3664	3670#
	3737	3750#	3766	3820#	3826	3862#	3956	4022#	4038	4049#	4053	4054#	4057
\$TSK1 = 050003	4060#	4063	4130#	4141	4149#	4215	4237#	4253	4299#	4344			
	1306#	1316	1377#	1389	1585#	1589	1630#	1670	1850#	1864	2069#	2075	2284#
	2356	2357#	2370	2425#	2463	2464#	2467	2474#	2484	2504#	2507	2515#	2548
	2814#	2834	2835#	2863	2948#	3062	3080#	3095	3096#	3191	3263#	3288	3289#
	3303	3411#	3459	3682#	3698	3702#	3732	3870#	3910	3911#	3954	4162#	4173
	4179#	4210	4306#	4330									
\$TSK2 = 050017	1311#	1314	1382#	1385	1649#	1657	2300#	2316	2317#	2349	2838#	2847	2848#
	2861	2954#	2997	2998#	3054	3102#	3153	3154#	3189	3277#	3280	3709#	3730
	3917#	3934	3935#	3952	4187#	4208							
\$TSK3 = 050035	2338#	2347	2971#	2974	2986#	2989	3007#	3034	3035#	3052	3142#	3145	3167#
	3187												
\$TSK4 = 050025	3024#	3027											
\$BARGC = 000000	601#	1476#	1578#	1692#	1813#	1887#	2021#	2233#	2395#	2584#	3219#	3366#	3389#
\$BBYTE = 000402	3785#	3844#	4277#										
	601#	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	1626#	1628#	1846#	1848#
	2056#	2116#	2136#	2184#	2244#	2282#	2298#	2336#	2472#	2497#	2513#	2649#	2803#
	2812#	2836#	2913#	2937#	2944#	2946#	2952#	2969#	2984#	3002#	3004#	3022#	3078#
	3100#	3140#	3165#	3252#	3261#	3275#	3409#	3601#	3700#	3860#	3868#	3915#	4047#
	4177#	4304#											
\$BCASE = 000000	601#												
\$BDST = 000003	601#	1494#	1550#	2630#	3418#	3445#	3454#						
\$BELOC = 000402	601#	1014#	1019#	1152#	1157#	1283#	1292	1305#	1309#	1314#	1316#	1318#	1326#
	1330#	1376#	1380#	1385#	1389#	1498#	1505#	1511#	1521	1526#	1554#	1559#	1579#
	1584#	1588#	1589#	1595#	1648#	1656#	1657#	1736#	1744#	1745#	1755#	1763#	1764#
	1779#	1787#	1788#	2035#	2049#	2050#	2056#	2068#	2074#	2075#	2085#	2086#	2116#
	2128#	2129#	2136#	2148#	2149#	2184#	2196#	2197#	2244#	2254#	2255	2282#	2298#
	2312#	2316	2336#	2346#	2347#	2349#	2355	2369#	2370#	2372#	2424#	2462	2467#
	2472#	2484#	2497#	2513#	2547#	2548#	2550	2555#	2556#	2649#	2717	2743#	2744#
	2803#	2812#	2833	2836#	2846	2861#	2863#	2870	2877#	2913#	2933#	2937#	2944#
	2952#	2969#	2974#	2984#	2989#	2996	3002#	3022#	3027#	3033	3052#	3054#	3062#
	3067	3078#	3094	3100#	3140#	3145#	3152	3165#	3187#	3189#	3191#	3193#	3252#
	3261#	3275#	3280#	3287	3302#	3303#	3305	3347#	3348#	3542#	3564#	3578#	3595#
	3601#	3614#	3647#	3664#	3681#	3698#	3700#	3708#	3730#	3732#	3749#	3766#	3819#
	3825#	3826#	3860#	3868#	3909#	3910	3915#	3933#	3934	3951#	3952#	3954#	3956#
	4021#	4038#	4047#	4052	4057#	4129#	4141#	4161#	4173#	4177#	4186#	4208#	4210#
	4236#	4253#	4304#	4329#	4330#								
\$BERFL = 000000	601#												
\$FLAG = 000001	601#	1014#	1016#	1019#	1152#	1154#	1157#	1283#	1305#	1309#	1311#	1314#	1316#
	1318#	1326#	1330#	1376#	1380#	1382#	1385#	1389#	1498#	1500#	1505#	1511#	1513#

	1526#	1554#	1556#	1559#	1579#	1581#	1584#	1589#	1595#	1625#	1626#	1628#	1630#
	1648#	1657#	1736#	1745#	1755#	1764#	1779#	1788#	1845#	1846#	1848#	1850#	2035#
	2050#	2056#	2058#	2068#	2075#	2086#	2116#	2118#	2129#	2136#	2138#	2149#	2184#
	2186#	2197#	2244#	2246#	2282#	2284#	2298#	2300#	2336#	2338#	2347#	2349#	2370#
	2372#	2424#	2467#	2472#	2474#	2484#	2497#	2499#	2513#	2515#	2548#	2556#	2649#
	2651#	2744#	2803#	2805#	2812#	2814#	2836#	2838#	2861#	2863#	2877#	2913#	2915#
	2933#	2937#	2939#	2944#	2946#	2948#	2952#	2954#	2969#	2971#	2974#	2984#	2986#
	2989#	3002#	3004#	3007#	3022#	3024#	3027#	3052#	3054#	3062#	3078#	3080#	3100#
	3102#	3140#	3142#	3145#	3165#	3167#	3187#	3189#	3191#	3193#	3252#	3254#	3261#
	3263#	3275#	3277#	3280#	3303#	3348#	3408#	3409#	3411#	3542#	3564#	3578#	3595#
	3601#	3603#	3614#	3647#	3664#	3681#	3698#	3700#	3702#	3708#	3730#	3732#	3749#
	3766#	3819#	3826#	3860#	3862#	3868#	3870#	3915#	3917#	3952#	3954#	3956#	4021#
	4038#	4047#	4049#	4057#	4129#	4141#	4161#	4173#	4177#	4179#	4186#	4208#	4210#
	4236#	4253#	4304#	4306#	4330#								
\$\$FROM= 000000	601#	1582#	1615#	1634#	1643#	1662#	1720#	1731#	1750#	1771#	1952#	1975#	2030#
	2063#	2092#	2158#	2260#	2419#	3537#	3549#	3569#	3580#	3641#	3649#	3674#	3683#
	3703#	3710#	3715#	3743#	3751#	3799#	3814#	4015#	4023#	4097#	4108#	4123#	4131#
\$\$LOC = 021360	4153#	4163#	4180#	4188#	4193#	4221#	4230#	4238#	4243#	4311#			
	601#	1015#	1016	1153#	1154	1283#	1284	1305#	1306	1310#	1311	1326#	1327
	1360#	1361	1376#	1377	1381#	1382	1392#	1393	1409#	1410	1499#	1500	1512#
	1513	1555#	1556	1580#	1581	1584#	1585	1627#	1628	1629#	1630	1648#	1649
	1736#	1737	1755#	1756	1779#	1780	1847#	1848	1849#	1850	2035#	2036	2057#
	2058	2068#	2069	2117#	2118	2137#	2138	2185#	2186	2245#	2246	2283#	2284
	2299#	2300	2337#	2338	2424#	2425	2473#	2474	2491#	2492	2498#	2499	2508#
	2509	2514#	2515	2643#	2644	2650#	2651	2804#	2805	2813#	2814	2837#	2838
	2914#	2915	2938#	2939	2945#	2946	2947#	2948	2953#	2954	2970#	2971	2985#
	2986	3005#	3006	3023#	3024	3079#	3080	3101#	3102	3141#	3142	3166#	3167
	3241#	3242	3253#	3254	3262#	3263	3276#	3277	3410#	3411	3542#	3543	3578#
	3579	3602#	3603	3631#	3632	3647#	3648	3681#	3682	3701#	3702	3708#	3709
	3738#	3739	3749#	3750	3819#	3820	3861#	3862	3869#	3870	3916#	3917	4021#
	4022	4048#	4049	4064#	4065	4129#	4130	4161#	4162	4178#	4179	4186#	4187
	4216#	4217	4236#	4237	4305#	4306	4345#	4346					
\$\$LOCN= 000000	601#	1476#	1478#	1479#	1482	1563	1578#	1679	1692#	1693#	1694#	1697	1796
	1813#	1814#	1815#	1818	1872	1887#	2010	2021#	2214	2233#	2234#	2235#	2238
	2380	2395#	2396#	2397#	2400	2561	2584#	2585#	2586#	2589	2756	3219#	3353
\$\$REG = 177777	3366#	3369	3389#	3390#	3391#	3394	3472	3785#	3831	3844#	3967	4277#	4356
	601#	1478#	1563	1693#	1796	1814#	1872	2234#	2380	2396#	2561	2585#	2756
\$\$RETU= 000001	3390#	3472											
	601#	1476#	1562	1578#	1586#	1654#	1677	1692#	1742#	1761#	1785#	1794	1813#
	1871	1887#	2009	2021#	2047#	2072#	2126#	2146#	2194#	2212	2233#	2252#	2310#
	2344#	2367#	2378	2395#	2545#	2553#	2559	2584#	2741#	2754	3219#	3300#	3345#
	3351	3366#	3368	3389#	3471	3785#	3823#	3829	3844#	3907#	3931#	3949#	3965
	4277#	4327#	4354										
\$\$RTN1 050000	601#	1476#	1561	1578#	1676	1692#	1793	1813#	1870	1887#	2008	2021#	2084
	2211	2233#	2377	2395#	2558	2584#	2753	3219#	3350	3366#	3367	3389#	3470
	3785#	3828	3844#	3964	4277#	4353							
\$\$RTN2- 050001	601#	1476#	1562	1578#	1587	1655	1678	1692#	1743	1762	1786	1795	1813#
	1871	1887#	2009	2021#	2048	2073	2127	2147	2195	2213	2233#	2253	2311
	2345	2368	2379	2395#	2546	2554	2560	2584#	2742	2755	3219#	3301	3346
	3352	3366#	3368	3389#	3471	3785#	3824	3830	3844#	3908	3932	3950	3966
	4277#	4328	4355										
\$\$SRC = 000064	601#	1494#	1550#	2630#	3418#	3445#	3454#						
\$\$TGSV= 000000	601#												
\$\$TGS1= 000000	601#												
\$\$TGS2= 000000	601#												
\$\$TO = 000001	601#	1582#	1583	1615#	1616	1617	1634#	1635	1639	1643#	1644	1662#	1663





