EXTENDED ARITHMETIC UNIT DIAGNOSTIC

HP Product No. 24186

HEWLETT hp PACKARD

11000 Wolfe Road
Cupertino, California 95014

Manual of Diagnostics
Diagnostic Program Procedure
12579-90013

January 1971

EXTENDED ARITHMETIC UNIT DIAGNOSTIC

This diagnostic program confirms proper operation of the HP 12579A Extended Arithmetic Unit (EAU) option for the HP 2115 and 2116 computers.

The program tests the EAU instruction set (i.e., DLD, DST, MPY, DIV etc.) by running non-EAU subroutines and EAU instructions with the same arguments and comparing the actual results with the expected results.

HARDWARE CONFIGURATION

This program runs on an HP 2115 or 2116 computer (4K minimum core size) with the HP 12579A EAU option. A teleprinter can optionally be included to report diagnostic messages during execution of the program.

FUNCTIONAL AND OPERATIONAL CHARACTERISTICS

Since the actual EAU results are compared with the expected results obtained by running the non-EAU routines, the diagnostic should be executed only after the following diagnostics have been successfully executed:

- Alter-Skip Instruction Test
- Memory Reference Instruction Test
- Shift-Rotate Instruction Test
- Teletype (TTY) Diagnostic (if a teleprinter is used)

If the teleprinter is used, the SIO teleprinter driver is loaded and configured first. Then the diagnostic program is loaded and optionally configured by setting the SWITCH REGISTER as shown in Table EAU-1. The configuration process loads the hardware SWITCH REGISTER bit settings into an internal switch register, which is read by the diagnostic during execution (if hardware SWITCH REGISTER bit 0 is set OFF).

To modify the program options during execution, set the SWITCH REGISTER bit 0 ON and set the other program option bits according to Table EAU-1. With SWITCH REGISTER bit 0 set ON, the diagnostic ignores the internal switch register and reads the hardware SWITCH REGISTER to determine program control. The program can be reconfigured at any time by starting the program at location 111_8*, setting the SWITCH REGISTER program option bits and pressing RUN. (See Operating Instructions, step h.)

To avoid reconfiguring the driver and diagnostic before subsequent uses, load the SIO System Dump program and use it to punch a paper tape copy of the configured driver and diagnostic.

To start a run of the diagnostic after configuration is complete or after a configured tape has been loaded, set the computer to the starting address, select program options by setting the SWITCH REGISTER as listed in Table EAU-1 and press RUN.

If an error is detected, the program can print a message on the teleprinter and/or halt with a value displayed in the computer MEMORY DATA Register, depending upon the SWITCH REGISTER settings. (See Diagnostic Messages.)

If a trap cell halt occurs, the computer displays 1060*xx* (*xx* = the trap cell location). The cause of the halt must be determined by the user; after the error is corrected, the user restarts the program from location 100_8.

LIMITATIONS

This diagnostic is a unit diagnostic. No system interaction with the Direct Memory Access (DMA), Memory Protect, or Memory Parity options is checked.

**The program can be configured starting at 2_8 only after loading and before the SIO System Dump is loaded or the program is run for the first time.*

This diagnostic does not check the EAU in single cycle operation.

This diagnostic does not test that the CRS signal resets the EAU option.

PROGRAM ORGANIZATION

This diagnostic consists of ten sections; each tests one of the ten EAU instructions:

Test Number (octal)	Name
1	DLD (Double Load)
2	DST (Double Store)
3	MPY (Multiply)
4	DIV (Divide)
5	ASR (Arithmetic Long Shift Right)
6	ASL (Arithmetic Long Shift Left)
7	LSR (Logical Long Shift Right)
10	LSL (Logical Long Shift Left)
11	RRR (Rotate A and B Right)
12	RRL (Rotate A and B Left)

The diagnostic executes the ten sections in sequence. One pass through the diagnostic is defined as 1500 loops through the ten sections. A loop is defined as execution of all ten sections with a new argument generated for each section (program option bit 6 set OFF for all ten instructions). If program option bit 6 is set ON at any time before or during any execution of the ten sections, the loop is not counted toward the 1500 which constitute a pass.

MEMORY ALLOCATION

The arguments generated before each instruction test (as long as program option bit 6 is set OFF) are Located in specific areas of core. (See Figure EAU-1.)

Before each EAU instruction test (except DLD) the A- and B-Registers are loaded with the contents of RNA and RNB respectively. For DLD, RNA and RNB are expected to be in the A- and B-Registers after the instruction is executed.

Bit 0 of the contents of RNE and RNO are set Into the E and OV Register, respectively, before each EAU instruction. RNM holds the multiplier for the MYP instruction and the divisor for the DIV instruction.

The expected and actual results of each instruction test are also stored in memory in fixed locations. Figure EAU-1 shows where they are stored.

To check an instruction using specific arguments (supplied by the user), the argument storage locations can be set manually through the front panel. Once the arguments have been loaded into core, the diagnostic is run with program option bit 6 set ON. For shift/rotate tests, the shift value is set in SHFT (memory location 117_8).

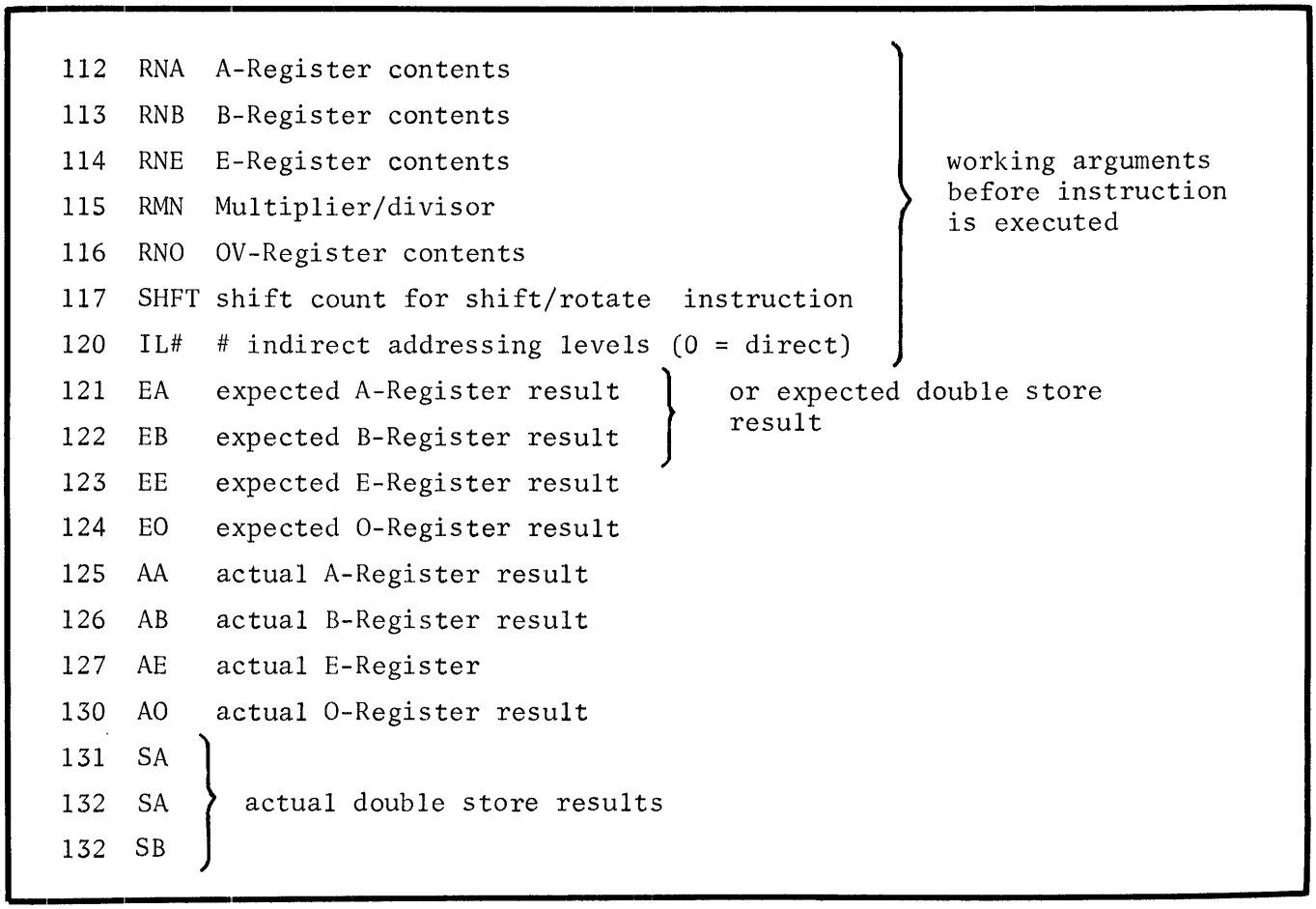

Figure EAU-1. Fixed Core Locations For Arguments And Results

The E-Register (loaded with bit 0 of RNE before the EAU instruction is executed) is not expected to change after execution of the EAU instruction and an error condition occurs if it does not remain intact.

The OV-Register (overflow) is loaded with bit 0 of RNO before execution. The OV-Register is tested after instruction execution in the same way as the E-Register, with the expected results shown in Figure EAU-2.

DST, DSD, RRR, RRL, LSR, LSL	should not change
ASR, MPY	should always be 0
DIV	should be 0 unless a divide error occurred, then it should be 1
ASL	should be 0 unless a significant bit was shifted out, then it should be 1.

Figure EAU-2. OV-Register Expected Results

DIAGNOSTIC MESSAGES

Two kinds of messages, general and error, are typed on the system teleprinter. The general messages are typed only if program option bits 1 and 10 are set OFF. The general messages list as follows:

Message	Description
EAU DIAGNOSTIC	An introductory message printed only at the beginning of the diagnostic.
EOT *x*	This message is printed before an "end-of'test" halt (program option bit 15 set ON). *x* is the octal test number.
END OF PASS *x*	The "end-of-pass" message is printed at the end of a diagnostic cycle where *x* is the decimal pass count.

Error messages are printed if the hardware includes a teleprinter and only if both program option bits 1 and 11 are set OFF. All error messages are prefixed with "E-*x*," where *x* is the octal number of the test where the error occurred. The prefix is followed by the mnemonic instruction name. If the instruction test is in the shift/rotate group, the octal shift count follows; if the instruction was not shift/rotate, the indirect addressing counter, IL = *x*, is indicated, where *x* is the number of indirect addressing levels.

If the register or core results fail to match the expected results, the actual contents and the expected contents are printed. Only registers that failed are printed; all others are assumed correct. For all instruction tests except DLD and DST, the original contents before execution of the B- and A-Registers are listed as part of the error message, whether the registers failed during execution or not. The E- and OV-Registers are never listed unless they failed.

For Example:

E-1 DLD IL=4 B,A = 052352 122516 SB 052352 066421 E=0 SB 1 OV=1 SB 0

This message occurred in test 1 (DLD instruction) with the instruction executed at the fourth indirect addressing level. The B,A,E, and OV results did not compare with the expected results.

E-11 RRR 20 B,A WAS 000007 000003 B,A = 000003 000006 SB 000003 000007

An error occurred in test 11 (RRR instruction) with the shift count set at 20_8. The original B- and A-Register contents are shown, regardless of whether or not they failed (which they did). The current values and the expected values are both shown to indicate that the registers failed.

E-5 ASR 20 B,A WAS 154056 166602 OV=0 SB 1

The error occurred in test 5 (ASR instruction) with the shift count set at 20_8. The B,A Register contents before execution are reported, but they did not fail. Only the OV-Register failed.

The abbreviations used in the above messages are:

Abbreviation	Meaning
A	A-Register
B	B-Register
E	If at beginning, signifies error message; otherwise, E-Register.
OV	OV-Register (overflow indicator)
M	Core memory
SB	Should be
IL	Indirect addressing level

The B-Register contents is always listed before the A-Register contents. Memory, B, and A-Register contents are listed as six octal digits. The octal shift field is two digits.

COMPUTER HALTS

If no teleprinter is available for error reporting and if program option bit 14 is set OFF, the computer halts when an error occurs with the actual register results loaded into the A,B,E and OV-Registers. (For a DST instruction, A,B contain the results actually double stored.) After an error halt, check the memory locations listed in Figure EAU-1 to find the working arguments and the expected results.

Other halts and their causes are listed below in Figure EAU-3.

Memory Data	Meaning
1020*xx*	Error halts (*xx* = test #)
102076	End of test halt (A←B ←test #)
102077	End of pass halt
1060*xx*	Unexpected trap cell halt (*xx* = select code)
107000	Start of configuration halt
107077	End of configuration halt

Figure EAU-3. Computer Halts

Table EAU-1

Program Options--SWITCH REGISTER Settings

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits	Function
0	Set ON to use the external hardware SWITCH REGISTER settings. Set OFF to use the internal switch register settings.
1	Set ON to indicate that no teleprinter is available. Set OFF to indicate that a teleprinter is being used.
2-5	Spares
6	Set ON to suppress the number generator at the beginning of each instruction test. Set OFF to generate new arguments for each execution of an instruction test.
7	Set ON to suppress the indirect addressing portions of the instruction tests. Set OFF to execute the indirect addressing portions of the instruction tests.
8	Set ON to repeat the current test with new arguments if bit 6 is set OFF.
9	Set ON to break out of any test which finds an error.
10	Set ON to suppress non-error messages.
11	Set ON to suppress error messages.
12	Set ON to halt the program at the end of a pass. (MEMORY DATA Register = 102077_8.)
13	Set ON to repeat the current test with the same arguments each time.
14	Set ON to suppress halting the program upon finding an error.
15	Set ON to halt the program at the end of a test. (MEMORY DATA Register = 102076_8 and A- and B-Registers contain the octal test number.)

OPERATING INSTRUCTIONS

a. If a configured version of the diagnostic is available, load it with the Basic Binary Loader and skip to step e; otherwise, if a teleprinter is included in the hardware, load and configure the SIO teleprinter driver.

b. Load the binary object tape using the Basic Binary Loader.

c. If an internal switch register is to be used, then configure the diagnostic as follows:

   1. LOAD ADDRESS 2_8.

   2. Press RUN. The computer halts with 107000_8 in the MEMORY DATA Register.

   3. Set the SWITCH REGISTER according to Table EAU-1.

   4. Press RUN. The computer halts with 107077_8 in the MEMORY DATA Register. The internal switch register now contains the desired options.

d. If desired, use the SIO System Dump program to punch a copy of the configured diagnostic (only if the teleprinter driver was loaded).

e. LOAD ADDRESS 100_8.

f. Set the SWITCH REGISTER for any desired options according to Table EAU-1.

g. Press RUN.

   1. If program option bits 1 and 10 are both set OFF, the diagnostic types a preamble message and starts the first test.

   2. To access a specific test, set program option bit 15 ON. The program halts at the end of each test with 102076_8 in the MEMORY DATA Register and the test number in the A- and B-Registers. Press RUN to continue to the next test. If necessary, set program option bits 11 and 14 ON to suppress error halts and error messages until the desired test is reached. When the desired test is reached, set program option bit 15 OFF, and set program option bit 8 ON. The test is looped with new arguments each time.

3. Normally, the program should be run with all program option bits set OFF. If an error is found, the program prints a message. If no error is found, the program cycles once approximately every minute. If errors do occur, set the program option bits according to Table EAU-1.

h. To reconfigure the internal switch register, LOAD ADDRESS 111_8 and perform steps c-2 through c-4.

APPENDIX A
DIAGNOSTIC CONTROL

The following is a description of diagnostic control within the DLD, DST and ASR tests. Control for the MPY and DIV tests is similar to the DLD test. The ASR test is representative of all shift/rotate instruction tests.

DLD Test

DLD Tests the "double load" instruction as follows:

a. If program option bit 6 is set OFF, the program generates new arguments. If program option bit 6 is set ON, then previously generated arguments are used.

b. The indirect addressing level counter is cleared.

c. The expected results (obtained by the non-EAU routine) are computed for later comparison.

d. The actual EAU instruction is modified to reflect the level of indirect addressing (level 0 is direct).

e. The E- and OV-Registers are loaded with bit 0 of generated arguments RNE and RNO respectively.

f. The A- and B-Registers are cleared.

g. The DLD EAU instruction is executed and the expected results compared with the actual results.

h. If no errors exist, the program continues with step i; if an error is found, DLD test performs the following:

    1. If program option bit 1 is set OFF (indicating that a teleprinter is available) and if program option bit 11 is set OFF (indicating that error messages are to be printed), then an error message is printed on the teleprinter. If either program option bit 1 or 11 is set ON, the error message is bypassed.

2. If the program option bit 14 is set ON, the program jumps to step 3. If program option bit 14 is set OFF, then the program halts. Press RUN to continue the program.

3. If program option bit 9 is set ON, the program jumps to step k. Otherwise, the program continues below.

i. If program option bit 13 is set ON, the program loops back to step e. Otherwise, it continues below.

j. If program option bit 7 is set ON, the indirect addressing tests are skipped. If program option bit 7 is set OFF, then the program increments the indirect addressing level counter. If the counter value is less than 5, the program loops back to step d. If the counter is equal to 5, then the program continues below.

k. If bit 15 is set OFF the program jumps to step 1 below; otherwise, the following occurs:

1. If option bits 1 and 10 are set OFF, the "EOT 1" message is printed. If either program option bit 1 or 10 is set ON, the message is bypassed.

2. The program halts with 102076_8 in the MEMORY DATA Register. The A- and B-Registers contain the test number (1). Press RUN to continue.

1. If program option bit 8 is set OFF, the program continues on to the next EAU instruction test. If program option bit 8 is set ON, the program loops back to step a.

DST Test

DST tests the "double store" instruction as follows:

a. If program option bit 6 is set OFF, new arguments are generated. If program option bit 6 is set ON, then previously generated arguments are used.

b. The indirect addressing level counter is cleared.

c. The EAU instruction is modified to reflect the level of indirect addressing (level 0 is direct).

d. The expected results (obtained by the non-EAU routine) are computed and stored.

e. The DST EAU instruction is executed and the expected results compared with the actual results.

f. If no errors exist, the program jumps to step g. If errors are found, DST performs the following:

   1. If program option bits 1 and 11 are both set OFF, an error message is printed. If either bit 1 or 11 is set ON, the message is suppressed.

   2. If program option bit 14 is set OFF, the computer halts with 102002_8 in the MEMORY DATA Register. The program continues when the operator presses RUN. If program option bit 14 is set ON, the program does not halt.

   3. If program option bit 9 is set ON, the program jumps to step i. If program option bit 9 is set OFF, then the program continues below.

g. If program option bit 13 is set ON, then the program repeats the test using the same arguments (loops to step d). If program option bit 13 is set OFF, the program continues below.

h. If program option bit 7 is set ON, the indirect addressing tests are skipped. If program option bit 7 is set OFF, the indirect addressing level counter is incremented. If the counter value is less than 5, the program loops back to step c.

i. If program option bit 15 is set OFF, then the program jumps to step j; otherwise, the following occurs:

   1. If program option bits 1 and 10 are both set OFF, then "EOT 2" message is printed. If either program option bit 1 or 10 is set ON, the message is bypassed.

   2. The program halts with 102076_8 in the MEMORY DATA Register. The A- and B-Registers contain the test number (2). Press RUN to continue.

j. If program option bit 8 is set OFF, the program continues on to the next instruction test. If program option bit 8 is set ON, the program loops back to step a.

ASR Test

ASR tests the "arithmetic long shift right" instruction as follows:

a. If program option bit 6 is set OFF, new arguments are generated. If program option bit 6 is set ON, then previously generated arguments are used.

b. The shift counter is cleared if switch 6 was OFF.

c. The non-EAU subroutine is executed to obtain the expected results.

d. The actual EAU instruction is modified to reflect the shift level.

e. The registers are initialized.

f. The EAU instruction is executed and the expected results compared with the actual results.

g. If no errors exist, the program jumps to step h.
If an error is found, ASR performs the following:

   1. If program option bits 1 and 11 are both set OFF, then an error message is printed on the teleprinter. If either program option bit 1 or 11 is set ON, the message is bypassed.

   2. If program option bit 14 is set ON, the program jumps to step 3. If program option bit 14 is set OFF, the program halts with 102005_8 in the MEMORY DATA Register. The program continues when the operator presses RUN.

   3. If program option bit 9 is set ON, the program jumps to step j. Otherwise, it continues below.

h. If program option bit 13 is set ON, the program loops back to step d. Otherwise, it continues below.

i. If program option bit 6 was ON at the beginning of the instruction test, the program skips to step j. Otherwise, the program increments the shift counter and checks to see if the counter value is equal to 20. If it is less than 20, the program loops back to step c. Otherwise, it continues below.

j. If program option bit 15 is set OFF, the program skips to step k; otherwise, the following occurs:

   1. If program option bits 1 and 10 are OFF, the "EOT 5" message is printed. Otherwise, the message is bypassed.

   2. The program halts with 102076_8 in the MEMORY DATA Register. The A and B registers contain the test number (5). Press RUN to continue.

k. If program option bit 8 is set OFF, the program continues on to the next test. If program option bit 8 is set ON, the program loops back to step a.