ABORTW	2856	3182													
ALLOCA	1479	1694	1815	2235	2397	2586	3391								
BGNAU	1254														
BGNCLM	1433														
BGNDU	1259														
BGNHRD	4378														
BGNHW	1220														
BGNINI	1267														
BGNMOD	606	651	1216	1244	1457	3509	4374								
BGNMSG	903	921	926	948	962	986	1000	1004	1008	1052	1070	1126	1142	1176	1198
	1203														
BGNRPT	1249														
BGNSEG	4222														
BGNSFT	4425														
BGNSRV	2792	2892													
BGNSW	1228														
BGNTST	3519	3999													
BREAK	2504														
CALL	1581	1614	1633	1642	1661	1719	1730	1749	1770	1951	1974	2029	2062	2091	2157
	2259	2418	3536	3548	3568	3579	3640	3648	3673	3682	3702	3709	3714	3742	3750
	3798	3813	4014	4022	4096	4107	4122	4130	4152	4162	4179	4187	4192	4220	4229
	4237	4242	4310												
CKLOOP	3558	3589	3608	3658	3692	3724	3760	4032	4142	4174	4202	4254			
DEALLO	1561#	1563	1676#	1793#	1796	1870#	1872	2008#	2211#	2377#	2380	2558#	2561	2753#	2756
	3350#	3367#	3470#	3472	3828#	3964#	4353#								
DECLAR	1477	1692	1813	2233	2395	2584	3389								
DEVREG	734														
DEVTYP	730														
DISPAT	1239														
ELSE	1291	1520	2254	2315	2354	2461	2549	2716	2832	2845	2869	2995	3032	3066	3093
	3151	3286	3304	3909	3933	4051									
ENDAU	1255														
ENDCLM	1453														
ENDDO	1669	1863	3458												
ENDDU	1260														
ENDHRD	4402														
ENDHW	1226														
ENDIF	1018	1156	1313	1315	1317	1329	1384	1388	1504	1525	1558	1588	1594	1656	1744
	1763	1787	2049	2074	2085	2128	2148	2196	2346	2348	2369	2371	2466	2483	2547
	2555	2743	2860	2862	2876	2932	2973	2988	3026	3051	3053	3061	3144	3186	3188
	3190	3192	3279	3302	3347	3563	3594	3613	3663	3697	3729	3731	3765	3825	3951
	3953	3955	4037	4056	4140	4172	4207	4209	4252	4329					
ENDINI	1415														
ENDMOD	650	1215	1243	1456	3508	4373	4470								
ENDMSG	911	922	935	958	973	995	1001	1005	1035	1060	1106	1133	1167	1185	1199
	1204														
ENDRPT	1250														
ENDRTN	1560	1675	1792	1869	2007	2210	2376	2557	2752	3349	3366	3469	3827	3963	4352
ENDSEG	4256														
ENDSFT	4443														
ENDSRV	2881	3199													
ENDSW	1233														
ENDTST	3770	4262													
EQUALS	655														
ERRDF	2474														
ERRHRD	1649	1737	1756	1780	2036	2118	2138	2186	2246	2304	2338	2361	2451	2539	2735

	3292	3336	3553	3584	3603	3653	3687	3719	3755	3896	3920	3938	4027	4135	4167
	4197	4247													
ERRSOF	4315														
EXIT	3560	3591	3610	3660	3694	3726	3762	4034	4204						
GPHARD	1368														
GPRMA	4380	4385													
GPRMO	4390	4396	4427	4437											
GPRML	4433														
HEADER	607														
IF	1013	1151	1308	1379	1497	1510	1553	1578	2055	2115	2183	2243	2281	2297	2335
	2471	2496	2512	2648	2802	2811	2835	2912	2936	2968	2983	3021	3077	3099	3139
	3164	3251	3260	3274	3600	3699	4046	4176							
IFB	2135	2943	2951	3001	3859	3867	3914	4303							
IF.ERR	1282	1325	1583	1647	1735	1754	1778	2034	2067	2423	3541	3577	3646	3680	3707
	3748	3818	4020	4128	4150	4185	4235								
IF.NO.	1304	1375													
LASTAD	4471														
LET	1011	1016	1020	1074	1149	1154	1158	1287	1289	1306	1311	1327	1335	1337	1342
	1344	1346	1348	1350	1354	1356	1366	1377	1382	1386	1397	1401	1403	1405	1439
	1441	1448	1486	1492	1500	1502	1513	1515	1523	1531	1533	1535	1540	1542	1548
	1556	1586#	1590	1592	1599	1601	1604	1607	1609	1654#	1703	1705	1710	1713	1742#
	1761#	1768	1785#	1822	1828	1833	1839	1853	1855	1857	1859	1861	1894	1900	1905
	1907	1909	1911	1916	1918	1923	1928	1930	1935	1938	1943	1946	1962	1967	1969
	1985	1988	1990	1992	1997	2000	2047#	2058	2060	2072#	2076	2079	2084#	2102	2105
	2107	2109	2126#	2146#	2153	2155	2168	2171	2173	2175	2177	2194#	2201	2203	2206
	2252#	2267	2270	2272	2274	2276	2287	2290	2292	2300	2302	2310#	2317	2320	2322
	2324	2327	2329	2344#	2357	2359	2367#	2404	2406	2408	2437	2434	2436	2438	2440
	2442	2456	2464	2479	2481	2523	2525	2527	2529	2531	2533	2545#	2553#	2595	2601
	2603	2605	2610	2620	2628	2633	2654	2657	2659	2661	2674	2676	2692	2695	2701
	2703	2708	2710	2721	2723	2725	2727	2729	2741#	2750	2817	2819	2821	2823	2825
	2827	2841	2843	2851	2854	2858	2872	2874	2907	2915	2920	2923	2925	2927	2934
	2958	2960	2962	2964	2966	2971	2975	2977	2979	2981	2986	2993	3011	3013	3015
	3017	3019	3024	3028	3030	3039	3044	3047	3049	3059	3072	3085	3087	3089	3091
	3108	3110	3112	3114	3116	3124	3126	3132	3137	3142	3146	3158	3160	3167	3172
	3175	3177	3184	3222	3232	3246	3258	3266	3268	3270	3272	3277	3284	3297	3300#
	3310	3314	3319	3321	3323	3325	3327	3345#	3401	3403	3405	3411	3413	3415	3417
	3422	3424	3426	3428	3430	3432	3434	3438	3440	3442	3444	3449	3451	3453	3461
	3523	3528	3531	3618	3620	3627	3635	3638	3789	3791	3793	3809	3811	3820	3823#
	3853	3865	3878	3880	3882	3884	3886	3898	3890	3892	3894	3907#	3931#	3949#	4003
	4005	4007	4012	4044	4049	4054	4060	4068	4071	4076	4079	4086	4091	4118	4120
	4227	4288	4290	4292	4306	4308	4327#	4334	4336	4338					
LOCAL	1478	1693	1814	2234	2396	2585	3390								
MSBYTE	608#	613	614	615											
MSCHEC	3561#	3592#	3611#	3661#	3695#	3727#	3763#	4035#	4205#						
MSCKID	630#														
MSOUN	905#	928#	951#	966#	988#	1023#	1030#	1054#	1077#	1084#	1091#	1098#	1128#	1144#	1161#
	1178#														
MSDATA	608#	616	617	618	619	620	621	622	623	624	625	626	627	628	629
	630#	631	632	633	635	636	637	639	640	641	642	643	644	645	646
	647	648	649	731#											
MSDECR	651#	912#	923#	936#	959#	974#	996#	1002#	1006#	1036#	1061#	1107#	1134#	1168#	1186#
	1200#	1205#	1216#	1227#	1234#	1244#	1251#	1256#	1261#	1416#	1454#	1457#	2882#	3200#	3509#
	3771#	4257#	4263#	4374#	4403#	4444#	4471#								
MSDEFA	4381#	4386#	4391#	4397#	4428#	4434#	4438#								
MSINDE	651#	912#	923#	936#	959#	974#	996#	1002#	1006#	1036#	1061#	1107#	1134#	1168#	1186#
	1200#	1205#	1216#	1227#	1234#	1244#	1251#	1256#	1261#	1416#	1454#	1457#	2882#	3200#	3509#

MSERR1	3771#	4257#	4263#	4374#	4403#	4444#	4471#	2187#	2247#	2305#	2339#	2362#	2452#	2475#	2540#
	1650#	1738#	1757#	1781#	2037#	2119#	2139#	368#	3720#	3756#	3897#	3921#	3939#	4028#	4136#
	2736#	3293#	3337#	3554#	3585#	3604#	3654#								
	4168#	4198#	4248#	4316#											
MSKCP	4381#	4386#	4391#	4397#	4428#	4438#									
MSK11	3561#	3562#	3592#	3593#	3611#	3617#	3661#	3662#	3695#	3696#	3727#	3728#	3763#	3764#	4035#
	4036#	4205#	4206#												
MSKSE	3561#	3592#	3611#	3661#	3695#	3727#	3763#	4035#	4205#						
MSKXJ	3561#	3592#	3611#	3661#	3695#	3727#	3763#	4035#	4205#						
MSGEM	607#	608#	616#	617#	618#	619#	620#	621#	622#	623#	624#	625#	626#	627#	628#
	629#	630#	631#	632#	633#	635#	636#	637#	639#	640#	641#	642#	643#	644#	645#
	646#	647#	648#	649#	652#	731#	736#	738#	904#	912#	922#	923#	927#	936#	949#
	959#	963#	974#	987#	996#	1001#	1002#	1005#	1006#	1009#	1036#	1053#	1061#	1071#	1107#
	1127#	1134#	1143#	1168#	1177#	1186#	1199#	1200#	1204#	1205#	1217#	1222#	1227#	1230#	1234#
	1241#	1245#	1250#	1251#	1255#	1256#	1260#	1261#	1268#	1416#	1434#	1454#	1458#	2793#	2882#
MSGETS	2893#	3200#	3510#	3520#	3771#	4000#	4257#	4263#	4375#	4380#	4404#	4427#	4445#	4473#	
	651#	912#	923#	936#	959#	974#	996#	1002#	1006#	1036#	1061#	1107#	1134#	1168#	1186#
	1200#	1205#	1216#	1227#	1234#	1244#	1251#	1256#	1261#	1416#	1454#	1457#	2882#	3200#	3509#
MSGETT	3771#	4257#	4263#	4374#	4403#	4444#	4471#								
MSGNGB	3561#	3592#	3611#	3661#	3695#	3727#	3763#	4035#	4205#						
	607#	608#	616#	617#	618#	619#	620#	621#	622#	623#	624#	625#	626#	627#	628#
	629#	630#	631#	632#	633#	635#	636#	637#	639#	640#	641#	642#	643#	644#	645#
	646#	647#	648#	649#	652#	731#	735#	736#	738#	904#	922#	927#	949#	963#	987#
	1001#	1005#	1009#	1053#	1071#	1127#	1143#	1177#	1199#	1204#	1217#	1221#	1222#	1229#	1230#
	1240#	1241#	1245#	1250#	1255#	1260#	1268#	1434#	1458#	2793#	2893#	3510#	4375#	4379#	4380#
MSGNIN	4426#	4427#	4472#	4473#											
	608#	609#	610#	611#	612#	613#	614#	615#	616#	617#	618#	619#	620#	621#	622#
	623#	624#	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#	731#	733#	735#
	736#	737#	738#	905#	906#	907#	908#	909#	910#	913#	924#	928#	929#	930#	931#
	932#	933#	934#	937#	951#	952#	953#	954#	955#	956#	957#	960#	966#	967#	968#
	969#	970#	971#	972#	975#	988#	989#	990#	991#	992#	993#	994#	997#	1003#	1007#
	1023#	1024#	1025#	1026#	1027#	1028#	1030#	1031#	1032#	1033#	1034#	1037#	1054#	1055#	1056#
	1057#	1058#	1059#	1062#	1077#	1078#	1079#	1080#	1081#	1082#	1084#	1085#	1086#	1087#	1088#
	1089#	1091#	1092#	1093#	1094#	1095#	1096#	1098#	1099#	1100#	1101#	1102#	1103#	1108#	1128#
	1129#	1130#	1131#	1132#	1135#	1144#	1145#	1146#	1147#	1148#	1161#	1162#	1163#	1164#	1165#
	1166#	1169#	1178#	1179#	1180#	1181#	1182#	1183#	1184#	1187#	1201#	1206#	1221#	1229#	1240#
	1241#	1242#	1252#	1257#	1262#	1277#	1278#	1302#	1303#	1323#	1324#	1369#	1370#	1371#	1417#
	1455#	1650#	1651#	1652#	1738#	1739#	1740#	1757#	1758#	1759#	1781#	1782#	1783#	2037#	2038#
	2039#	2040#	2119#	2120#	2121#	2122#	2139#	2140#	2141#	2142#	2187#	2188#	2189#	2190#	2247#
	2248#	2249#	2250#	2305#	2306#	2307#	2308#	2339#	2340#	2341#	2342#	2362#	2363#	2364#	2365#
	2452#	2453#	2454#	2455#	2475#	2476#	2477#	2478#	2505#	2540#	2541#	2542#	2543#	2623#	2624#
	2665#	2666#	2667#	2668#	2669#	2670#	2683#	2684#	2685#	2686#	2687#	2688#	2736#	2737#	2738#
	2739#	2797#	2798#	2857#	2882#	2883#	2897#	2898#	3183#	3200#	3201#	3230#	3231#	3293#	3294#
	3295#	3296#	3337#	3338#	3339#	3340#	3554#	3555#	3556#	3557#	3559#	3561#	3562#	3585#	3586#
	3587#	3588#	3590#	3592#	3593#	3604#	3605#	3606#	3607#	3609#	3611#	3612#	3625#	3626#	3654#
	3655#	3656#	3657#	3659#	3661#	3662#	3688#	3689#	3690#	3691#	3693#	3695#	3696#	3720#	3721#
	3722#	3723#	3725#	3727#	3728#	3756#	3757#	3758#	3759#	3761#	3763#	3764#	3772#	3897#	3898#
	3899#	3921#	3922#	3923#	3939#	3940#	3941#	4028#	4029#	4030#	4031#	4033#	4035#	4036#	4136#
	4137#	4138#	4139#	4143#	4168#	4169#	4170#	4171#	4175#	4198#	4199#	4200#	4201#	4203#	4205#
	4206#	4223#	4248#	4249#	4250#	4251#	4255#	4258#	4264#	4316#	4317#	4318#	4319#	4379#	4381#
	4382#	4383#	4384#	4386#	4387#	4388#	4389#	4391#	4392#	4393#	4394#	4395#	4397#	4398#	4399#
	4400#	4401#	4403#	4426#	4428#	4429#	4430#	4431#	4432#	4434#	4435#	4436#	4438#	4439#	4440#
	4441#	4442#	4444#	4472#											
MSGNLS	4257#														
MSGNTA	912#	923#	936#	959#	974#	996#	1002#	1006#	1036#	1061#	1107#	1134#	1168#	1186#	1200#