EXTENDED ARITHMETIC UNIT DIAGNOSTIC
(REV. B)

BINARY TAPE	24186-60001
SOURCE TAPES	24186-80001 24186-80002
SOURCE LISTING	24186-90001

```
0001                    ASMB,A,B,L,T,C
DLD       004200
DST       004400
MPY       000200
DIV       000400
ASR       001020
ASL       000020
LSR       001040
LSL       000040
RRR       001100
RRL       000100
LIST      000102
EXEC      000107
A         000000
B         000001
CRLF      006412
RNA       000112
RNB       000113
RNE       000114
RNM       000115
RNO       000116
SHFT      000117
IL#       000120
EA        000121
EB        000122
EE        000123
EO        000124
AA        000125
AB        000126
AE        000127
AO        000130
SA        000131
SB        000132
DSA       000133
FLG6      000134
ISR       000135
LCNT      000136
REI       000137
S6        000140
S6CNT     000141
TST#      000142
T1        000143
T2        000144
T3        000145
APT       000146
DAD       000147
B7        000154
B10       000155
B11       000156
B12       000157
B17       000160
B20       000161
B30       000162
B60       000163
B72       000164
B77       000165
```

```
B366     000166
B377     000167
BIT0     000170
BIT6     000171
BIT7     000172
BIT9     000173
BIT12    000174
BIT15    000175
M1       000176
M5       000177
M6       000200
M15      000201
.1       000170
.2       000202
.3       000203
.4       000204
.5       000205
.6       000206
.7       000154
.8       000155
.10      000157
.11      000207
.16      000161
.20      000210
.38      000211
.63      000165
.1500    000212
A00      000213
A2S      000214
A2002    000215
A4002    000216
D5000    000217
D7400    000220
HLT1     000221
HLT77    000222
ASR0     000223
ASL0     000224
LSR0     000225
LSL0     000226
RRR0     000227
RRL0     000230
AMWMA    000231
BAMA     000232
BAWMA    000233
MIA      000234
MA77     000235
MA77B    000236
RNAA     000237
RNMA     000240
STRMA    000241
MI       000242
MG77     000252
MG77A    000260
MG77B    000263
BAM      000264
AMWM     000265
```

```
BAWM     000271
STRM     000275
ALT      000303
DBUF     000330
BPT      000331
BCNT     000332
BUF      000333
RNG      000377
RNGBC    000434
RNG1     000445
RNG2     000454
RNG3     000463
RNBA     000472
RN1      000500
RN2      000501
GBR      000502
CEM      000507
CEM1     000545
EDMA     000547
EDM      000550
CNEM     000551
CIA      000557
CIA1     000567
C88      000571
C89      000602
C813     000610
CEHLT    000616
CEHT1    000625
EOT      000636
EOT1     000644
EOT2     000655
EOTMA    000660
EOTM     000661
IREG     000663
IREGO    000677
CHECK    000705
CBUF     000737
PBUF     000745
PAK      000752
..B      000767
MOVE     000770
MOVEC    000774
BPC1     001005
CUTOX    001011
CUTO6    001015
CUTO     001021
CUTO1    001026
PIL#     001045
ILMA     001054
ILM      001055
PBB      001057
SBMA     001064
SBM      001066
PBA      001067
BAQMA    001113
BAQM     001114
```

```
PEO       001116
PE01      001135
EQMA      001154
OQMA      001155
EQM       001156
OQM       001157
CEOB      001161
.MPY      001173
.MPY1     001212
.MPY2     001225
.DIV      001230
LOOP      001256
OVFLO     001303
NOV       001304
.DIV1     001307
COUNT     000143
MIND      000144
.ASR      001312
.ASR1     001316
.ASL      001326
.ASL1     001330
.ASL2     001340
ASLOV     001351
.LSR      001354
.LSR1     001356
.LSL      001365
.LSL1     001367
.RRR      001376
.RRR1     001400
.RRL      001410
.RRL1     001412
IISR      001422
ISRT      001432
ISRT1     001443
SRMP      001447
CSHFT     001465
CSHF2     001474
CONFG     001476
START     001502
ST1       001505
SOP       001525
AGAIN     001527
DLDL      001533
DLDC      001544
DLDR      001547
DLDA      001554
DLD2      001567
DLD1      001573
DLDE      001577
TST2      001603
DSTL      001605
DSTC      001612
DSTR      001620
DSTA      001625
DST3      001657
DST2      001661
```

```
DST1     001665
DSTE     001671
TST3     001675
MPYL     001677
MPYC     001705
MPYR     001710
MPYA     001713
MPY2     001735
MPY1     001741
MPYE     001745
PG1      001777
TST4     001777
DIVL     002001
DIVC     002007
DIVR     002012
DIVA     002015
DIV2     002041
DIV1     002046
DIVE     002051
TST5     002055
ASRL     002057
ASRC     002060
ASRR     002064
ASRI     002066
ASR1     002100
ASRE     002104
TST6     002110
ASLL     002112
ASLC     002113
ASLR     002117
ASLI     002121
ASL1     002133
ASLE     002137
TST7     002143
LSRL     002145
LSRC     002146
LSRR     002152
LSRI     002154
LSR1     002166
LSRE     002172
TST10    002176
LSLL     002200
LSLC     002201
LSLR     002205
LSLI     002207
LSL1     002221
LSLE     002225
TST11    002231
RRRL     002233
RRRC     002234
RRRR     002240
RRRI     002242
RRR1     002254
RRRE     002260
TST12    002264
RRLL     002266
```

```
RRLC     002267
RRLR     002273
RRLI     002275
RRL1     002307
RRLE     002313
EOP      002330
NEXT2    002331
DONE     002355
EOP1     002362
EXRTN    002363
END      002367
**   NO ERRORS*
```

```
0001                    ASMB,A,B,L,T,C
0003  00002                     ORG 2
0004                            SUP             SUPPRESS EXTENDED STRINGS
0005*
0006* A TELETYPE OR LIST DEVICE IS OPTIONAL - BUT RECOMMENDED
0007*
0008* SOFTWARE REQUIREMENTS ARE:
0009*  BINARY OBJECT TAPE
0010*  SIO DRIVER FOR LIST DEVICE (OPTIONAL)
0011*
0012* THE CONFIGURATION ADDRESS IS OCTAL 2 OR OCTAL 111.
0013* THE DIAGNOSTIC STARTS AT OCTAL LOCATION 100
0014*
0015* NOTE - RUN THIS DIAGNOSTIC ONLY AFTER THE FOLLOWING HAVE BEEN
0016*  SUCESSFULLY EXECUTED:
0017*    ALTER-SKIP INSTRUCTION TEST
0018*    MEMORY REFERENCE INSTRUCTION TEST
0019*    SHIFT ROTATE INSTRUCTION TEST
0020*    DIAGNOSTIC FOR LIST DEVICE (IF A LIST DEVICE IS TO BE USED)
0021*
0022*  COMPUTER HALTS
0023* T=102000  ERROR HALTS  (XX=TEST#)
0024* T=102076  END OF TEST HALT (A=B=TEST#)
0025* T=102077  END OF PASS HALT
0026* T=106000  UNEXPECTED TRAP CELL HALT (XX = SELECT CODE)
0027* T=107000  START OF CONFIGURATION HALT
0028* T=107077  END OF CONFIGURATION HALT
0029*
0030* SWITCH SETTINGS
0031*
0032*    SWITCH    MEANING IF SET OR UP
0033*      15        HALT AT END OF TEST  (A=B=TEST#)
0034*      14        SUPPRESS ERROR HALTS
0035*      13        REPEAT LAST TEST WITH SAME VARIABLES
0036*      12        HALT AT END OF PASS (HLT 77)
0037*      11        SUPPRESS ERROR MESSAGES
0038*      10        SUPPRESS NON-ERROR MESSAGES
0039*       9        BREAKOUT OF TEST ON ERROR
0040*       8        LOOP IN TEST - NEW DATA IF SW. 6 IS DOWN
0041*       7        SKIP INDIRECT ADDRESSING TESTS
0042*       6        SUPPRESS THE NUMBER GENERATOR
0043*       5
0044*       4
0045*       3
0046*       2
0047*       1        INDICATES THAT NO LIST DEVICE IS AVAILABLE
0048*       0        USE EXTERNAL SWITCH REGISTER
```

```
0050* EAU MACRO CODES WITHOUT BIT 15
0051*
0052  04200          DLD   EQU 4200B      DOUBLE LOAD
0053  04400          DST   EQU 4400B      DOUBLE STORE
0054  00200          MPY   EQU 200B       MULTIPLY
0055  00400          DIV   EQU 400B       DIVIDE
0056  01020          ASR   EQU 1020B      ARITHMETIC SHIFT RIGHT 16
0057  00020          ASL   EQU 20B        ARITHMETIC SHIFT LEFT 16
0058  01040          LSR   EQU 1040B      LOGICAL SHIFT RIGHT 16
0059  00040          LSL   EQU 40B        LOGICAL SHIFT LEFT 16
0060  01100          RRR   EQU 1100B      ROTATE RIGHT 16
0061  00100          RRL   EQU 100B       ROTATE LEFT 16



0063  00002 025476         JMP CONFG      GO CONFIGURE
0064*
0065  00100                ORG 100B
0066  00100 025502         JMP START      GO RUN EAU DIAGNOSTIC
0067*
0068  00102          LIST  EQU 102B       LINK TO SIO LIST DRIVER
0069  00107          EXEC  EQU 107B       LINK TO DIAGNOSTIC MONITOR
0070  00000          A     EQU 0
0071  00001          B     EQU 1
0072  06412          CRLF  EQU 6412B      ASCII CARRIAGE RETURN/LINE FEED
0073*
0074  00105                ORG 105B
0075  00105 002367         DEF END        LAST WORD ADDRESS - FOR SIO DUMP
0076*
0077  00107                ORG 107B
0078  00107 002363         DEF EXRTN      RETURN ADDRESS FROM EXEC CALL
0079*
0080  00110                ORG 110B
0081  00110 000000         NOP            SECONDARY ENTRY POINT - NOT USED
0082  00111 025476         JMP CONFG      GO CONFIGURE DIAGNOSTIC
```

```
0084   00112 000000  RNA    NOP            WORKING ARGUMENTS OR OPERANDS
0085   00113 000000  RNB    NOP
0086   00114 000000  RNE    NOP
0087   00115 000000  RNM    NOP
0088   00116 000000  RNO    NOP
0089   00117 000000  SHFT   NOP            INSTRUCTION SHIFT FIELD
0090   00120 000000  IL#    NOP            # INDIRECT ADDRESSING LEVELS
0091   00121 000000  EA     NOP            EXPECTED REGISTER RESULTS: A REG
0092   00122 000000  EB     NOP                                       B REG
0093   00123 000000  EE     NOP                                       E REG
0094   00124 000000  EO     NOP                                       O REG
0095   00125 000000  AA     NOP            ACTUAL REGISTER RESULTS:  A REG
0096   00126 000000  AB     NOP                                      B REG
0097   00127 000000  AE     NOP                                      E REG
0098   00130 000000  AO     NOP                                      O REG