MBSTL	909#	913#	924#	933#	937#	956#	960#	971#	975#	993#	997#	1003#	1007#	1027#	1033#
	1037#	1058#	1062#	1081#	1088#	1095#	1102#	1108#	1131#	1135#	1147#	1165#	1169#	1183#	1187#
	1201#	1206#	1252#	1257#	1262#	1278#	1303#	1324#	1370#	1417#	1455#	1650#	1738#	1757#	1781#
	2037#	2119#	2139#	2187#	2247#	2305#	2339#	2362#	2452#	2475#	2505#	2540#	2624#	2669#	2687#
	2736#	2798#	2857#	2898#	3183#	3231#	3293#	3337#	3554#	3559#	3561#	3585#	3590#	3592#	3604#
	3609#	3611#	3626#	3654#	3659#	3661#	3688#	3693#	3695#	3720#	3725#	3727#	3756#	3761#	3763#
	3772#	3897#	3921#	3939#	4028#	4033#	4035#	4136#	4143#	4168#	4175#	4198#	4203#	4205#	4223#
	4248#	4255#	4258#	4264#	4316#										
MSWORD	630#	638	735#	736#	737	1240#	1241	1242	1650#	1651	1652	1738#	1739	1740	1757#
	1758	1759	1781#	1782	1783	2037#	2038	2039	2040	2119#	2120	2121	2122	2139#	2140
	2141	2142	2187#	2188	2189	2190	2247#	2248	2249	2250	2305#	2306	2307	2308	2339#
	2340	2341	2342	2362#	2363	2364	2365	2452#	2453	2454	2455	2475#	2476	2477	2478
	2540#	2541	2542	2543	2736#	2737	2738	2739	3293#	3294	3295	3296	3337#	3338	3339
	3340	3554#	3555	3556	3557	3561#	3585#	3586	3587	3588	3592#	3604#	3605	3606	3607
	3611#	3654#	3655	3656	3657	3661#	3688#	3689	3690	3691	3695#	3720#	3721	3722	3723
	3727#	3756#	3757	3758	3759	3763#	3897#	3898	3899	3921#	3922	3923	3939#	3940	3941
	4028#	4029	4030	4031	4035#	4136#	4137	4138	4139	4168#	4169	4170	4171	4198#	4199
	4200	4201	4205#	4248#	4249	4250	4251	4316#	4317	4318	4319	4381#	4386#	4391#	4397#
	4428#	4434#	4438#												
POINTE	605														
POP	1104	1106	1413	1415	1443	1445	2003	2005	2007	2044	2046	2069	2071	2081	2083
	2123	2125	2143	2145	2191	2193	2208	2210	2444	2446	2448	2535	2537	2539	2731
	2733	2735	2748	2750	2864	2866	2929	2931	3118	3120	3197	3199	3329	3331	3333
	3463	3465	3467	3469	3900	3902	3904	3906	3924	3926	3928	3930	3942	3944	3946
	3948	3957	3959	3961	3963	4081	4083	4320	4322	4324	4326	4346	4348	4350	4352
PRINTX	904	927	950	965	987	1022	1029	1053	1076	1083	1090	1097	1127	1143	1160
	1177														
PUSH	1072	1074	1271	1273	1437	1439	1887	1889	1891	2024	2026	2425	2427	2429	2515
	2517	2519	2590	2592	2719	2721	2809	2811	2899	2901	3106	3108	3312	3314	3317
	3319	3395	3397	3399	3401	3844	3846	3848	3850	4042	4044	4277	4279	4281	4283
REDEF	1276	1301	1322												
REPEAT	1352	1364	1399	2413	2502	2615	3227	3622	3668	4058	4147	4297			
RETURN	1585	1653	1741	1760	1784	2046	2071	2083	2125	2145	2193	2251	2309	2343	2366
	2544	2552	2740	3299	3344	3822	3906	3930	3948	4326					
ROUTIN	1474	1576	1690	1811	1885	2019	2231	2393	2582	3217	3364	3387	3783	3842	4275
SETPRI	2796	2896													
SETVFC	2664	2682													
STRUCT	600#														
SVC	600#														
UNTIL	1358	1390	1407	2489	2506	2639	3237	3629	3736	4062	4214	4343			
WAITUS	2622	3229	3624												
WHILE	1624	1844	3407												
XFER	3561#	3592#	3611#	3661#	3695#	3727#	3763#	4035#	4205#						
SADDON	1014#	1016#	1152#	1154#	1283#	1284#	1292#	1294#	1305#	1306#	1309#	1311#	1326#	1327#	1353#
	1354#	1365#	1366#	1376#	1377#	1380#	1382#	1400#	1401#	1475#	1476#	1498#	1500#	1511#	1513#
	1521#	1523#	1554#	1556#	1577#	1578#	1579#	1581#	1584#	1585#	1625#	1626#	1628#	1630#	1648#
	1649#	1670#	1691#	1692#	1736#	1737#	1755#	1756#	1779#	1780#	1812#	1813#	1845#	1846#	1848#
	1850#	1864#	1886#	1887#	2020#	2021#	2035#	2036#	2056#	2058#	2068#	2069#	2116#	2118#	2136#
	2138#	2184#	2186#	2232#	2233#	2244#	2246#	2255#	2256#	2282#	2284#	2298#	2300#	2316#	2317#
	2336#	2338#	2355#	2357#	2394#	2395#	2414#	2415#	2424#	2425#	2462#	2464#	2472#	2474#	2497#
	2499#	2503#	2504#	2513#	2515#	2550#	2552#	2583#	2584#	2616#	2617#	2640#	2645	2649#	2651#
	2717#	2719#	2803#	2805#	2812#	2814#	2833#	2835#	2836#	2838#	2846#	2848#	2870#	2872#	2913#
	2915#	2937#	2939#	2944#	2946#	2948#	2952#	2954#	2969#	2971#	2984#	2986#	2996#	2998#	3002#
	3004#	3007#	3022#	3024#	3033#	3035#	3067#	3069#	3078#	3080#	3094#	3096#	3100#	3102#	3140#
	3142#	3152#	3154#	3165#	3167#	3218#	3219#	3228#	3229#	3238#	3243	3252#	3254#	3261#	3263#
	3275#	3277#	3287#	3289#	3305#	3307#	3365#	3366#	3388#	3389#	3408#	3409#	3411#	3459#	3542#

	3543#	3578#	3579#	3601#	3603#	3623#	3624#	3647#	3648#	3669#	3670#	3681#	3682#	3700#	3702#
	3708#	3709#	3749#	3750#	3784#	3785#	3819#	3820#	3843#	3844#	3860#	3862#	3868#	3870#	3910#
	3911#	3915#	3917#	3934#	3935#	4021#	4022#	4047#	4049#	4052#	4054#	4059#	4060#	4129#	4130#
	4148#	4149#	4161#	4162#	4177#	4179#	4186#	4187#	4236#	4237#	4276#	4277#	4298#	4299#	4304#
	4306#														
\$AND	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	1626#	1628#	1846#	1848#	2056#	2116#	2136#
	2184#	2244#	2282#	2298#	2336#	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#	2937#	2944#
	2946#	2952#	2969#	2984#	3002#	3004#	3022#	3078#	3100#	3140#	3165#	3252#	3261#	3275#	3409#
	3601#	3700#	3860#	3868#	3915#	4047#	4177#	4304#							
\$BRANC	1015#	1153#	1283#	1292#	1305#	1310#	1326#	1360#	1376#	1381#	1392#	1409#	1499#	1512#	1521#
	1555#	1580#	1584#	1586#	1587#	1627#	1629#	1648#	1654#	1655#	1670#	1736#	1742#	1743#	1755#
	1761#	1762#	1779#	1785#	1786#	1847#	1849#	1864#	2035#	2047#	2048#	2057#	2068#	2072#	2073#
	2084#	2117#	2126#	2127#	2137#	2146#	2147#	2185#	2194#	2195#	2245#	2252#	2253#	2255#	2283#
	2299#	2310#	2311#	2316#	2337#	2344#	2345#	2355#	2367#	2368#	2424#	2462#	2473#	2491#	2498#
	2508#	2514#	2545#	2546#	2550#	2553#	2554#	2640#	2641#	2643#	2650#	2717#	2741#	2742#	2804#
	2813#	2833#	2837#	2846#	2870#	2914#	2938#	2945#	2947#	2953#	2970#	2985#	2996#	3002#	3003#
	3005#	3023#	3033#	3067#	3079#	3094#	3101#	3141#	3152#	3166#	3238#	3239#	3241#	3253#	3262#
	3276#	3287#	3300#	3301#	3305#	3345#	3346#	3410#	3459#	3542#	3578#	3602#	3631#	3647#	3681#
	3701#	3708#	3738#	3749#	3819#	3823#	3824#	3861#	3869#	3907#	3908#	3910#	3916#	3931#	3932#
	3934#	3949#	3950#	4021#	4048#	4052#	4064#	4129#	4161#	4178#	4186#	4216#	4236#	4305#	4327#
	4328#	4345#													
\$BRCOD	2640#	3002#	3238#												
\$CALL	1582#	1615#	1634#	1643#	1662#	1720#	1731#	1750#	1771#	1952#	1975#	2030#	2063#	2092#	2158#
	2260#	2419#	3537#	3549#	3569#	3580#	3641#	3649#	3674#	3683#	3703#	3710#	3715#	3743#	3751#
	3799#	3814#	4015#	4023#	4097#	4108#	4123#	4131#	4153#	4163#	4180#	4188#	4193#	4221#	4230#
	4238#	4243#	4311#												
\$CHECK	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	1625#	1626#	1845#	1846#	2056#	2116#	2136#
	2184#	2244#	2282#	2298#	2336#	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#	2937#	2944#
	2952#	2969#	2984#	3002#	3022#	3078#	3100#	3140#	3165#	3252#	3261#	3275#	3408#	3409#	3601#
	3700#	3860#	3868#	3915#	4047#	4177#	4304#								
\$CKOP1	1012#	1017#	1075#	1150#	1155#	1288#	1312#	1328#	1336#	1338#	1343#	1345#	1347#	1349#	1351#
	1355#	1383#	1387#	1398#	1402#	1440#	1487#	1524#	1532#	1536#	1591#	1593#	1600#	1605#	1608#
	1610#	1704#	1706#	1714#	1769#	1829#	1840#	1854#	1901#	1906#	1908#	1910#	1912#	1924#	1931#
	1939#	1947#	1963#	1968#	1970#	1989#	1991#	1993#	2059#	2061#	2080#	2106#	2108#	2110#	2174#
	2176#	2178#	2207#	2271#	2273#	2277#	2291#	2301#	2303#	2321#	2328#	2358#	2360#	2405#	2407#
	2409#	2433#	2435#	2437#	2439#	2441#	2443#	2465#	2482#	2524#	2526#	2528#	2530#	2532#	2534#
	2596#	2611#	2658#	2675#	2702#	2704#	2722#	2724#	2726#	2728#	2730#	2618#	2824#	2908#	2916#
	2928#	2959#	2961#	2963#	2972#	2976#	2980#	2987#	2994#	3012#	3014#	3016#	3025#	3031#	3073#
	3086#	3109#	3113#	3125#	3143#	3185#	3223#	3247#	3259#	3267#	3271#	3278#	3285#	3298#	3311#
	3320#	3322#	3324#	3326#	3328#	3402#	3404#	3406#	3412#	3423#	3431#	3443#	3452#	3462#	3524#
	3532#	3619#	3621#	3639#	3790#	3792#	3794#	3810#	3812#	3866#	3881#	3883#	3885#	3887#	3891#
	3895#	4004#	4006#	4008#	4013#	4045#	4050#	4055#	4061#	4072#	4080#	4087#	4092#	4119#	4121#
	4228#	4289#	4291#	4293#	4307#	4309#									
\$CKOP2	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1493#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1549#	1557#	1602#	1711#	1823#	1834#	1856#	1858#	1860#	1862#	1895#
	1917#	1919#	1929#	1936#	1944#	1986#	1998#	2001#	2077#	2103#	2154#	2156#	2169#	2172#	2202#
	2204#	2268#	2275#	2288#	2293#	2318#	2323#	2325#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2629#	2634#	2655#	2660#	2662#	2677#	2693#	2696#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2852#	2855#	2859#	2873#	2875#	2921#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3045#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3173#	3176#	3178#	3233#	3269#	3273#	3315#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3529#	3628#	3636#	3821#
	3854#	3879#	3889#	3893#	4069#	4077#	4335#	4337#	4339#						
\$CKOP6	1494#	1550#	2630#	3418#	3445#	3454#									
\$CMD	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	1625#	1626#	1628#	1845#	1846#	1848#	2056#
	2116#	2136#	2184#	2244#	2282#	2298#	2336#	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#



	2937#	2944#	2946	2952#	2969#	2984#	3002#	3004	3022#	3078#	3100#	3140#	3165#	3252#	3261#
	3275#	3408#	3409	3631#	3700#	3860#	3868#	3915#	4047#	4177#	4304#				
SRMPA	1014#	1152#	1283#	1305#	1309#	1326#	1376#	1380#	1498#	1511#	1554#	1579#	1584#	1626#	1648#
	1736#	1755#	1774#	1846#	2035#	2056#	2068#	2116#	2136#	2184#	2244#	2282#	2298#	2336#	2424#
	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#	2937#	2944#	2952#	2969#	2984#	3002#	3022#
	3078#	3100#	3140#	3165#	3252#	3261#	3275#	3409#	3542#	3578#	3601#	3647#	3681#	3700#	3708#
	3749#	3819#	3860#	3868#	3915#	4021#	4047#	4129#	4161#	4177#	4186#	4236#	4304#		
BLOUNT	1582#	1615#	1634#	1643#	1662#	1720#	1731#	1750#	1771#	1952#	1975#	2030#	2063#	2092#	2158#
	2260#	2419#	3537#	3549#	3569#	3580#	3641#	3649#	3674#	3683#	3703#	3710#	3715#	3743#	3751#
	3799#	3814#	4015#	4023#	4097#	4108#	4123#	4131#	4153#	4163#	4180#	4188#	4193#	4221#	4230#
	4238#	4243#	4311#												
SDO	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	1626#	1628#	1846#	1848#	2056#	2116#	2136#
	2184#	2244#	2282#	2298#	2336#	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#	2937#	2944#
	2946#	2952#	2969#	2984#	3002#	3004#	3022#	3078#	3100#	3140#	3165#	3252#	3261#	3275#	3409#
	3601#	3700#	3860#	3868#	3915#	4047#	4177#	4304#							
SELSE	1292#	1521#	2255#	2316#	2355#	2462#	2550#	2717#	2833#	2846#	2870#	2996#	3033#	3067#	3094#
	3152#	3287#	3305#	3910#	3934#	4052#									
SERRMS	1012#	1014#	1015#	1017#	1019#	1021#	1075#	1150#	1152#	1153#	1155#	1157#	1159#	1283#	1288#
	1290#	1292#	1293#	1305#	1307#	1309#	1310#	1312#	1314#	1316#	1318#	1326#	1328#	1330#	1336#
	1338#	1343#	1345#	1347#	1349#	1351#	1355#	1357#	1359#	1360#	1367#	1376#	1378#	1380#	1381#
	1383#	1385#	1387#	1389#	1391#	1392#	1398#	1402#	1404#	1406#	1408#	1409#	1440#	1442#	1449#
	1478#	1480#	1487#	1493#	1494#	1498#	1499#	1501#	1503#	1505#	1511#	1512#	1514#	1516#	1521#
	1522#	1524#	1526#	1532#	1534#	1536#	1541#	1543#	1549#	1550#	1554#	1555#	1557#	1559#	1561#
	1579#	1580#	1582#	1584#	1586#	1589#	1591#	1593#	1595#	1600#	1602#	1603#	1605#	1608#	1610#
	1615#	1625#	1626#	1627#	1628#	1629#	1634#	1643#	1648#	1654#	1657#	1662#	1670#	1671#	1676#
	1693#	1695#	1704#	1706#	1711#	1712#	1714#	1720#	1731#	1736#	1742#	1745#	1750#	1755#	1761#
	1764#	1769#	1771#	1779#	1785#	1788#	1793#	1814#	1816#	1823#	1824#	1829#	1834#	1835#	1840#
	1845#	1846#	1847#	1848#	1849#	1854#	1856#	1858#	1860#	1862#	1864#	1865#	1870#	1895#	1896#
	1901#	1906#	1908#	1910#	1912#	1917#	1919#	1924#	1929#	1931#	1936#	1937#	1939#	1944#	1945#
	1947#	1952#	1963#	1968#	1970#	1975#	1986#	1987#	1989#	1991#	1995#	1998#	1999#	2001#	2002#
	2008#	2030#	2035#	2047#	2050#	2056#	2057#	2059#	2061#	2063#	2068#	2072#	2075#	2077#	2078#
	2080#	2084#	2086#	2092#	2103#	2104#	2106#	2108#	2110#	2116#	2117#	2126#	2129#	2136#	2137#
	2146#	2149#	2154#	2156#	2158#	2169#	2170#	2172#	2174#	2176#	2178#	2184#	2185#	2194#	2197#
	2202#	2204#	2205#	2207#	2211#	2234#	2236#	2244#	2245#	2252#	2255#	2260#	2268#	2269#	2271#
	2273#	2275#	2277#	2282#	2283#	2288#	2289#	2291#	2293#	2298#	2299#	2301#	2303#	2310#	2316#
	2318#	2319#	2321#	2323#	2325#	2326#	2328#	2330#	2336#	2337#	2344#	2347#	2349#	2355#	2356#
	2358#	2360#	2367#	2370#	2372#	2377#	2396#	2398#	2405#	2407#	2409#	2419#	2424#	2433#	2435#
	2437#	2439#	2441#	2443#	2457#	2462#	2463#	2465#	2467#	2472#	2473#	2480#	2482#	2484#	2490#
	2491#	2497#	2498#	2507#	2508#	2513#	2514#	2524#	2526#	2528#	2530#	2532#	2534#	2545#	2548#
	2550#	2551#	2553#	2556#	2558#	2585#	2587#	2596#	2602#	2604#	2606#	2611#	2621#	2629#	2630#
	2634#	2640#	2642#	2643#	2649#	2650#	2655#	2656#	2658#	2660#	2662#	2663#	2675#	2677#	2678#
	2693#	2694#	2696#	2697#	2702#	2704#	2709#	2711#	2717#	2718#	2722#	2724#	2726#	2728#	2730#
	2741#	2744#	2751#	2753#	2803#	2804#	2812#	2813#	2818#	2820#	2822#	2824#	2826#	2828#	2833#
	2834#	2836#	2837#	2842#	2844#	2846#	2847#	2852#	2853#	2855#	2859#	2861#	2863#	2870#	2871#
	2873#	2875#	2877#	2908#	2913#	2914#	2916#	2921#	2922#	2924#	2926#	2928#	2933#	2935#	2937#
	2938#	2944#	2945#	2946#	2947#	2952#	2953#	2959#	2961#	2963#	2965#	2967#	2969#	2970#	2972#
	2974#	2976#	2978#	2980#	2982#	2984#	2985#	2987#	2989#	2994#	2996#	2997#	3002#	3004#	3005#
	3012#	3014#	3016#	3018#	3020#	3022#	3023#	3025#	3027#	3029#	3031#	3033#	3034#	3040#	3045#
	3046#	3048#	3050#	3052#	3054#	3060#	3062#	3067#	3068#	3073#	3078#	3079#	3086#	3088#	3090#
	3092#	3094#	3095#	3100#	3101#	3109#	3111#	3113#	3115#	3117#	3125#	3127#	3133#	3138#	3140#
	3141#	3143#	3145#	3147#	3152#	3153#	3159#	3161#	3165#	3166#	3168#	3173#	3174#	3176#	3178#
	3185#	3187#	3189#	3191#	3193#	3223#	3233#	3238#	3240#	3241#	3247#	3252#	3253#	3259#	3261#
	3262#	3267#	3269#	3271#	3273#	3275#	3276#	3278#	3280#	3285#	3287#	3288#	3298#	3300#	3303#
	3305#	3306#	3311#	3315#	3316#	3320#	3322#	3324#	3326#	3328#	3345#	3348#	3350#	3367#	3390#
	3392#	3402#	3404#	3406#	3408#	3409#	3410#	3412#	3414#	3416#	3418#	3423#	3425#	3427#	3429#
	3431#	3433#	3435#	3439#	3441#	3443#	3445#	3450#	3452#	3454#	3459#	3460#	3462#	3470#	3524#