0099   00131 000000  SA     NOP            CONTENTS OF DST MACRO
0100   00132 000000  SB     NOP            CONTENTS OF DST MACRO
0101*
0102   00133 000131  DSA    DEF SA
0103   00134 000000  FLG6   NOP            0 MEANS SW. 6 DOWN DURING LOOP
0104   00135 000000  ISR    NOP            INTERNAL SWITCH REGISTER
0105   00136 000000  LCNT   NOP            COUNT OF LOOPS WITH SW. 6 DOWN
0106   00137 000000  REI    NOP            REGISTER ERROR INDICATOR
0107   00140 000000  S6     NOP            STATE OF SWITCH 6
0108   00141 177777  S6CNT  OCT 177777     COUNTER USED WITH SWITCH 6
0109   00142 000000  TST#   NOP            TEST (ERROR) #
0110   00143 000000  T1     NOP            TEMPORARY STORAGE
0111   00144 000000  T2     NOP            TEMPORARY STORAGE
0112   00145 000000  T3     NOP            TEMPORARY STORAGE
0113*
0114   00146 000147  APT    DEF DAD
0115   00147 000000  DAD    NOP            DIRECT ADDRESS
0116   00150 100147         DEF *-1,I      1 INDIRECT LEVEL
0117   00151 100150         DEF *-1,I      2 INDIRECT LEVELS
0118   00152 100151         DEF *-1,I      3 INDIRECT LEVELS
0119   00153 100152         DEF *-1,I      4 INDIRECT LEVELS
0120*
0121   00154 000007  B7     OCT 7
0122   00155 000010  B10    OCT 10
0123   00156 000011  B11    OCT 11
0124   00157 000012  B12    OCT 12
0125   00160 000017  B17    OCT 17
0126   00161 000020  B20    OCT 20
0127   00162 000030  B30    OCT 30
0128   00163 000060  B60    OCT 60         ASCII 0
0129   00164 000072  B72    OCT 72
0130   00165 000077  B77    OCT 77
0131   00166 000366  B366   OCT 366
0132   00167 000377  B377   OCT 377
0133*
0134   00170 000001  BIT0   OCT 1
0135   00171 000100  BIT6   OCT 100
0136   00172 000200  BIT7   OCT 200
0137   00173 001000  BIT9   OCT 1000
0138   00174 010000  BIT12  OCT 10000
0139   00175 100000  BIT15  OCT 100000
```

```
0140*
0141  00176 177777  M1     DEC -1
0142  00177 177773  M5     DEC -5
0143  00200 177772  M6     DEC -6
0144  00201 177760  M16    DEC -16
0145*
0146  00170           .1     EQU BIT0
0147  00202 000002  .2     DEC 2
0148  00203 000003  .3     DEC 3
0149  00204 000004  .4     DEC 4
0150  00205 000005  .5     DEC 5
0151  00206 000006  .6     DEC 6
0152  00154           .7     EQU B7
0153  00155           .8     EQU B10
0154  00157           .10    EQU B12
0155  00207 000013  .11    DEC 11
0156  00161           .16    EQU B20
0157  00210 000024  .20    DEC 20
0158  00211 000046  .38    DEC 38
0159  00165           .63    EQU B77
0160  00212 002734  .1500  DEC 1500
0161*
0162  00213 030060  A00    ASC 1,00        2 ASCII 0'S
0163  00214 020040  A2S    ASC 1,          2 ASCII SPACES
0164*
0165  00215 002002  A2002  OCT 2002
0166  00216 004002  A4002  OCT 4002
0167  00217 035000  D5000  OCT 35000
0168  00220 037400  D7400  OCT 37400
0169*
0170  00221 106001  HLT1   OCT 106001
0171  00222 106077  HLT77  OCT 106077
0172*
0173  00223 101020  ASR0   DEF ASR,I      ASR WITHOUT SHIFT FIELD
0174  00224 100020  ASL0   DEF ASL,I      ASL WITHOUT SHIFT FIELD
0175  00225 101040  LSR0   DEF LSR,I      LSR WITHOUT SHIFT FIELD
0176  00226 100040  LSL0   DEF LSL,I      LSL WITHOUT SHIFT FIELD
0177  00227 101100  RRR0   DEF RRR,I      RRR WITHOUT SHIFT FIELD
0178  00230 100100  RRL0   DEF RRL,I      RRL WITHOUT SHIFT FIELD
0179*
0180  00231 000266  AMWMA  DEF AMWM
0181  00232 000264  BAMA   DEF BAM
0182  00233 000271  BAWMA  DEF BAWM
0183  00234 000242  MIA    DEF MI
0184  00235 000252  MA77   DEF MG77
0185  00236 000263  MA77B  DEF MG77B
0186  00237 000112  RNAA   DEF RNA
0187  00240 000115  RNMA   DEF RNM
0188  00241 000275  STRMA  DEF STRM
0189*
0190  00242 006412  MI     ABS CRLF
0191  00243 042501          ASC 7,EAU DIAGNOSTIC
0192*
0193  00252 042516  MG77   ASC 6,END OF PASS
0194  00260 054130  MG77A  ASC 3,XXXXXX
0195  00263 054130  MG77B  ASC 1,XX
```

```
0196*
0197   00264 041054   BAM    ASC 1,B,
0198   00265 040454   AMWM   ASC 4,A,M WAS
0199*
0200   00271 041054   BAWM   ASC 4,B,A WAS
0201*
0202   00275 051524   STRM   ASC 6,STORED WAS



0204* ASCII LABLE TABLE
0205*
0206   00303 000302   ALT    DEF *-1
0207   00304 042114          ASC 2,DLD       1
0208   00306 042123          ASC 2,DST       2
0209   00310 046520          ASC 2,MPY       3
0210   00312 042111          ASC 2,DIV       4
0211   00314 040523          ASC 2,ASR       5
0212   00316 040523          ASC 2,ASL       6
0213   00320 046123          ASC 2,LSR       7
0214   00322 046123          ASC 2,LSL       8
0215   00324 051122          ASC 2,RRR       9
0216   00326 051122          ASC 2,RRL       10



0218   00330 000333   DBUF   DEF BUF         BUFFER ADDRESS
0219   00331 000000   BPT    NOP             BUFFER POINTER
0220   00332 000000   BCNT   NOP             BUFFER CHARACTER COUNT
0221   00333 000000   BUF    BSS 36          72 CHARACTER BUFFER
```

```
0223* SUBROUTINE TO GENERATE NEW ARGUMENTS
0224*
0225* IF SWITCH 6 IS DOWN, GENERATE NEW NUMBERS AND PUT THEM IN
0226*  RNA, RNB, RNM, RNE, & RNO.
0227*   IF LCNT=0,  RNM←000000 & RNA←NEGATIVE
0228*   IF LCNT = 1,  RNB←000000, RNA←100000, RNM←000001
0229*   IF LCNT = 2,  RNB←000000, RNA←100000, RNM←177777
0230*   IF LCNT=63,127,191,...,  RNB←177777
0231*   IF LCNT=64,128,192,...,  RNB←000000
0232*   (IF SWITCH 6 IS UP, SUBSTITUTE S6CNT FOR LCNT)
0233*    RETURN P+1
0234* IF SWITCH 6 IS UP, FLG6 ← NON 0, RETURN P+1
0235*
0236  00377 000000  RNG    NOP
0237  00400 014502         JSB GSR        GET SWITCH REGISTER
0238  00401 010171         AND BIT6        CHECK SWITCH 6
0239  00402 002002         SZA
0240  00403 070134         STA FLG6         UP = FLG6 ← NON 0
0241  00404 002002         SZA
0242  00405 124377         JMP RNG,I        UP = RETURN P+1
0243  00406 060500         LDA RN1
0244  00407 040501         ADA RN2
0245  00410 070500         STA RN1        SAVE NEW BASE
0246  00411 070112         STA RNA        NEW RNA
0247  00412 001700         ALF
0248  00413 070113         STA RNB        NEW RNB
0249  00414 001700         ALF
0250  00415 070115         STA RNM        NEW RNM
0251  00416 001200         RAL
0252  00417 010170         AND BIT0
0253  00420 070114         STA RNE        NEW RNE (BIT 0 ONLY)
0254  00421 060115         LDA RNM
0255  00422 001222         RAL,RAL
0256  00423 010170         AND BIT0
0257  00424 070116         STA RNO        NEW RNO (BIT 0 ONLY)
0258  00425 014571         JSB C6B        CHECK SWITCH 6
0259  00426 002001         RSS
0260  00427 024434         JMP RNG6C
0261  00430 060141         LDA S6CNT        SET:  A←S6CNT←S6CNT+1
0262  00431 002004         INA
0263  00432 070141         STA S6CNT
0264  00433 002001         RSS
0265  00434 060136  RNG6C  LDA LCNT         CLEAR:  A←LCNT
0266  00435 002002         SZA
0267  00436 024445         JMP RNG1
0268  00437 070115         STA RNM        LCNT=0,  RNM←000000
0269  00440 064112         LDB RNA
0270  00441 006021         SSB,RSS
0271  00442 007000         CMB
0272  00443 074112         STB RNA        LCNT=0,  RNA←NEGATIVE
0273  00444 124377         JMP RNG,I
0274  00445 050170  RNG1   CPA .1
0275  00446 002001         RSS
0276  00447 024454         JMP RNG2     LCNT NOT 1
0277  00450 014472         JSB RNBA       RNB,RNA ← 000000,100000
0278  00451 002404         CLA,INA
```

```
0279   00452 070115          STA RNM         RNM ← 000001
0280   00453 124377          JMP RNG,I
0281   00454 050202  RNG2    CPA .2
0282   00455 002001          RSS
0283   00456 024463          JMP RNG3        LCNT NOT 2
0284   00457 014472          JSB RNBA        RNB,RNA ← 000000,100000
0285   00460 003400          CCA
0286   00461 070115          STA RNM         RNM ← 177777
0287   00462 124377          JMP RNG,I
0288   00463 010165  RNG3    AND B77
0289   00464 002003          SZA,RSS         IF LCNT = 64,128,192,...
0290   00465 070113          STA RNB           RNB ← 000000
0291   00466 007400          CCB
0292   00467 050165          CPA B77         IF LCNT = 63,127,191,...
0293   00470 074113          STB RNB           RNB ← 177777
0294   00471 124377          JMP RNG,I
0295*
0296   00472 000000  RNBA    NOP
0297   00473 006400          CLB
0298   00474 074113          STB RNB         RNB←000000
0299   00475 064175          LDB BIT15
0300   00476 074112          STB RNA         RNA←100000
0301   00477 124472          JMP RNBA,I
0302*
0303   00500 143703  RN1     DEC -14397      BASE
0304   00501 034075  RN2     DEC 14397       CONSTANT



0306* SUBROUTINE TO GET SWITCH REGISTER IN A
0307*
0308   00502 000000  GSR     NOP
0309   00503 102501          LIA 1           START WITH EXTERNAL SWITCH REG.
0310   00504 002011          SLA,RSS         CHECK SWITCH 0
0311   00505 000135          LDA ISR         IF CLEAR, REPLACE WITH INTERNAL
0312   00506 124502          JMP GSR,I       RETURN P+1 WITH B INTACK



0314* SUBROUTINE TO CHECK FOR AN ERROR MESSAGE
0315*  RETURN P+1 FOR NO MESSAGE - SWITCH  1 OR 11 UP
0316*  RETURN P+2 FOR AN ERROR MESSAGE - BOTH SWITCHES 1 AND 11 DOWN
0317*
0318* BEFORE RETURNING FOR AN ERROR MESSAGE, INITIALIZE THE MESSAGE BUFF
0319* PACK E-TST#, MACRO NAME & SHFT (IF A SHIFT ROTATE INSTR.)
0320*
0321   00507 000000  CEM     NOP
0322   00510 014502          JSB GSR         GET SWITCH REGISTER