	3529#	3530#	3532#	3537#	3542#	3549#	3564#	3569#	3578#	3580#	3595#	3601#	3602#	3614#	3619#
	3621#	3628#	3630#	3631#	3636#	3637#	3639#	3641#	3647#	3649#	3664#	3674#	3681#	3683#	3698#
	3700#	3701#	3703#	3708#	3710#	3715#	3730#	3732#	3737#	3738#	3743#	3749#	3751#	3766#	3790#
	3792#	3794#	3799#	3810#	3812#	3814#	3819#	3821#	3823#	3826#	3828#	3854#	3855#	3860#	3861#
	3866#	3868#	3869#	3879#	3881#	3883#	3885#	3887#	3889#	3891#	3893#	3895#	3907#	3910#	3915#
	3916#	3931#	3934#	3949#	3952#	3954#	3956#	3964#	4004#	4006#	4008#	4013#	4015#	4021#	4023#
	4038#	4045#	4047#	4048#	4050#	4052#	4053#	4055#	4057#	4061#	4063#	4064#	4069#	4070#	4072#
	4077#	4078#	4080#	4087#	4092#	4097#	4108#	4119#	4121#	4123#	4129#	4131#	4141#	4153#	4161#
	4163#	4173#	4177#	4178#	4180#	4186#	4188#	4193#	4208#	4210#	4215#	4216#	4221#	4228#	4230#
	4236#	4238#	4243#	4253#	4289#	4291#	4293#	4304#	4305#	4307#	4309#	4311#	4327#	4330#	4335#
	4337#	4339#	4344#	4345#	4353#										
%GENBR	1015#	1153#	1283#	1292#	1305#	1310#	1326#	1360#	1376#	1381#	1392#	1409#	1499#	1512#	1521#
	1555#	1580#	1584#	1587#	1627#	1629#	1648#	1655#	1670#	1736#	1743#	1755#	1762#	1779#	1786#
	1847#	1849#	1864#	2035#	2048#	2057#	2068#	2073#	2084#	2117#	2127#	2137#	2147#	2185#	2195#
	2245#	2253#	2283#	2299#	2311#	2337#	2345#	2355#	2368#	2424#	2462#	2473#	2491#	2498#	2508#
	2514#	2546#	2550#	2554#	2641#	2643#	2650#	2717#	2742#	2804#	2813#	2833#	2837#	2846#	2870#
	2914#	2938#	2945#	2947#	2953#	2970#	2985#	2996#	3003#	3005#	3023#	3033#	3067#	3079#	3094#
	3101#	3141#	3152#	3166#	3239#	3241#	3253#	3262#	3276#	3287#	3301#	3305#	3346#	3410#	3459#
	3542#	3578#	3602#	3631#	3647#	3681#	3701#	3708#	3738#	3749#	3819#	3824#	3861#	3869#	3908#
	3916#	3932#	3950#	4021#	4048#	4052#	4064#	4129#	4161#	4178#	4186#	4216#	4236#	4305#	4328#
	4345#														
%GENTA	1014#	1019#	1152#	1157#	1292#	1293#	1309#	1314#	1316#	1318#	1330#	1353#	1365#	1380#	1385#
	1389#	1400#	1498#	1505#	1511#	1521#	1522#	1526#	1554#	1559#	1561#	1562#	1579#	1589#	1595#
	1625#	1626#	1628#	1657#	1670#	1671#	1676#	1678#	1745#	1764#	1788#	1793#	1795#	1845#	1846#
	1848#	1864#	1865#	1870#	1871#	2008#	2009#	2050#	2056#	2075#	2086#	2116#	2129#	2136#	2149#
	2184#	2197#	2211#	2213#	2244#	2255#	2282#	2298#	2316#	2336#	2347#	2349#	2355#	2356#	2370#
	2372#	2377#	2379#	2414#	2462#	2463#	2467#	2472#	2484#	2497#	2503#	2513#	2548#	2550#	2551#
	2556#	2558#	2560#	2616#	2640#	2644#	2649#	2717#	2718#	2744#	2753#	2755#	2803#	2812#	2833#
	2834#	2836#	2846#	2847#	2861#	2863#	2870#	2871#	2877#	2913#	2933#	2937#	2944#	2946#	2952#
	2969#	2974#	2984#	2989#	2996#	2997#	3002#	3004#	3006#	3022#	3027#	3033#	3034#	3052#	3054#
	3062#	3067#	3068#	3078#	3094#	3095#	3100#	3140#	3145#	3152#	3153#	3165#	3187#	3189#	3191#
	3193#	3228#	3238#	3242#	3252#	3261#	3275#	3280#	3287#	3288#	3303#	3305#	3306#	3348#	3350#
	3352#	3367#	3368#	3408#	3409#	3459#	3460#	3470#	3471#	3564#	3595#	3601#	3614#	3623#	3664#
	3669#	3698#	3700#	3730#	3732#	3766#	3826#	3828#	3830#	3860#	3868#	3910#	3915#	3934#	3952#
	3954#	3956#	3964#	3966#	4038#	4047#	4052#	4053#	4057#	4059#	4141#	4148#	4173#	4177#	4208#
	4210#	4253#	4298#	4304#	4330#	4353#	4355#								
%IF	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	2056#	2116#	2136#	2184#	2244#	2282#	2298#
	2336#	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#	2937#	2944#	2952#	2969#	2984#	3002#
	3022#	3078#	3100#	3140#	3165#	3252#	3261#	3275#	3601#	3700#	3860#	3868#	3915#	4047#	4177#
	4304#														
%IFCOD	1014#	1152#	1309#	1359#	1380#	1391#	1408#	1498#	1511#	1554#	1579#	1626#	1628#	1846#	1848#
	2056#	2116#	2136#	2184#	2244#	2282#	2298#	2336#	2472#	2490#	2497#	2507#	2513#	2640#	2642#
	2649#	2803#	2812#	2836#	2913#	2937#	2944#	2946#	2952#	2969#	2984#	3004#	3022#	3078#	3100#
	3140#	3165#	3238#	3240#	3252#	3261#	3275#	3409#	3601#	3630#	3700#	3737#	3860#	3868#	3915#
	4047#	4063#	4177#	4215#	4304#	4344#									
%IFCON	1283#	1305#	1326#	1376#	1584#	1648#	1736#	1755#	1779#	2035#	2068#	2424#	3542#	3578#	3647#
	3681#	3778#	3749#	3819#	4021#	4129#	4161#	4186#	4236#						
%IFOPR	1014#	1015#	1152#	1153#	1283#	1305#	1309#	1310#	1326#	1359#	1360#	1376#	1380#	1381#	1391#
	1392#	1408#	1409#	1498#	1499#	1511#	1512#	1554#	1555#	1579#	1580#	1584#	1626#	1627#	1628#
	1629#	1648#	1736#	1755#	1779#	1846#	1847#	1848#	1849#	2035#	2056#	2057#	2068#	2116#	2117#
	2136#	2137#	2184#	2185#	2244#	2245#	2282#	2283#	2298#	2299#	2336#	2337#	2424#	2472#	2473#
	2490#	2491#	2497#	2498#	2507#	2508#	2513#	2514#	2642#	2643#	2649#	2650#	2803#	2804#	2812#
	2813#	2836#	2837#	2913#	2914#	2937#	2938#	2944#	2945#	2946#	2947#	2952#	2953#	2969#	2970#
	2984#	2985#	3004#	3005#	3022#	3023#	3078#	3079#	3100#	3101#	3140#	3141#	3165#	3166#	3240#
	3241#	3252#	3253#	3261#	3262#	3275#	3276#	3409#	3410#	3542#	3578#	3601#	3602#	3630#	3631#
	3647#	3681#	3700#	3701#	3708#	3737#	3738#	3749#	3819#	3860#	3861#	3868#	3869#	3915#	3916#

	4021#	4047#	4048	4063#	4064	4129#	4161#	4177#	4178	4186#	4215#	4216	4236#	4304#	4305
SLFT	4344#	4345													
	1012#	1017#	1021#	1075#	1150#	1155#	1159#	1288#	1290#	1307#	1312#	1328#	1336#	1338#	1343#
	1345#	1347#	1349#	1351#	1355#	1357#	1367#	1378#	1383#	1387#	1398#	1402#	1404#	1406#	1440#
	1442#	1449#	1487#	1493#	1501#	1503#	1514#	1516#	1524#	1532#	1534#	1536#	1541#	1543#	1549#
	1557#	1591#	1593#	1600#	1602#	1605#	1608#	1610#	1704#	1706#	1711#	1714#	1769#	1823#	1829#
	1834#	1840#	1854#	1856#	1858#	1860#	1862#	1895#	1901#	1906#	1908#	1910#	1912#	1917#	1919#
	1924#	1929#	1931#	1936#	1939#	1944#	1947#	1963#	1968#	1970#	1986#	1989#	1991#	1993#	1998#
	2001#	2059#	2061#	2077#	2080#	2103#	2106#	2108#	2110#	2154#	2156#	2169#	2172#	2174#	2176#
	2178#	2202#	2204#	2207#	2268#	2271#	2273#	2275#	2277#	2288#	2291#	2293#	2301#	2303#	2318#
	2321#	2323#	2325#	2328#	2330#	2358#	2360#	2405#	2407#	2409#	2433#	2435#	2437#	2439#	2441#
	2443#	2457#	2465#	2480#	2482#	2524#	2526#	2528#	2530#	2532#	2534#	2596#	2602#	2604#	2606#
	2611#	2621#	2629#	2634#	2655#	2658#	2660#	2662#	2675#	2677#	2693#	2696#	2702#	2704#	2709#
	2711#	2722#	2724#	2726#	2728#	2730#	2751#	2818#	2820#	2822#	2824#	2826#	2828#	2842#	2844#
	2852#	2855#	2859#	2873#	2875#	2908#	2916#	2921#	2924#	2926#	2928#	2935#	2959#	2961#	2963#
	2965#	2967#	2972#	2976#	2978#	2980#	2982#	2987#	2994#	3012#	3014#	3016#	3018#	3020#	3025#
	3029#	3031#	3040#	3045#	3048#	3050#	3060#	3073#	3086#	3088#	3090#	3092#	3109#	3111#	3113#
	3115#	3117#	3125#	3127#	3133#	3138#	3143#	3147#	3159#	3161#	3168#	3173#	3176#	3178#	3185#
	3223#	3233#	3247#	3259#	3267#	3269#	3271#	3273#	3278#	3285#	3298#	3311#	3315#	3320#	3322#
	3324#	3326#	3328#	3402#	3404#	3406#	3412#	3414#	3416#	3418#	3423#	3425#	3427#	3429#	3431#
	3433#	3435#	3439#	3441#	3443#	3445#	3450#	3452#	3454#	3462#	3524#	3529#	3532#	3619#	3621#
	3628#	3636#	3639#	3790#	3792#	3794#	3810#	3812#	3821#	3854#	3866#	3879#	3881#	3883#	3885#
	3887#	3889#	3891#	3893#	3895#	4004#	4006#	4008#	4013#	4045#	4050#	4055#	4061#	4069#	4072#
	4077#	4080#	4087#	4092#	4119#	4121#	4228#	4289#	4291#	4293#	4307#	4309#	4335#	4337#	4339#
SMCHIG	601#														
SMCLOW	601#														
SOPABS	1910#	1991#	2108#	2176#	2273#	2482#	3113#	3452#							
SOPADD	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3528#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
SOPAND	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
SOPCD1	1910#	1991#	2108#	2176#	2273#	2482#	3113#	3452#							
SOPCD2	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1493#	1494#	1501#	1503#
	1514#	1516#	1534#	1541#	1543#	1549#	1550#	1557#	1602#	1603#	1711#	1712#	1823#	1824#	1834#
	1835#	1856#	1858#	1860#	1862#	1895#	1896#	1917#	1919#	1929#	1936#	1937#	1944#	1945#	1986#
	1987#	1998#	1999#	2001#	2002#	2077#	2078#	2103#	2104#	2154#	2156#	2169#	2170#	2172#	2202#
	2204#	2205#	2268#	2269#	2275#	2288#	2289#	2293#	2318#	2319#	2323#	2325#	2326#	2330#	2457#
	2480#	2602#	2604#	2606#	2621#	2629#	2630#	2634#	2655#	2656#	2660#	2662#	2663#	2677#	2678#
	2693#	2694#	2696#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#	2842#	2844#	2852#	2853#
	2855#	2859#	2873#	2875#	2921#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#	3018#	3020#

SOPCOM  
SOPDEF

3029#	3040#	3045#	3046	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#	3133#
3138#	3147#	3159#	3161#	3168#	3173#	3174	3176#	3178#	3233#	3269#	3273#	3315#	3316	3414#
3416#	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3529#	3530	3628#
3636#	3637	3821#	3854#	3855	3879#	3889#	3893#	4069#	4070	4077#	4078	4335#	4337#	4339#
1910#	1991#	2108#	2176#	2273#	2482#	3113#	3452#							
1012#	1014#	1015#	1017#	1021#	1073#	1074#	1075#	1105#	1106#	1150#	1152#	1153#	1155#	1159#
1272#	1273#	1283#	1288#	1290#	1292#	1305#	1307#	1309#	1310#	1312#	1326#	1328#	1336#	1338#
1343#	1345#	1347#	1349#	1351#	1355#	1357#	1359#	1360#	1367#	1376#	1378#	1380#	1381#	1383#
1387#	1391#	1392#	1398#	1402#	1404#	1406#	1408#	1409#	1414#	1415#	1438#	1439#	1440#	1442#
1444#	1445#	1449#	1480#	1481	1482	1487#	1493#	1494#	1495	1496	1498#	1499#	1501#	1503#
1511#	1512#	1514#	1516#	1521#	1524#	1532#	1534#	1536#	1541#	1543#	1549#	1550#	1551	1552
1554#	1555#	1557#	1561#	1563#	1564	1565	1579#	1580#	1582#	1584#	1586#	1587#	1591#	1593#
1600#	1602#	1603#	1605#	1608#	1610#	1615#	1616	1617	1618	1626#	1627#	1628#	1629#	1634#
1635	1636#	1637	1638	1639	1640	1641	1643#	1648#	1654#	1655#	1662#	1663	1664	1665
1670#	1676#	1677	1679	1695#	1696	1697	1704#	1706#	1711#	1712#	1714#	1720#	1721	1722#
1723	1724	1725	1726	1731#	1736#	1742#	1743#	1750#	1755#	1761#	1762#	1769#	1771#	1772#
1773	1774	1779#	1785#	1786#	1793#	1794	1796#	1797	1798	1816#	1817	1818	1823#	1824#
1829#	1834#	1835#	1840#	1846#	1847#	1848#	1849#	1854#	1856#	1858#	1860#	1862#	1864#	1870#
1872#	1873	1874	1888#	1889#	1890#	1891#	1895#	1896#	1901#	1906#	1908#	1910#	1912#	1917#
1919#	1924#	1929#	1931#	1936#	1937#	1939#	1944#	1945#	1947#	1952#	1953	1954#	1955	1956
1957	1958	1963#	1968#	1970#	1975#	1976	1977#	1978	1979	1980	1981	1986#	1987#	1989#
1991#	1993#	1998#	1999#	2001#	2002#	2004#	2005#	2006#	2007#	2008#	2010	2025#	2026#	2030#
2035#	2045#	2046#	2047#	2048#	2056#	2057#	2059#	2061#	2063#	2064#	2065	2066	2068#	2070#
2071#	2072#	2073#	2077#	2078#	2080#	2082#	2083#	2084#	2092#	2093	2094#	2095	2096	2097
2098	2103#	2104#	2106#	2108#	2110#	2116#	2117#	2124#	2125#	2126#	2127#	2136#	2137#	2144#
2145#	2146#	2147#	2154#	2156#	2158#	2159	2160#	2161	2162	2163	2164	2169#	2170#	2172#
2174#	2176#	2178#	2184#	2185#	2192#	2193#	2194#	2195#	2202#	2204#	2205#	2207#	2209#	2210#
2211#	2212	2214	2236#	2237	2238	2244#	2245#	2252#	2253#	2260#	2261	2262#	2263	2264
2265	2266	2268#	2269#	2271#	2273#	2275#	2277#	2282#	2283#	2288#	2289#	2291#	2293#	2298#
2299#	2301#	2303#	2310#	2311#	2318#	2319#	2321#	2323#	2325#	2326#	2328#	2330#	2336#	2337#
2344#	2345#	2355#	2358#	2360#	2367#	2368#	2377#	2378	2380#	2381	2382	2398#	2399	2400
2405#	2407#	2409#	2419#	2424#	2426#	2427#	2428#	2429#	2433#	2435#	2437#	2439#	2441#	2443#
2445#	2446#	2447#	2448#	2457#	2462#	2465#	2472#	2473#	2480#	2482#	2490#	2491#	2497#	2498#
2507#	2508#	2513#	2514#	2516#	2517#	2518#	2519#	2524#	2526#	2528#	2530#	2532#	2534#	2536#
2537#	2538#	2539#	2545#	2546#	2550#	2553#	2554#	2558#	2559	2561#	2562	2563	2587#	2588
2589	2591#	2592#	2596#	2602#	2604#	2606#	2611#	2621#	2629#	2630#	2631	2632	2634#	2640#
2641#	2642#	2643#	2649#	2650#	2655#	2656#	2658#	2660#	2662#	2663#	2675#	2677#	2678#	2693#
2694#	2696#	2697#	2702#	2704#	2709#	2711#	2717#	2720#	2721#	2722#	2724#	2726#	2728#	2730#
2732#	2733#	2734#	2735#	2741#	2742#	2749#	2750#	2751#	2753#	2754	2756#	2757	2758	2803#
2804#	2810#	2811#	2812#	2813#	2818#	2820#	2822#	2824#	2826#	2828#	2833#	2836#	2837#	2842#
2844#	2846#	2852#	2853#	2855#	2859#	2865#	2866#	2870#	2873#	2875#	2900#	2901#	2908#	2913#
2914#	2916#	2921#	2922#	2924#	2926#	2928#	2930#	2931#	2935#	2937#	2938#	2944#	2945#	2946#
2947#	2952#	2953#	2959#	2961#	2963#	2965#	2967#	2969#	2970#	2972#	2976#	2978#	2980#	2982#
2984#	2985#	2987#	2994#	2996#	3002#	3003#	3004#	3005#	3012#	3014#	3016#	3018#	3020#	3022#
3023#	3025#	3029#	3031#	3033#	3040#	3045#	3046#	3048#	3050#	3060#	3067#	3073#	3078#	3079#
3086#	3088#	3090#	3092#	3094#	3100#	3101#	3107#	3108#	3109#	3111#	3113#	3115#	3117#	3119#
3120#	3125#	3127#	3133#	3138#	3140#	3141#	3143#	3147#	3152#	3159#	3161#	3165#	3166#	3168#
3173#	3174#	3176#	3178#	3185#	3198#	3199#	3223#	3233#	3238#	3239#	3240#	3241#	3247#	3252#
3253#	3259#	3261#	3262#	3267#	3269#	3271#	3273#	3275#	3276#	3278#	3285#	3287#	3298#	3300#
3301#	3305#	3311#	3313#	3314#	3315#	3316#	3318#	3319#	3320#	3322#	3324#	3326#	3328#	3330#
3331#	3332#	3333#	3345#	3346#	3350#	3351	3353	3367#	3369	3392#	3393	3394	3396#	3397#
3398#	3399#	3400#	3401#	3402#	3404#	3406#	3409#	3410#	3412#	3414#	3416#	3418#	3419	3420
3421	3423#	3425#	3427#	3429#	3431#	3433#	3435#	3436	3437	3439#	3441#	3443#	3445#	3446
3447	3448	3450#	3452#	3454#	3455	3456	3457	3459#	3462#	3464#	3465#	3466#	3467#	3468#
3469#	3470#	3472#	3473	3474	3524#	3529#	3530#	3532#	3537#	3542#	3549#	3550#	3551	3552
3569#	3570	3571	3578#	3580#	3581#	3582	3583	3601#	3602#	3619#	3621#	3628#	3630#	3631#

	3636#	3637#	3639#	3641#	3647#	3649#	3650#	3651#	3652#	3674#	3675#	3676#	3681#	3683#	3684#
	3685#	3686#	3700#	3701#	3703#	3708#	3710#	3711#	3712#	3713#	3715#	3716#	3717#	3718#	3737#
	3738#	3743#	3749#	3751#	3752#	3753#	3754#	3790#	3792#	3794#	3799#	3800#	3801#	3802#	3803#
	3804#	3805#	3810#	3812#	3814#	3819#	3821#	3823#	3824#	3828#	3829#	3831#	3845#	3846#	3847#
	3848#	3849#	3850#	3854#	3855#	3860#	3861#	3866#	3868#	3869#	3879#	3881#	3883#	3885#	3887#
	3889#	3891#	3893#	3895#	3901#	3902#	3903#	3904#	3905#	3906#	3907#	3908#	3915#	3916#	3925#
	3926#	3927#	3928#	3929#	3930#	3931#	3932#	3943#	3944#	3945#	3946#	3947#	3948#	3949#	3950#
	3958#	3959#	3960#	3961#	3962#	3963#	3964#	3965#	3967#	4004#	4006#	4008#	4013#	4015#	4021#
	4023#	4024#	4025#	4026#	4043#	4044#	4045#	4047#	4048#	4050#	4052#	4055#	4061#	4063#	4064#
	4069#	4070#	4072#	4077#	4078#	4080#	4082#	4083#	4087#	4092#	4097#	4098#	4099#	4100#	4101#
	4102#	4103#	4108#	4109#	4110#	4111#	4112#	4113#	4114#	4119#	4121#	4123#	4129#	4131#	4132#
	4133#	4134#	4153#	4154#	4155#	4161#	4163#	4164#	4165#	4166#	4177#	4178#	4180#	4186#	4188#
	4189#	4190#	4191#	4193#	4194#	4195#	4196#	4215#	4216#	4221#	4228#	4230#	4236#	4238#	4239#
	4240#	4241#	4243#	4244#	4245#	4246#	4278#	4279#	4280#	4281#	4282#	4283#	4289#	4291#	4293#
	4304#	4305#	4307#	4309#	4311#	4312#	4313#	4314#	4321#	4322#	4323#	4324#	4325#	4326#	4327#
	4328#	4335#	4337#	4339#	4344#	4345#	4347#	4348#	4349#	4350#	4351#	4352#	4353#	4354#	4356#
%OPFOU	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
%OPNAN	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
%OPNFC	1910#	1991#	2108#	2176#	2273#	2482#	3113#	3452#							
%OPNOR	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
%OPNOT	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#