0323   00511 010216          AND A4002         CHECK SWITCHES 1 AND 11
0324   00512 002002          SZA
0325   00513 124507          JMP CEM,I       EITHER UP - RETURN P+1
0326   00514 034507          ISZ CEM         SET TO RETURN P+2
0327   00515 014737          JSB CBUF        BOTH DOWN - INITIALIZE MESSAGE
0328   00516 050202          LDA .2
0329   00517 064547          LDB EDMA
0330   00520 014770          JSB MOVE        PAK "E-"
```

```
0331   00521 064142          LDB TST#
0332   00522 015011          JSB CUTOX      CONVERT & PAK TST#
0333   00523 064142          LDB TST#
0334   00524 005000          BLS
0335   00525 044303          ADB ALT
0336   00526 060004          LDA .4
0337   00527 014770          JSB MOVE       PAK MACRO NAME, SPACE
0338   00530 064142          LDB TST#
0339   00531 044177          ADB M5
0340   00532 006020          SSB
0341   00533 124507          JMP CEM,I      NOT SHIFT/ROTATE
0342   00534 064117          LDB SHFT       SHIFT COUNT
0343   00535 006003          SZB,RSS
0344   00536 044161          ADB .16        MAKE 16 IF 0
0345   00537 060162          LDA B30
0346   00540 010001          AND B
0347   00541 002002          SZA
0348   00542 024545          JMP CEM1
0349   00543 000163          LDA B60        PAK A LEADING 0 IF SHIFT COUNT
0350   00544 014752          JSB PAK          IS < 10 OCTAL
0351   00545 015011  CEM1    JSB CUTOX      CONVERT AND PAK
0352   00546 124507          JMP CEM,I      RETURN P+2
0353*
0354   00547 000550  EDMA    DEF EDM
0355   00550 042455  EDM     ASC 1,E-


0357* SUBROUTINE TO CHECK FOR AN NON-ERROR MESSAGE
0358*  RETURN P+1 FOR NO MESSAGE - SWITCH 1 OR 10 UP
0359*  RETURN P+2 FOR MESSAGE - BOTH SWITCHES 1 AND 10 DOWN
0360*
0361   00551 000000  CNEM    NOP
0362   00552 014502          JSB GSR        GET SWITCH REGISTER
0363   00553 010215          AND A2002        CHECK SWITCHES 1 AND 10
0364   00554 002003          SZA,RSS
0365   00555 034551          ISZ CNEM       BOTH DOWN - RETURN P+2
0366   00556 124551          JMP CNEM,I     EITHER UP - RETURN P+1


0368* SUBROUTINE TO CHECK FOR NEXT INDIRECT LEVEL ADDRESSING
0369*  RETURN P+2 IF SWITCH 7 IS UP
0370*  ELSE, STEP IL#; IF 5, RETURN P+2; ELSE RETURN P+1
0371* RETURN WITH IL# IN A
0372*
0373   00557 000000  CIA     NOP
0374   00560 014502          JSB GSR
0375   00561 010172          AND BIT7
0376   00562 002002          SZA            CHECK SWITCH 7
0377   00563 024567          JMP CIA1         UP - RETURN P+2
0378   00564 034120          ISZ IL#
0379   00565 060120          LDA IL#          DOWN -  IL# + IL# +1
0380   00566 050205          CPA .5
0381   00567 034557  CIA1    ISZ CIA          IL# = 5,  RETURN P+2
0382   00570 124557          JMP CIA,I      ELSE RETURN P+1
```

```
0384* SUBROUTINE TO CHECK SWITCH 8 TO LOOP ON TEST
0385*  IF SET:  RETURN P+1
0386*  IF CLEAR:  S8CNT+177777, RETURN P+2
0387*
0388   00571 000000  CS8    NOP
0389   00572 014502         JSB GSR
0390   00573 001727         ALF,ALF        CHECK SWITCH 8
0391   00574 000010         SLA
0392   00575 124571         JMP CS8,I        SET:  RETURN P+1
0393   00576 003400         CCA            CLEAR:
0394   00577 070141         STA S8CNT
0395   00600 034571         ISZ CS8          S8CNT+177777 (RESET)
0396   00601 124571         JMP CS8,I        RETURN P+2



0398* SUBROUTINE TO CHECK SWITCH 9 FOR AN ERROR BREAKOUT
0399*
0400   00602 000000  CS9    NOP
0401   00603 014502         JSB GSR
0402   00604 010173         AND BIT9       CHECK SWITCH 9
0403   00605 002003         SZA,RSS          UP = RETURN P+1
0404   00606 034602         ISZ CS9          DOWN = RETURN P+2
0405   00607 124602         JMP CS9,I



0407* SUBROUTINE TO CHECK SWITCH 13 TO REPEAT A MACRO CALL
0408*
0409   00610 000000  CS13   NOP
0410   00611 014502         JSB GSR
0411   00612 001222         RAL,RAL        CHECK SWITCH 13
0412   00613 002021         SSA,RSS          UP = RETURN P+1
0413   00614 034610         ISZ CS13         DOWN = RETURN P+2
0414   00615 124610         JMP CS13,I



0416* SUBROUTINE TO CHECK FOR AN ERROR HALT
0417*  IF SWITCH 14 IS DOWN, RESTORE THE ACTUAL REGISTERS AND RETURN P+2
0418*   FOR AN ERROR HALT; ELSE RETURN P+2
0419*
0420   00616 000000  CEHLT  NOP
0421   00617 014502         JSB GSR        GET SWITCH REGISTER
0422   00620 001200         RAL
0423   00621 002021         SSA,RSS
0424   00622 024625         JMP CEHT1
0425   00623 034616         ISZ CEHLT      SWITCH 14 WAS UP
0426   00624 124616         JMP CEHLT,I      RETURN P+2
0427   00625 060127  CEHT1  LDA AE
0428   00626 000003         CLE,RAR        RESTORE E
0429   00627 103101         CLO
0430   00630 060130         LDA AO
0431   00631 000010         SLA
0432   00632 102101         STO            RESTORE OV
0433   00633 060126         LDA AA         RESTORE A
```

```
0434  00634 064126          LDB AB          RESTORE B
0435  00635 124616          JMP CEHLT,I       RETURN P+1



0437* END OF TEST ROUTINE
0438*  CHECK SWITCH 15:  IF DOWN, RETURN P+2
0439*    ELSE CHECK SWITCHES 1&10:  IF BOTH DOWN, PAK AND PRINT
0440*      END OF TEST MESSAGE (EOT XX),
0441*        THEN RETURN P+1 WITH TST# IN BOTH A & B REGISTERS.
0442*
0443  00636 000000  EOT    NOP
0444  00637 014502          JSB GSR
0445  00640 002020          SSA             CHECK SWITCH 15
0446  00641 024644          JMP EOT1
0447  00642 034636          ISZ EOT           DOWN = RETURN P+2
0448  00643 124636          JMP EOT,I
0449  00644 014551  EOT1    JSB CNEM          UP = CHECK FOR NON-ERROR MSG.
0450  00645 024655          JMP EOT2            USER DOESN'T WANT MESSAGE
0451  00646 014737          JSB CBUF        CLEAR BUFFER
0452  00647 060004          LDA .4
0453  00650 064660          LDB EOTMA
0454  00651 014770          JSB MOVE        PAK "EOT" & SPACE
0455  00652 064142          LDB TST#
0456  00653 015011          JSB CUTOX       PAK TST#
0457  00654 014745          JSB PBUF        PRINT EOT MESSAGE
0458  00655 060142  EOT2    LDA TST#
0459  00656 064142          LDB TST#
0460  00657 124636          JMP EOT,I       RETURN P+1, A=B=TST#
0461*
0462  00660 000661  EOTMA  DEF EOTM
0463  00661 042517  EOTM   ASC 2,EOT



0465* SUBROUTINE TO INITIALIZE REGISTERS BEFORE A MACRO CALL
0466* E = EE = RNE(0)
0467* OV = RNO(0)
0468*  A = RNA
0469*  B = RNB
0470* RETURN P+1
0471*
0472  00663 000000  IREG   NOP
0473  00664 103101          CLO
0474  00665 064116          LDB RNO
0475  00666 004010          SLB
0476  00667 102101          STO             OV = RNB(0)
0477  00670 060114          LDA RNE
0478  00671 010170          AND BIT0
0479  00672 070123          STA EE          EE = RNE(0)
0480  00673 001500          ERA             E = RNE(0)
0481  00674 060112          LDA RNA         A = RNA
0482  00675 064113          LDB RNB         B = RNB
0483  00676 124663          JMP IREG,I        RETURN P+1
```

```
0485* GET AND SAVE THE EXPECTED OV REGISTER
0486* THEN CALL IREG TO INITIALIZE THE REST
0487*
0488   00677 000000  IREGO NOP
0489   00700 060116        LDA RNO
0490   00701 010170        AND BIT8
0491   00702 070124        STA EO          EO ← RNO(8)
0492   00703 014663        JSB IREG
0493   00704 124677        JMP IREGO,I



0495* SUBROUTINE TO SAVE THE ACTUAL REGISTER RESULTS
0496*  THEN TO COMPARE THEM WITH THE EXPECTED
0497* RETURN P+1 WITH A & REI INDICATING ANY ERRORS
0498*    BITS(3-0) ← OV,E,B,A ERROR BITS RESPECTIVELY
0499*
0500   00705 000000  CHECK NOP
0501   00706 070125        STA AA          SAVE ACTUAL A
0502   00707 074126        STB AB          SAVE ACTUAL B
0503   00710 002400        CLA             ASSUME NO ERRORS
0504   00711 054122        CPB EB
0505   00712 002001        RSS             B OK
0506   00713 040202        ADA .2            BIT 1 INDICATES B ERROR
0507   00714 064125        LDB AA
0508   00715 054121        CPB EA
0509   00716 002001        RSS             A OK
0510   00717 002004        INA               BIT 0 INDICATES A ERROR
0511   00720 006440        CLB,SEZ
0512   00721 006004        INB
0513   00722 074127        STB AE          SAVE ACTUAL E
0514   00723 054123        CPB EE
0515   00724 002001        RSS             E OK
0516   00725 040204        ADA .4            BIT 2 INDICATES E ERROR
0517   00726 006400        CLB
0518   00727 102201        SOC
0519   00730 006004        INB
0520   00731 074130        STB AO          SAVE ACTUAL OV
0521   00732 054124        CPB EO
0522   00733 002001        RSS             OV OK
0523   00734 040155        ADA .8            BIT 3 INDICATES OV ERROR
0524   00735 070137        STA REI         SAVE REGISTER ERROR INDICATOR
0525   00736 124705        JMP CHECK,I     RETURN P+1
```

```
0527* SUBROUTINE TO CLEAR THE BUFFER
0528*
0529  00737 000000  CBUF  NOP
0530  00740 002400       CLA
0531  00741 070332       STA BCNT       RESET COUNTER
0532  00742 064330       LDB DBUF
0533  00743 074331       STB BPT        RESET POINTER
0534  00744 124737       JMP CBUF,I


0536* SUBROUTINE TO PRINT BUFFER - RETURN P+1
0537*
0538  00745 000000  PBUF  NOP
0539  00746 060332       LDA BCNT       # CHARACTERS
0540  00747 064330       LDB DBUF       BUFFER ADDRESS
0541  00750 114102       JSB LIST,I
0542  00751 124745       JMP PBUF,I     RETURN P+1


0544* SUBROUTINE TO ADD CHARACTER IN A LOWER TO THE BUFFER
0545*  RETURN P+1 WITH B UNCHANGED.
0546*
0547  00752 000000  PAK   NOP
0548  00753 074767       STB ..B        SAVE B REGISTER
0549  00754 010167       AND B377       CLEAR UPPER HALF
0550  00755 064332       LDB BCNT       BUFFER CHARACTER COUNT
0551  00756 006011       SLB,RSS
0552  00757 001737       ALF,SLA,ALF    IF EVEN,ROTATE AND STORE
0553  00760 130331       IOR BPT,I      IF ODD, MERGE AND STORE
0554  00761 170331       STA BPT,I
0555  00762 004010       SLB
0556  00763 034331       ISZ BPT        IF ODD, BUMP BUFFER ADDRESS
0557  00764 034332       ISZ BCNT       STEP BUFFER CHARACTER COUNT
0558  00765 064767       LDB ..B        RESTORE B
0559  00766 124752       JMP PAK,I      RETURN P+1
0560*
0561  00767 000000  ..B   NOP           TEMPORARY STORAGE FOR B REGISTER


0563* SUBROUTINE TO MOVE A STRING OF ASCII CHARACTERS TO THE BUFFER
0564*  GIVEN: CHARACTER COUNT IN A
0565*         STRING ADDRESS IN B
0566* THE 1ST CHARACTER IS ASSUMED IN BITS 15-8 OF THE FIRST WORD
0567* RETURN P+1
0568*
0569  00770 000000  MOVE  NOP
0570  00771 070143       STA T1         # CHARACTERS
0571  00772 074144       STB T2         ADDRESS
0572  00773 006400       CLB            POSITION IN STRING
0573  00774 054143  MOVEC CPB T1
0574  00775 124770       JMP MOVE,I     RETURN IF END OF STRING
0575  00776 160144       LDA T2,I       NEXT WORD OF 2 CHARACTERS
0576  00777 006011       SLB,RSS
```

```
0577  01000 001727          ALF,ALF         ROTATE IF POSITION IS EVEN
0578  01001 014752          JSB PAK         PAK CHARACTER IN BUFFER
0579  01002 005014          SLB,INB         TEST AND STEP POSITION
0580  01003 034144          ISZ T2          STEP ADDRESS IF POSITION WAS ODD
0581  01004 024774          JMP MOVEC       CONTINUE



0583* SUBROUTINE TO ADD 1 ASCII SPACE TO BUFFER
0584*
0585  01005 000000  SPC1    NOP
0586  01006 060214          LDA A28         ASCII SPACE
0587  01007 014752          JSB PAK           TO BUFFER
0588  01010 125005          JMP SPC1,I      RETURN P+1



0590* SUBROUTINE TO PAK VARIABLE IN B AS OCTAL CHARACTERS
0591* LEADING 0'S ARE NOT PACKED
0592* IF B=0, A SINGLE ASCII 0 IS PACKED
0593* RETURN P+1 AFTER ADDING 1 TRAILING SPACE TO THE BUFFER
0594*
0595  01011 000000  CUTOX   NOP
0596  01012 000040          CLE             E=0 INDICATING NO LEADING 0'S
0597  01013 015021          JSB CUTO        CONVERT & PAK
0598  01014 125011          JMP CUTOX,I



0600* SUBROUTINE TO PAK VARIABLE IN B AS 6 OCTAL CHARACTERS
0601* RETURN P+1 AFTER 1 TRAILING SPACE HAS BEEN ADDED TO THE BUFFER
0602*
0603  01015 000000  CUTO6   NOP
0604  01016 002300          CCE             I=1 INDICATING SIGNIFICANT 0'S
0605  01017 015021          JSB CUTO        CONVERT & PAK
0606  01020 125015          JMP CUTO6,I



0608  01021 000000  CUTO    NOP
0609  01022 060200          LDA M6
0610  01023 070143          STA T1          6 DIGIT COUNTER
0611  01024 005200          RBL             RIGHT JUSTIFY 1ST DIGIT
0612  01025 002404          CLA,INA         1ST DIGIT MASK
0613  01026 040001  CUTO1   AND B           MASK OFF NEXT DIGIT
0614  01027 002002          SZA
0615  01030 002300          CCE             SET E IF NOT 0
0616  01031 030163          IOR B60         ADD ASCII BASE
0617  01032 002040          SEZ
0618  01033 014752          JSB PAK         PAK ONLY IF E=1
0619  01034 005723          BLF,RBR         RIGHT JUSTIFY NEXT DIGIT
0620  01035 060154          LDA B7          NEXT DIGIT MASK
0621  01036 034143          ISZ T1
0622  01037 025026          JMP CUTO1       CONTINUE IF NOT 6 DIGITS
0623  01040 060163          LDA B60
0624  01041 002041          SEZ,RSS         IF VARIABLE WAS 0
```

```
0625  01042 014752          JSB PAK          PUT 1 ASCII 0 IN BUFFER
0626  01043 015005          JSB SPC1       ADD 1 ASCII SPACE TO BUFFER
0627  01044 125021          JMP CUTO,I     RETURN P+1



0629* SUBROUTINE TO PAK INDIRECT LEVEL STRING
0630*
0631  01045 000000  PIL#    NOP
0632  01046 060203          LDA .3
0633  01047 065054          LDB ILMA
0634  01050 014770          JSB MOVE       PAK IL*
0635  01051 064120          LDB IL#
0636  01052 015011          JSB CUTOX      PAK X, SPACE
0637  01053 125045          JMP PIL#,I
0638*
0639  01054 001055  ILMA    DEF ILM
0640  01055 044514  ILM     ASC 2,IL*



0642* SUBROUTINE TO PAK "SB" SPACE
0643*
0644  01057 000000  PSB     NOP
0645  01060 060203          LDA .3
0646  01061 065064          LDB SBMA
0647  01062 014770          JSB MOVE       PAK SB, SPACE
0648  01063 125057          JMP PSB,I      RETURN P+1
0649*
0650  01064 001065  SBMA    DEF SBM
0651  01065 051502  SBM     ASC 2,SB



0653* SUBROUTINE TO INCLUDE B&A ERRORS IN ERROR MESSAGE
0654*
0655  01067 000000  PBA     NOP
0656  01070 060137          LDA REI
0657  01071 010203          AND .3
0658  01072 002003          SZA,RSS        CHECK REGISTER ERROR INDICATOR
0659  01073 125067          JMP PBA,I      NO B OR A REGISTER ERRORS
0660  01074 060211          LDA .38
0661  01075 015161          JSB CEOB       CHECK MESSAGE LENGTH
0662  01076 060204          LDA .4
0663  01077 065113          LDB BAQMA
0664  01100 014770          JSB MOVE       PAK "B,A*"
0665  01101 064126          LDB AB
0666  01102 015015          JSB CUTO6      CNVT & PAK ACTUAL B & SPACE
0667  01103 064125          LDB AA
0668  01104 015015          JSB CUTO6      CNVT & PAK ACTUAL A & SPACE
0669  01105 015057          JSB PSB        PAK "SB "
0670  01106 064122          LDB EB
0671  01107 015015          JSB CUTO6      CNVT & PAK EXPECTED B & SPACE
0672  01110 064121          LDB EA
0673  01111 015015          JSB CUTO6      CNVT & PAK EXPECTED A & SPACE
0674  01112 125067          JMP PBA,I         RETURN P+1
```

```
0675*
0676  01113 001114  BAQMA DEF BAQM
0677  01114 041054  BAQM  ASC 2,B,A*



0679* SUBROUTINE TO INCLUDE E & OV REGISTERS ERRORS IN MESSAGE
0680*
0681  01116 000000  PEO   NOP
0682  01117 060137        LDA REI
0683  01120 010204        AND .4
0684  01121 002003        SZA,RSS
0685  01122 025135        JMP PEO1        E OK
0686  01123 060165        LDA .63
0687  01124 015161        JSB CEOB        CHECK MESSAGE LENGTH
0688  01125 060202        LDA .2
0689  01126 065154        LDB EQMA
0690  01127 014770        JSB MOVE        PAK "E*"
0691  01130 064127        LDB AE
0692  01131 015011        JSB CUTOX       CNVT & PAK ACTUAL E
0693  01132 015057        JSB PSB         PAK "SB"
0694  01133 064123        LDB EE
0695  01134 015011        JSB CUTOX       CNVT & PAK EXPECTED E
0696  01135 060137  PEO1  LDA REI
0697  01136 010155        AND .8
0698  01137 002003        SZA,RSS
0699  01140 125116        JMP PEO,I       OV OK
0700  01141 060165        LDA .63
0701  01142 015161        JSB CEOB        CHECK MESSAGE LENGTH
0702  01143 060203        LDA .3
0703  01144 065155        LDB OQMA
0704  01145 014770        JSB MOVE        PAK "OV*"
0705  01146 064130        LDB AO
0706  01147 015011        JSB CUTOX       CNTV & PAK ACTUAL OV
0707  01150 015057        JSB PSB         PAK "SB"
0708  01151 064124        LDB EO
0709  01152 015011        JSB CUTOX       CNVT & PAK EXPECTED OV
0710  01153 125116        JMP PEO,I       RETURN P+1
0711*
0712  01154 001156  EQMA  DEF EQM
0713  01155 001157  OQMA  DEF OQM
0714  01156 042475  EQM   ASC 1,E*
0715  01157 047526  OQM   ASC 2,OV*



0717* SUBROUTINE TO CHECK BUFFER LENGTH
0718* IF BCNT >= A, THEN
0719*  PRINT THE BUFFER, RESET IT, THEN ADD 2 SPACES
0720* RETURN P+1
0721*
0722  01161 000000  CEOB  NOP
0723  01162 003004        CMA,INA
0724  01163 040332        ADA BCNT
0725  01164 002020        SSA
0726  01165 125161        JMP CEOB,I      OK: BCNT<A
```

```
0727  01166 014745          JSB PBUF        PRINT BUFFER CONTENTS
0728  01167 014737          JSB CBUF        RESET BUFFER
0729  01170 015005          JSB SPC1
0730  01171 015005          JSB SPC1        ADD 2 SPACES
0731  01172 125161          JMP CEOB,I      RETURN P+1
```

```
0002* MULTIPLY RNA BY RNM
0003* STORE PRODUCT IN EB,EA
0004* SET EO TO 0 INDICATING THAT OV IS EXPECTED TO BE CLEARED
0005* RETURN P+1
0006*
0007  01173 000000  .MPY  NOP
0008  01174 060112        LDA RNA
0009  01175 002120        CLE,SSA
0010  01176 003204        CMA,CME,INA    A = POSITIVE MULTIPICAND
0011  01177 064115        LDB RNM
0012  01200 006020        SSB
0013  01201 007204        CMB,CME,INB
0014  01202 074143        STB T1         T1 = POSITIVE MULTIPLIER
0015  01203 007400        CCB
0016  01204 006500        ERB
0017  01205 074144        STB T2         T2 = PRODUCT SIGN: -1 IF NEG,
0018  01206 064201        LDB M16
0019  01207 074145        STB T3         COUNTER = -16
0020  01210 006400        CLB
0021  01211 074124        STB EO         EXPECT OV=0
0022  01212 000000  .MPY1 CLE,SLA        DO THE MULTIPLY
0023  01213 044143        ADB T1           16 ADDS & SHIFTS
0024  01214 006500        ERB
0025  01215 001500        ERA
0026  01216 034145        ISZ T3
0027  01217 025212        JMP .MPY1
0028  01220 034144        ISZ T2         TEST PRODUCT SIGN
0029  01221 025225        JMP .MPY2        IT'S POSITIVE
0030  01222 007000        CMB            DOUBLE LENGTH COMPLIMENT
0031  01223 003007        CMA,INA,SZA,RSS
0032  01224 006004        INB