\$OPOR	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#	1442#	1449#	1494#	1501#	1503#	1514#
	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1835#	1856#	1858#	1860#	1862#	1896#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	2104#	2154#	2156#	2170#	2172#	2202#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2457#	2480#	2602#	2604#	2606#	2621#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2711#	2751#	2820#	2822#	2826#	2828#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2922#	2924#	2926#	2935#	2965#	2967#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
\$OPROT	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#	1442#	1449#	1494#	1501#	1503#	1514#
	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1835#	1856#	1858#	1860#	1862#	1896#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	2104#	2154#	2156#	2170#	2172#	2202#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2457#	2480#	2602#	2604#	2606#	2621#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2711#	2751#	2820#	2822#	2826#	2828#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2922#	2924#	2926#	2935#	2965#	2967#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
\$OPRO	1012#	1017#	1021#	1075#	1150#	1155#	1159#	1288#	1290#	1307#	1312#	1328#	1336#	1338#	1343#
	1345#	1347#	1349#	1351#	1355#	1357#	1367#	1378#	1383#	1387#	1398#	1402#	1404#	1406#	1440#
	1442#	1449#	1487#	1493#	1501#	1503#	1514#	1516#	1524#	1532#	1534#	1536#	1541#	1543#	1549#
	1557#	1591#	1593#	1600#	1602#	1605#	1608#	1610#	1704#	1706#	1711#	1714#	1769#	1823#	1829#
	1834#	1840#	1854#	1856#	1858#	1860#	1862#	1895#	1901#	1906#	1908#	1910#	1912#	1917#	1919#
	1924#	1929#	1931#	1936#	1939#	1944#	1947#	1963#	1968#	1970#	1986#	1989#	1991#	1993#	1998#
	2001#	2059#	2061#	2077#	2080#	2103#	2106#	2108#	2110#	2154#	2156#	2169#	2172#	2174#	2176#
	2178#	2202#	2204#	2207#	2268#	2271#	2273#	2275#	2277#	2288#	2291#	2293#	2301#	2303#	2318#
	2321#	2323#	2325#	2328#	2330#	2358#	2360#	2405#	2407#	2409#	2433#	2435#	2437#	2439#	2441#
	2443#	2457#	2465#	2480#	2482#	2524#	2526#	2528#	2530#	2532#	2534#	2596#	2602#	2604#	2606#
	2611#	2621#	2629#	2634#	2655#	2658#	2660#	2662#	2675#	2677#	2693#	2696#	2702#	2704#	2709#
	2711#	2722#	2724#	2726#	2728#	2730#	2751#	2818#	2820#	2822#	2824#	2826#	2828#	2842#	2844#
	2852#	2855#	2859#	2873#	2875#	2908#	2916#	2921#	2924#	2926#	2928#	2935#	2959#	2961#	2963#
	2965#	2967#	2972#	2976#	2978#	2980#	2982#	2987#	2994#	3012#	3014#	3016#	3018#	3020#	3025#
	3029#	3031#	3040#	3045#	3048#	3050#	3060#	3073#	3086#	3088#	3090#	3092#	3109#	3111#	3113#
	3115#	3117#	3125#	3127#	3133#	3138#	3143#	3147#	3159#	3161#	3168#	3173#	3176#	3178#	3185#
	3223#	3233#	3247#	3259#	3267#	3269#	3271#	3273#	3278#	3285#	3298#	3311#	3315#	3320#	3322#
	3324#	3326#	3328#	3402#	3404#	3406#	3412#	3414#	3416#	3418#	3423#	3425#	3427#	3429#	3431#
	3433#	3435#	3439#	3441#	3443#	3445#	3450#	3452#	3454#	3462#	3524#	3529#	3532#	3619#	3621#
	3628#	3636#	3639#	3790#	3792#	3794#	3810#	3812#	3821#	3854#	3866#	3879#	3881#	3883#	3885#
	3887#	3889#	3891#	3893#	3895#	4004#	4006#	4008#	4013#	4045#	4050#	4055#	4061#	4069#	4072#
	4077#	4080#	4087#	4092#	4119#	4121#	4228#	4289#	4291#	4293#	4307#	4309#	4335#	4337#	4339#
\$OPR1	1012#	1017#	1021#	1075#	1150#	1155#	1159#	1288#	1290#	1307#	1312#	1328#	1336#	1338#	1343#
	1345#	1347#	1349#	1351#	1355#	1357#	1367#	1378#	1383#	1387#	1398#	1402#	1404#	1406#	1440#
	1442#	1449#	1487#	1493#	1501#	1503#	1514#	1516#	1524#	1532#	1534#	1536#	1541#	1543#	1549#
	1557#	1591#	1593#	1600#	1602#	1605#	1608#	1610#	1704#	1706#	1711#	1714#	1769#	1823#	1829#
	1834#	1840#	1854#	1856#	1858#	1860#	1862#	1895#	1901#	1906#	1908#	1910#	1912#	1917#	1919#
	1924#	1929#	1931#	1936#	1939#	1944#	1947#	1963#	1968#	1970#	1986#	1989#	1991#	1993#	1998#
	2001#	2059#	2061#	2077#	2080#	2103#	2106#	2108#	2110#	2154#	2156#	2169#	2172#	2174#	2176#
	2178#	2202#	2204#	2207#	2268#	2271#	2273#	2275#	2277#	2288#	2291#	2293#	2301#	2303#	2318#
	2321#	2323#	2325#	2328#	2330#	2358#	2360#	2405#	2407#	2409#	2433#	2435#	2437#	2439#	2441#
	2443#	2457#	2465#	2480#	2482#	2524#	2526#	2528#	2530#	2532#	2534#	2596#	2602#	2604#	2606#
	2611#	2621#	2629#	2634#	2655#	2658#	2660#	2662#	2675#	2677#	2693#	2696#	2702#	2704#	2709#
	2711#	2722#	2724#	2726#	2728#	2730#	2751#	2818#	2820#	2822#	2824#	2826#	2828#	2842#	2844#
	2852#	2855#	2859#	2873#	2875#	2908#	2916#	2921#	2924#	2926#	2928#	2935#	2959#	2961#	2963#



	2965#	2967#	2972#	2976#	2978#	2980#	2982#	2987#	2994#	3012#	3014#	3016#	3018#	3020#	3025#
	3029#	3031#	3040#	3045#	3048#	3050#	3060#	3073#	3086#	3088#	3090#	3092#	3109#	3111#	3113#
	3115#	3117#	3125#	3127#	3133#	3138#	3143#	3147#	3159#	3161#	3168#	3173#	3176#	3178#	3185#
	3223#	3233#	3247#	3259#	3267#	3269#	3271#	3273#	3278#	3285#	3298#	3311#	3315#	3320#	3322#
	3324#	3326#	3328#	3402#	3404#	3406#	3412#	3414#	3416#	3418#	3423#	3425#	3427#	3429#	3431#
	3433#	3435#	3439#	3441#	3443#	3445#	3450#	3452#	3454#	3462#	3524#	3529#	3532#	3619#	3621#
	3628#	3636#	3639#	3790#	3792#	3794#	3810#	3812#	3821#	3854#	3866#	3879#	3881#	3883#	3885#
	3887#	3889#	3891#	3893#	3895#	4004#	4006#	4008#	4013#	4045#	4050#	4055#	4061#	4069#	4072#
	4077#	4080#	4087#	4092#	4119#	4121#	4228#	4289#	4291#	4293#	4307#	4309#	4335#	4337#	4339#
\$OPR2	1012#	1017#	1021#	1075#	1150#	1155#	1159#	1288#	1290#	1307#	1312#	1328#	1336#	1338#	1343#
	1345#	1347#	1349#	1351#	1355#	1357#	1367#	1378#	1383#	1387#	1398#	1402#	1404#	1406#	1440#
	1442#	1449#	1487#	1493#	1501#	1503#	1514#	1516#	1524#	1532#	1534#	1536#	1541#	1543#	1549#
	1557#	1591#	1593#	1600#	1602#	1605#	1608#	1610#	1704#	1706#	1711#	1714#	1769#	1823#	1829#
	1834#	1840#	1854#	1856#	1858#	1860#	1862#	1895#	1901#	1906#	1908#	1910#	1912#	1917#	1919#
	1924#	1929#	1931#	1936#	1939#	1944#	1947#	1963#	1968#	1970#	1986#	1989#	1991#	1993#	1998#
	2001#	2059#	2061#	2077#	2080#	2103#	2106#	2108#	2110#	2154#	2156#	2169#	2172#	2174#	2176#
	2178#	2202#	2204#	2207#	2268#	2271#	2273#	2275#	2277#	2288#	2291#	2293#	2301#	2303#	2318#
	2321#	2323#	2325#	2328#	2330#	2358#	2360#	2405#	2407#	2409#	2433#	2435#	2437#	2439#	2441#
	2443#	2457#	2465#	2480#	2482#	2524#	2526#	2528#	2530#	2532#	2534#	2596#	2602#	2604#	2606#
	2611#	2621#	2629#	2634#	2655#	2658#	2660#	2662#	2675#	2677#	2693#	2696#	2702#	2704#	2709#
	2711#	2722#	2724#	2726#	2728#	2730#	2751#	2818#	2820#	2822#	2824#	2826#	2828#	2842#	2844#
	2852#	2855#	2859#	2873#	2875#	2908#	2916#	2921#	2924#	2926#	2928#	2935#	2959#	2961#	2963#
	2965#	2967#	2972#	2976#	2978#	2980#	2982#	2987#	2994#	3012#	3014#	3016#	3018#	3020#	3025#
	3029#	3031#	3040#	3045#	3048#	3050#	3060#	3073#	3086#	3088#	3090#	3092#	3109#	3111#	3113#
	3115#	3117#	3125#	3127#	3133#	3138#	3143#	3147#	3159#	3161#	3168#	3173#	3176#	3178#	3185#
	3223#	3233#	3247#	3259#	3267#	3269#	3271#	3273#	3278#	3285#	3298#	3311#	3315#	3320#	3322#
	3324#	3326#	3328#	3402#	3404#	3406#	3412#	3414#	3416#	3418#	3423#	3425#	3427#	3429#	3431#
	3433#	3435#	3439#	3441#	3443#	3445#	3450#	3452#	3454#	3462#	3524#	3529#	3532#	3619#	3621#
	3628#	3636#	3639#	3790#	3792#	3794#	3810#	3812#	3821#	3854#	3866#	3879#	3881#	3883#	3885#
	3887#	3889#	3891#	3893#	3895#	4004#	4006#	4008#	4013#	4045#	4050#	4055#	4061#	4069#	4072#
	4077#	4080#	4087#	4092#	4119#	4121#	4228#	4289#	4291#	4293#	4307#	4309#	4335#	4337#	4339#
\$OPSHF	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
\$OPSUB	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#
	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
\$OPSWB	1910#	1991#	2108#	2176#	2273#	2482#	3113#	3452#							
\$OPXOR	1021#	1159#	1290#	1307#	1357#	1367#	1378#	1404#	1406#	1442#	1449#	1494#	1501#	1503#	1514#
	1516#	1534#	1541#	1543#	1550#	1557#	1603#	1712#	1824#	1835#	1856#	1858#	1860#	1862#	1896#
	1917#	1919#	1929#	1937#	1945#	1987#	1999#	2002#	2078#	2104#	2154#	2156#	2170#	2172#	2202#
	2205#	2269#	2275#	2289#	2293#	2319#	2323#	2326#	2330#	2457#	2480#	2602#	2604#	2606#	2621#