0033  01225 070121  .MPY2 STA EA         SAVE EXPECTED PRODUCT
0034  01226 074122        STB EB
0035  01227 125173        JMP .MPY,I     RETURN P+1



0037* DIVIDE RNB,RNA BY RMM
0038* RETURN P+1 WITH EXPECTED RESULTS IN EB, EA, & EO
0039*
0040  01230 000000  .DIV  NOP
0041  01231 064113        LDB RNB        B,A = DIVIDEND
0042  01232 060112        LDA RNA
0043  01233 006021        SSB,RSS        MAKE THE DIVIDEND POS. IF NEG,
0044  01234 025240        JMP *+4
0045  01235 007000        CMB
0046  01236 003007        CMA,INA,SZA,RSS
0047  01237 006004        INB
0048  01240 074122        STB EB         SAVE THE POSITIVE DIVIDEND
0049  01241 070121        STA EA           TEMPORARILY
0050  01242 064115        LDB RNM        DIVISOR
0051  01243 006021        SSB,RSS
0052  01244 007004        CMB,INB          MAKE NEGATIVE IF POSITIVE
0053  01245 074144        STB MIND       MIND = NEGATIVE DIVISOR
0054  01246 006003        SZB,RSS
0055  01247 025303        JMP OVFLO      OVERFLOW - DIVIDE BY 0
```

```
0056  01250 044122          ADB EB
0057  01251 006021          SSB,RSS
0058  01252 025303          JMP OVFLO      OVERFLOW IF ABS(B-DIVISOR) IS +
0059  01253 064201          LDB M16
0060  01254 074143          STB COUNT      COUNTER + -16
0061  01255 064122          LDB EB         RESTORE
0062  01256 001600  LOOP    ELA            DO THE DIVIDE
0063  01257 005640          ELB,CLE
0064  01260 074122          STB EB
0065  01261 044144          ADB MIND
0066  01262 002041          SEZ,RSS
0067  01263 064122          LDB EB
0068  01264 034143          ISZ COUNT
0069  01265 025256          JMP LOOP
0070  01266 001600          ELA
0071  01267 070121          STA EA         SAVE POS. QUOTIENT TEMPORARILY
0072  01270 060113          LDA RNB
0073  01271 002020          SSA            MAKE REMAINDER NEGATIVE IF SIGN
0074  01272 007004          CMB,INB          OF DIVIDEND WAS NEGATIVE
0075  01273 074122          STB EB         SAVE THE EXPECTED REMAINDER
0076  01274 064121          LDB EA         B + POSITIVE QUOTIENT
0077  01275 020115          XOR RNM        A(15) = SIGN OF QUOTIENT
0078  01276 002021          SSA,RSS        TEST SIGN
0079  01277 025307          JMP .DIV1        +, GO CHECK SIGN OF QUOTIENT
0080  01300 007104          CMB,CLE,INB      -, MAKE QUOTIENT NEG.
0081  01301 074121          STB EA         SAVE NEGATIVE EXPECTED QUOTIENT
0082  01302 006061          SEZ,SSB,RSS    CHECK SIGN & CARRY NOT BOTH 0
0083  01303 002405  OVFLO   CLA,INA,RSS
0084  01304 002400  NOV     CLA
0085  01305 070124          STA EO         SAVE EXPECTED OVERFLOW
0086  01306 125230          JMP .DIV,I       RETURN P+1
0087  01307 006020  .DIV1   SSB
0088  01310 025303          JMP OVFLO      OVERFLOW IF SIGN NOT 0
0089  01311 025304          JMP NOV
0090*
0091  00143         COUNT   EQU T1         LOOP COUNT
0092  00144         MIND    EQU T2         NEGATIVE DIVISOR



0094  01312 000000  .ASR    NOP
0095  01313 002400          CLA
0096  01314 070124          STA EO         EXPECT OV REGISTER TO BE 0
0097  01315 015422          JSB IISR       INITIALIZE SOFTWARE ROUTINE
0098  01316 004071  .ASR1   CLE,SLB,BRS
0099  01317 002300          CCE            DO ASR EQUIVALENT
0100  01320 001500          ERA              USING STANDARD INSTRUCTIONS
0101  01321 034143          ISZ T1
0102  01322 025316          JMP .ASR1
0103  01323 070121          STA EA         SAVE THE EXPECTED RESULTS
0104  01324 074122          STB EB
0105  01325 125312          JMP .ASR,I     RETURN P+1



0107  01326 000000  .ASL    NOP
```

```
0108  01327 015422          JSB IISR        INITIALIZE SOFTWARE ROUTINE
0109  01330 000066  .ASL1 CLE,ELA          DO ASL EQUIVALENT
0110  01331 005600          ELB               USING STANDARD INSTRUCTIONS
0111  01332 005225          RBL,ERB         MAKE SURE SIGN DOESN'T CHANGE
0112  01333 034143          ISZ T1
0113  01334 025330          JMP .ASL1
0114  01335 070121          STA EA          SAVE EXPECTED RESULTS
0115  01336 074122          STB EB
0116  01337 015422          JSB IISR        INITIALIZE SOFTWARE ROUTINE
0117  01340 000066  .ASL2 CLE,ELA           SIGNIFICANT BITS WERE
0118  01341 005600          ELB               SHIFTED OUT
0119  01342 006020          SSB                 IF SO, MAKE EO=1
0120  01343 002200          CME
0121  01344 002040          SEZ
0122  01345 025351          JMP ASLOV       OVERFLOW: B(15) NOT SAME AS E
0123  01346 034143          ISZ T1
0124  01347 025340          JMP .ASL2
0125  01350 002401          CLA,RSS         NO OVERFLOW: EO = 0
0126  01351 002404  ASLOV CLA,INA           OVERFLOW: EO = 1
0127  01352 070124          STA EO
0128  01353 125326          JMP .ASL,I      RETURN P+1


0130  01354 000000  .LSR  NOP
0131  01355 015422          JSB IISR        INITIALIZE SOFTWARE ROUTINE
0132  01356 004065  .LSR1 CLE,ERB          DO LSR EQUIVALENT
0133  01357 001500          ERA               USING STANDARD INSTRUCTIONS
0134  01360 034143          ISZ T1
0135  01361 025356          JMP .LSR1
0136  01362 070121          STA EA          SAVE THE EXPECTED RESULTS
0137  01363 074122          STB EB
0138  01364 125354          JMP .LSR,I      RETURN P+1


0140  01365 000000  .LSL  NOP
0141  01366 015422          JSB IISR        INITIALIZE SOFTWARE ROUTINE
0142  01367 000066  .LSL1 CLE,ELA          DO LSL EQUIVALENT
0143  01370 005600          ELB               USING STANDARD INSTRUCTIONS
0144  01371 034143          ISZ T1
0145  01372 025367          JMP .LSL1
0146  01373 070121          STA EA          SAVE THE EXPECTED RESULTS
0147  01374 074122          STB EB
0148  01375 125365          JMP .LSL,I      RETURN P+1


0150  01376 000000  .RRR  NOP
0151  01377 015422          JSB IISR        INITIALIZE SOFTWARE ROUTINE
0152  01400 004065  .RRR1 CLE,ERB
0153  01401 001500          ERA             DO RRR EQUIVALENT
0154  01402 005225          RBL,ERB           USING STANDARD INSTRUCTIONS
0155  01403 034143          ISZ T1
0156  01404 025400          JMP .RRR1
0157  01405 070121          STA EA          SAVE THE EXPECTED RESULTS
```

```
0158  01406 074122          STB EB
0159  01407 125376          JMP .RRR,I     RETURN P+1



0161  01410 000000  .RRL    NOP
0162  01411 015422          JSB IISR       INITIALIZE SOFTWARE ROUTINE
0163  01412 000066  .RRL1   CLE,ELA
0164  01413 005600          ELB            DO RRL EQUIVALENT
0165  01414 001326          RAR,ELA          USING STANDARD INSTRUCTIONS
0166  01415 034143          ISZ T1
0167  01416 025412          JMP .RRL1
0168  01417 070121          STA EA         SAVE THE EXPECTED RESULTS
0169  01420 074122          STB EB
0170  01421 125410          JMP .RRL,I     RETURN P+1



0172  01422 000000  IISR   NOP
0173  01423 060117          LDA SHFT
0174  01424 003007          CMA,INA,SZA,RSS     MAKE SHFT NEGATIVE
0175  01425 040201          ADA M16             MAKE -16 IF 0
0176  01426 070143          STA T1              SAVE COUNTER
0177  01427 064113          LDB RNB             GET ARGUMENTS IN B & A
0178  01430 060112          LDA RNA
0179  01431 125422          JMP IISR,I     RETURN P+1
```

```
0181* SUBROUTINE TO INITIALIZE SHIFT COUNT FOR SHIFT ROTATE TESTS
0182*  IF SWITCH 6 IS UP:  SHFT ← SHFT(3-0),  S6 ← NON 0
0183*  IF SWITCH 6 IS DOWN:  GENERATE NEW ARGUMENTS, SHFT ← S6 ← 0
0184* RETURN P+1
0185*
0186  01432 000000  ISRT  NOP
0187  01433 014502        JSB GSR
0188  01434 010171        AND BIT6
0189  01435 070140        STA S6          REMEMBER STATE OF SWITCH 6
0190  01436 002002        SZA
0191  01437 025443        JMP ISRT1       SWITCH 6 UP
0192  01440 070117        STA SHFT        SWITCH 6 DOWN: SHFT ← 0
0193  01441 014377        JSB RNG         GENERATE NEW ARGUMENTS
0194  01442 125432        JMP ISRT,I
0195  01443 060117  ISRT1 LDA SHFT
0196  01444 010160        AND B17
0197  01445 070117        STA SHFT        SHFT ← SHFT(3-0)
0198  01446 125432        JMP ISRT,I



0200* SHIFT/ROTATE MESSAGE PROCESSING
0201*
0202  01447 000000  SRMP  NOP
0203  01450 014507        JSB CEM         CHECK FOR ERROR MESSAGE
0204  01451 125447        JMP SRMP,I        NO MESG, WANTED - RETURN P+1
0205  01452 060155        LDA .8
0206  01453 064233        LDB BAWMA
0207  01454 014770        JSB MOVE        PAK "B,A WAS "
0208  01455 064113        LDB RNB         CONVERT AND PAK WHAT
0209  01456 015015        JSB CUTO6         B & A WAS
0210  01457 064112        LDB RNA
0211  01460 015015        JSB CUTO6
0212  01461 015067        JSB PBA         LIST ANY B & A REGISTER ERRORS
0213  01462 015116        JSB PEO         LIST ANY E OR OV REG ERRORS
0214  01463 014745        JSB PBUF        PRINT THE ERROR MESSAGE
0215  01464 125447        JMP SRMP,I        RETURN P+1



0217* SUBROUTINE TO CHECK SHIFT COUNT
0218*  IF S6 IS NOT 0, RETURN P+2
0219*  IF S6 IS 0, STEP THEN TEST SHFT
0220*   RETURN P+2 IF SHFT=16, ELSE RETURN P+1
0221*
0222  01465 000000  CSHFT NOP
0223  01466 060140        LDA S6
0224  01467 002002        SZA
0225  01470 025474        JMP CSHF2       S6 NOT 0, RETURN P+2
0226  01471 034117        ISZ SHFT        SHFT ← SHFT + 1
0227  01472 060117        LDA SHFT
0228  01473 060161        CPA .16
0229  01474 035465  CSHF2 ISZ CSHFT       SHFT = 16, RETURN P+2
0230  01475 125465        JMP CSHFT,I       ELSE RETURN P+1