	2630#	2634#	2656#	2660#	2663#	2678#	2694#	2697#	2709#	2711#	2751#	2820#	2822#	2826#	2828#
	2842#	2844#	2853#	2855#	2859#	2873#	2875#	2922#	2924#	2926#	2935#	2965#	2967#	2978#	2982#
	3018#	3020#	3029#	3040#	3046#	3048#	3050#	3060#	3088#	3090#	3092#	3111#	3115#	3117#	3127#
	3133#	3138#	3147#	3159#	3161#	3168#	3174#	3176#	3178#	3233#	3269#	3273#	3316#	3414#	3416#
	3418#	3425#	3427#	3429#	3433#	3435#	3439#	3441#	3445#	3450#	3454#	3530#	3628#	3637#	3821#
	3855#	3879#	3889#	3893#	4070#	4078#	4335#	4337#	4339#						
SSR	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	1626#	1628#	1846#	1848#	2056#	2116#	2136#
	2184#	2244#	2282#	2298#	2336#	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#	2937#	2944#
	2946#	2952#	2969#	2984#	3002#	3004#	3022#	3078#	3100#	3140#	3165#	3252#	3261#	3275#	3409#
	3601#	3700#	3860#	3868#	3915#	4047#	4177#	4304#							
SPUT	1582#	1615#	1634#	1636#	1643#	1662#	1720#	1722#	1731#	1750#	1771#	1772#	1952#	1954#	1975#
	1977#	2030#	2063#	2064#	2092#	2094#	2158#	2160#	2260#	2262#	2419#	3537#	3549#	3550#	3569#
	3580#	3581#	3641#	3649#	3650#	3674#	3683#	3684#	3703#	3710#	3711#	3715#	3716#	3743#	3751#
	3752#	3799#	3801#	3814#	4015#	4023#	4024#	4097#	4099#	4108#	4110#	4123#	4131#	4132#	4153#
	4163#	4164#	4180#	4188#	4199#	4193#	4194#	4221#	4230#	4238#	4239#	4243#	4244#	4311#	4312#
SSUBON	1019#	1157#	1293#	1314#	1316#	1318#	1330#	1359#	1385#	1389#	1391#	1408#	1505#	1522#	1526#
	1559#	1561#	1589#	1595#	1657#	1670#	1671#	1676#	1745#	1764#	1788#	1793#	1864#	1865#	1870#
	2008#	2050#	2075#	2086#	2129#	2149#	2197#	2211#	2255#	2316#	2347#	2349#	2356#	2370#	2372#
	2377#	2463#	2467#	2484#	2490#	2507#	2548#	2551#	2556#	2558#	2640#	2718#	2744#	2753#	2834#
	2847#	2861#	2863#	2871#	2877#	2933#	2974#	2989#	2997#	3027#	3034#	3052#	3054#	3062#	3068#
	3095#	3145#	3153#	3187#	3189#	3191#	3193#	3238#	3280#	3288#	3303#	3306#	3348#	3350#	3367#
	3459#	3460#	3470#	3564#	3595#	3614#	3630#	3664#	3698#	3730#	3732#	3737#	3766#	3826#	3828#
	3910#	3934#	3952#	3954#	3956#	3964#	4038#	4053#	4057#	4063#	4141#	4173#	4208#	4210#	4215#
	4253#	4330#	4344#	4353#											
STHEN	1014#	1152#	1309#	1380#	1498#	1511#	1554#	1579#	1626#	1628#	1846#	1848#	2056#	2116#	2136#
	2184#	2244#	2282#	2298#	2336#	2472#	2497#	2513#	2649#	2803#	2812#	2836#	2913#	2937#	2944#
	2946#	2952#	2969#	2984#	3002#	3004#	3022#	3078#	3100#	3140#	3165#	3252#	3261#	3275#	3409#
	3601#	3700#	3860#	3868#	3915#	4047#	4177#	4304#							
SUNTL2	1359#	1391#	1408#	2490#	2507#	2640#	3238#	3630#	3737#	4063#	4215#	4344#			
SUNTL3	1359#	1391#	1408#	2490#	2507#	2640#	3238#	3630#	3737#	4063#	4215#	4344#			
SUWILE	125#	1845#	3408#												
SSGEN	1019#	1157#	1293#	1314#	1316#	1318#	1330#	1353#	1365#	1385#	1389#	1400#	1475#	1505#	1522#
	1526#	1559#	1561#	1562#	1577#	1589#	1595#	1625#	1657#	1671#	1676#	1678#	1691#	1745#	1764#
	1788#	1793#	1795#	1812#	1845#	1865#	1870#	1871#	1886#	2008#	2009#	2020#	2050#	2075#	2086#
	2129#	2149#	2197#	2211#	2213#	2232#	2255#	2316#	2347#	2349#	2356#	2370#	2372#	2377#	2379#
	2394#	2414#	2463#	2467#	2484#	2503#	2548#	2551#	2556#	2558#	2560#	2583#	2616#	2644#	2718#
	2744#	2753#	2755#	2834#	2847#	2861#	2863#	2871#	2877#	2933#	2974#	2989#	2997#	3006#	3027#
	3034#	3052#	3054#	3062#	3068#	3095#	3145#	3153#	3187#	3189#	3191#	3193#	3218#	3228#	3242#
	3280#	3288#	3303#	3306#	3348#	3350#	3352#	3365#	3367#	3368#	3388#	3408#	3460#	3470#	3471#
	3564#	3595#	3614#	3623#	3664#	3669#	3698#	3730#	3732#	3766#	3784#	3826#	3828#	3830#	3843#
	3910#	3934#	3952#	3954#	3956#	3964#	3966#	4038#	4053#	4057#	4059#	4141#	4148#	4173#	4208#
	4210#	4253#	4276#	4298#	4330#	4353#	4355#								
SSGETS	1019#	1157#	1292#	1293#	1314#	1316#	1318#	1330#	1359#	1385#	1389#	1391#	1408#	1505#	1521#
	1522#	1526#	1559#	1561#	1589#	1595#	1657#	1670#	1671#	1676#	1745#	1764#	1788#	1793#	1864#
	1865#	1870#	2008#	2050#	2075#	2086#	2129#	2149#	2197#	2211#	2255#	2316#	2347#	2349#	2355#
	2356#	2370#	2372#	2377#	2462#	2463#	2467#	2484#	2490#	2507#	2548#	2550#	2551#	2556#	2558#
	2640#	2717#	2718#	2744#	2753#	2833#	2834#	2846#	2847#	2861#	2863#	2870#	2871#	2877#	2933#
	2974#	2989#	2996#	2997#	3027#	3033#	3034#	3052#	3054#	3062#	3067#	3068#	3094#	3095#	3145#
	3152#	3153#	3187#	3189#	3191#	3193#	3238#	3280#	3287#	3288#	3303#	3305#	3306#	3348#	3350#
	3367#	3459#	3460#	3470#	3564#	3595#	3614#	3630#	3664#	3698#	3730#	3732#	3737#	3766#	3826#
	3828#	3910#	3934#	3952#	3954#	3956#	3964#	4038#	4052#	4053#	4057#	4063#	4141#	4173#	4208#
	4210#	4215#	4253#	4330#	4344#	4353#									
SSGETT	1292#	1521#	2255#	2316#	2355#	2462#	2550#	2717#	2833#	2846#	2870#	2996#	3033#	3067#	3094#
	3152#	3287#	3305#	3910#	3934#	4052#									
SSPOP	1019#	1157#	1292#	1293#	1314#	1316#	1318#	1330#	1359#	1385#	1389#	1391#	1408#	1505#	1521#
	1522#	1526#	1559#	1561#	1589#	1595#	1657#	1670#	1671#	1676#	1745#	1764#	1788#	1793#	1864#

	1865	1870#	2008#	2050#	2075#	2086#	2129#	2149#	2197#	2211#	2255#	2316#	2347#	2349#	2355#
	2356	2370#	2372#	2377#	2462#	2463	2467#	2484#	2490#	2507#	2548#	2550#	2551	2556#	2558#
	2640#	2717#	2718	2744#	2753#	2833#	2834	2846#	2847	2861#	2863#	2870#	2871	2877#	2933#
	2974#	2989#	2996#	2997	3027#	3033#	3034	3052#	3054#	3062#	3067#	3068	3094#	3095	3145#
	3152#	3153	3187#	3189#	3191#	3193#	3238#	3280#	3287#	3288	3303#	3305#	3306	3348#	3350#
	3367#	3459#	3460	3470#	3564#	3595#	3614#	3630#	3664#	3698#	3730#	3732#	3737#	3766#	3826#
	3828#	3910#	3934#	3952#	3954#	3956#	3964#	4038#	4052#	4053	4057#	4063#	4141#	4173#	4208#
	4210#	4215#	4253#	4330#	4344#	4353#									
SSPUSH	1014#	1016	1152#	1154	1283#	1284	1292#	1294	1305#	1306	1309#	1311	1326#	1327	1353#
	1354	1365#	1366	1376#	1377	1380#	1382	1400#	1401	1475#	1476	1498#	1500	1511#	1513
	1521#	1523	1554#	1556	1577#	1578	1579#	1581	1584#	1585	1625#	1626	1630	1648#	1649
	1670#	1691#	1692	1736#	1737	1755#	1756	1779#	1780	1812#	181#	1845#	1846	1850	1864#
	1886#	1887	2020#	2021	2035#	2036	2056#	2058	2068#	2069	211#	2118	2136#	2138	2184#
	2186	2232#	2233	2244#	2246	2255#	2256	2282#	2284	2298#	2300	2316#	2317	2336#	2338
	2355#	2357	2394#	2395	2414#	2415	2424#	2425	2462#	2464	2472#	2474	2497#	2499	2503#
	2504	2513#	2515	2550#	2552	2583#	2584	2616#	2617	2649#	2651	2717#	2719	2803#	2805
	2812#	2814	2833#	2835	2836#	2838	2846#	2848	2870#	2872	2913#	2915	2937#	2939	2944#
	2948	2952#	2954	2969#	2971	2984#	2986	2996#	2998	3002#	3007	3022#	3024	3033#	3035
	3067#	3069	3078#	3080	3094#	3096	3100#	3102	3140#	3142	3152#	3154	3165#	3167	3218#
	3219	3228#	3229	3252#	3254	3261#	3263	3275#	3277	3287#	3289	3305#	3307	3365#	3366
	3388#	3389	3408#	3409	3411	3459#	3542#	3543	3578#	3579	3601#	3603	3623#	3624	3647#
	3648	3669#	3670	3681#	3682	3700#	3702	3708#	3709	3749#	3750	3784#	3785	3819#	3820
	3843#	3844	3860#	3862	3868#	3870	3910#	3911	3915#	3917	3934#	3935	4021#	4022	4047#
	4049	4052#	4054	4059#	4060	4129#	4130	4148#	4149	4161#	4162	4177#	4179	4186#	4187
	4236#	4237	4276#	4277	4298#	4299	4304#	4306							
SSSETS	1014#	1016#	1152#	1154#	1283#	1284#	1294#	1305#	1306#	1309#	1311#	1326#	1327#	1353#	1354#
	1365#	1366#	1376#	1377#	1380#	1382#	1400#	1401#	1476#	1498#	1500#	1511#	1513#	1523#	1554#
	1556#	1578#	1579#	1581#	1584#	1585#	1625#	1626#	1630#	1648#	1649#	1670#	1692#	1736#	1737#
	1755#	1756#	1779#	1780#	1813#	1845#	1846#	1850#	1864#	1887#	2021#	2035#	2036#	2056#	2058#
	2068#	2069#	2116#	2118#	2136#	2138#	2184#	2186#	2233#	2244#	2246#	2256#	2282#	2284#	2298#
	2300#	2317#	2336#	2338#	2357#	2395#	2414#	2415#	2424#	2425#	2464#	2472#	2474#	2497#	2499#
	2503#	2504#	2513#	2515#	2552#	2584#	2616#	2617#	2649#	2651#	2719#	2803#	2805#	2812#	2814#
	2835#	2836#	2838#	2848#	2872#	2913#	2915#	2937#	2939#	2944#	2948#	2952#	2954#	2969#	2971#
	2984#	2986#	2998#	3002#	3007#	3022#	3024#	3035#	3069#	3078#	3080#	3096#	3100#	3102#	3140#
	3142#	3154#	3165#	3167#	3219#	3228#	3229#	3252#	3254#	3261#	3263#	3275#	3277#	3289#	3307#
	3366#	3389#	3408#	3409#	3411#	3459#	3542#	3543#	3578#	3579#	3601#	3603#	3623#	3624#	3647#
	3648#	3669#	3670#	3681#	3682#	3700#	3702#	3708#	3709#	3749#	3750#	3785#	3819#	3820#	3844#
	3860#	3862#	3868#	3870#	3911#	3915#	3917#	3935#	4021#	4022#	4047#	4049#	4054#	4059#	4060#
	4129#	4130#	4148#	4149#	4161#	4162#	4177#	4179#	4186#	4187#	4236#	4237#	4277#	4298#	4299#
	4304#	4306#													

. ABS. 053230 000

ERRORS DETECTED: 0

(ZDPGB.BIN,CZDPGB.SEQ/CRF/SOL/NL:TOC=SMAC/ML,SPMAC/ML,(ZDPGB.P11  
RUN-TIME: 108 109 10 SECONDS  
RUN-TIME RATIO: 2280/228=9.9  
CORE USED: 32K (63 PACFS)