```

```
0232  01476 107000  CONFG OCT 107000     START OF CONFIGURATION HALT
0233  01477 102501        LIA 1          GET AND
0234  01500 070135        STA ISR          SAVE INTERNAL SWITCH REGISTER
0235  01501 107077        OCT 107077     END OF CONFIGURATION HALT



0237  01502 107700  START CLC 0,C        TURN EVERYTHING OFF
0238  01503 060221        LDA HLT1       SET TRAP CELL HALTS
0239  01504 006404        CLB,INB          PUT 1060XX INTO ADDRESS XX
0240  01505 002004  ST1   INA                 XX = 02,03,04,...77
0241  01506 006004        INB
0242  01507 170001        STA 0,I
0243  01510 050222        CPA HLT77
0244  01511 002001        RSS
0245  01512 025505        JMP ST1
0246  01513 060214        LDA A28        RESET END OF PASS COUNT
0247  01514 070260        STA MG77A        STORE 8 ASCII SPACES IN THE 4
0248  01515 070261        STA MG77A+1      WORD STARTING AT LOC. MG77A
0249  01516 070262        STA MG77A+2
0250  01517 070263        STA MG77B
0251  01520 014561        JSB CNEM       CHECK FOR NON-ERROR MESSAGE
0252  01521 025525        JMP $OP         NO MESSAGE
0253  01522 000161        LDA .16
0254  01523 064234        LDB MIA
0255  01524 114102        JSB LIST,I      PRINT INTRODUCTORY MESSAGE
```

```
0257   01525 002400   SOP    CLA
0258   01526 070136          STA LCNT       NEW PASS - RESET LOOP COUNT
0259   01527 002400   AGAIN  CLA
0260   01530 070134          STA FLG6       NEW LOOP - CLEAR SW, 6 FLAG
0261*
0262   01531 002404          CLA,INA
0263   01532 070142          STA TST#       TEST 1
0264   01533 014377   DLDL   JSB RNG        CHECK AND PERHAPS GET NEW ARG,
0265   01534 060112          LDA RNA        GET AND SAVE EXPECTED A & B
0266   01535 070121          STA EA
0267   01536 060113          LDA RNB
0268   01537 070122          STA EB
0269   01540 060237          LDA RNAA       ADDRESS OF RNA
0270   01541 070147          STA DAD        SAVE THE DIRECT ADDRESS
0271   01542 002400          CLA            START WITH 0
0272   01543 070120          STA IL#          INDIRECT ADDRESSING LEVELS
0273   01544 040146   DLDC   ADA APT        FORM THE IN-LINE ADDRESS
0274   01545 160000          LDA A,I
0275   01546 071554          STA DLDA         PUT IT IN-LINE
0276   01547 014677   DLDR   JSB IREGO      INITIALIZE THE REGISTERS
0277   01550 002400          CLA
0278   01551 006400          CLB
0279*
0280   01552 000000          NOP            ** FOR PATCHING **
0281   01553 104200          DEF DLD,I      DO THE DLD MACRO CALL
0282   01554 000000   DLDA   NOP              (ADDRESS)
0283   01555 000000          NOP            ** FOR PATCHING **
0284*
0285   01556 014705          JSB CHECK      SAVE THE REGISTERS & CHECK THEM
0286   01557 002003          SZA,RSS        A=0 INDICATES NO ERRORS
0287   01560 025573          JMP DLD1         NO ERRORS
0288*
0289   01561 014507          JSB CEM        CHECK FOR ERROR MESSAGE
0290   01562 025567          JMP DLD2         NO MESSAGE WANTED
0291   01563 015045          JSB PIL#       PAK # INDIRECT LEVELS
0292   01564 015067          JSB PBA        LIST ANY B OR A REG ERRORS
0293   01565 015116          JSB PEO        LIST ANY E OR OV REG ERRORS
0294   01566 014745          JSB PBUF       PRINT THE ERROR MESSAGE
0295*
0296   01567 014616   DLD2   JSB CEHLT      CHECK FOR ERROR HALT
0297   01570 102001          HLT 1            ERROR HALT - REGISTERS RESTORE
0298   01571 014602          JSB CS9
0299   01572 025577          JMP DLDE       SWITCH 9 UP - ERROR BREAKOUT
0300*
0301   01573 014610   DLD1   JSB CS13
0302   01574 025547          JMP DLDR       SWITCH 13 UP: REPEAT - SAME DATA
0303   01575 014557          JSB CIA        CHECK INDIRECT ADDRESSING
0304   01576 025544          JMP DLDC         DO FOR NEXT LEVEL
0305   01577 014636   DLDE   JSB EOT
0306   01600 102076          HLT 76B        END OF TEST HALT: A+B+1
0307   01601 014571          JSB CS8
0308   01602 025533          JMP DLDL       SWITCH 8 UP - LOOP ON THIS TEST
```

```
0310  01603 060202  TST2  LDA .2
0311  01604 070140        STA TST#      TEST 2
0312  01605 014377  DSTL  JSB RNG       CHECK & PERHAPS GET NEW ARG.
0313  01606 060133        LDA DSA       ADDRESS OF DSA
0314  01607 070147        STA DAD       SAVE THE DIRECT ADDRESS
0315  01610 002400        CLA           START WITH 0
0316  01611 070120        STA IL#         INDIRECT ADDRESSING LEVELS
0317  01612 040146  DSTC  ADA APT       FORM THE IN-LINE ADDRESS
0318  01613 100000        LDA A,I
0319  01614 071625        STA DSTA        PUT IT IN-LINE
0320  01615 002400        CLA
0321  01616 070131        STA SA        CLEAR STORE LOCATIONS TO
0322  01617 070132        STA SB          REMOVE ANY RESIDUAL DATA
0323  01620 014677  DSTR  JSB IREGO     INITIALIZE THE REGISTERS
0324  01621 070121        STA EA
0325  01622 074122        STB EB        SAVE ARG'S FOR CHECKING
0326*
0327  01623 000000        NOP           ** FOR PATCHING **
0328  01624 104400        DEF DST,I     DST SA MACRO CALL
0329  01625 000000  DSTA  NOP             (ADDRESS)
0330  01626 000000        NOP           ** FOR PATCHING **
0331*
0332  01627 060131        LDA SA        LOAD WHAT WAS ACTUALLY
0333  01630 064132        LDB SB          STORED FOR CHECKING
0334  01631 014705        JSB CHECK     GO CHECK RESULTS
0335  01632 002003        SZA,RSS       A=0 INDICATES NO ERRORS
0336  01633 025665        JMP DST1        NO ERRORS
0337*
0338  01634 014587        JSB CEM       CHECK FOR ERROR MESSAGE
0339  01635 025661        JMP DST2        NO MESSAGE WANTED
0340  01636 015045        JSB PIL#      PAK # INDIRECT LEVELS
0341  01637 060137        LDA REI
0342  01640 010203        AND .3
0343  01641 002003        SZA,RSS
0344  01642 025667        JMP DST3      ERROR NOT STORED CONTENTS
0345  01643 060207        LDA .11
0346  01644 064241        LDB STRMA
0347  01645 014770        JSB MOVE      PAK "STORED WAS "
0348  01646 064126        LDB AB
0349  01647 015015        JSB CUT06     CONVERT AND PACK CONTENTS FOUND
0350  01650 064125        LDB AA          IN THE STORE LOCATIONS
0351  01651 015015        JSB CUT06
0352  01652 015057        JSB PSB       PAK "SB "
0353  01653 064113        LDB RNB
0354  01654 015015        JSB CUT06     CONVERT AND PACK WHAT SHOULD
0355  01655 064112        LDB RNA       HAVE BEEN STORED (CONTENTS OF
0356  01656 015015        JSB CUT06       A & B WHEN DST CALL WAS MADE)
0357  01657 015116  DST3  JSB PEO       LIST E AND OV REG ERRORS IF ANY
0358  01660 014745        JSB PBUF      PRINT THE ERROR MESSAGE
0359*
0360  01661 014616  DST2  JSB CEHLT     CHECK FOR ERROR HALT
0361  01662 102002        HLT 2          ERROR HLT: A,B =(STORE LOC.)
0362  01663 014602        JSB C89
0363  01664 025671        JMP DSTE      SWITCH 9 UP = ERROR BREAKOUT
0364*
0365  01665 014610  DST1  JSB C813
```

```
0366  01666 025620        JMP DSTR      SWITCH 13 UP: REPEAT - SAME DATA
0367  01667 014567        JSB CIA       CHECK INDIRECT ADDRESSING
0368  01670 025612        JMP DSTC        DO FOR NEXT LEVEL
0369  01671 014636  DSTE  JSB EOT
0370  01672 102076        HLT 76B       END OF TEST HALT: A+B=2
0371  01673 014671        JSB C8B
0372  01674 025565        JMP DSTL      SWITCH 8 UP - LOOP ON THIS TEST
```

```
0374  01675 060203  TST3  LDA .3
0375  01676 070142        STA TST#      TEST 3
0376  01677 014377  MPYL  JSB RNG       CHECK AND PERHAPS GET NEW ARG,
0377  01700 015173        JSB .MPY      SOFTWARE ROUTINE - EXPECTED A,B,
0378  01701 060240        LDA RNMA      ADDRESS OF RNM
0379  01702 070147        STA DAD       SAVE THE DIRECT ADDRESS
0380  01703 002400        CLA           START WITH 0
0381  01704 070120        STA IL#         INDIRECT ADDRESSING LEVELS
0382  01705 040146  MPYC  ADA APT       FORM THE IN-LINE ADDRESS
0383  01706 160000        LDA A,I
0384  01707 071713        STA MPYA        PUT IT IN-LINE
0385  01710 014663  MPYR  JSB IREG      INITIALIZE THE REG'S
0386*
0387  01711 000000        NOP           ** FOR PATCHING **
0388  01712 100200        DEF MPY,I     DO THE MPY MACRO CALL
0389  01713 000000  MPYA  NOP             (ADDRESS)
0390  01714 000000        NOP           ** FOR PATCHING **
0391*
0392  01715 014705        JSB CHECK     SAVE THE REGISTERS & CHECK THEM
0393  01716 002003        SZA,RSS       A=0 INDICATES NO ERRORS
0394  01717 025741        JMP MPY1        NO ERROR
0395*
0396  01720 014507        JSB CEM       CHECK FOR ERROR MESSAGE
0397  01721 025735        JMP MPY2        NO MESSAGE WANTED
0398  01722 015045        JSB PIL#      PAK # INDIRECT LEVELS
0399  01723 060155        LDA .8
0400  01724 064231        LDB AMWMA
0401  01725 014770        JSB MOVE      PAK "A,M WAS "
0402  01726 064112        LDB RNA
0403  01727 015015        JSB CUT06     CONVERT & PAK RNA & SPACE
0404  01730 064115        LDB RNM
0405  01731 015015        JSB CUT06     CONVERT & PAK RNM & SPACE
0406  01732 015067        JSB PBA       LIST ANY B & A REG ERRORS
0407  01733 015116        JSB PEO       LIST ANY E AND OV REG ERRORS
0408  01734 014745        JSB PBUF      PRINT THE ERROR MESSAGE
0409*
0410  01735 014616  MPY2  JSB CEHLT     CHECK FOR ERROR HALT
0411  01736 102003        HLT 3           ERROR HALT: REGISTERS RESTORED
0412  01737 014602        JSB C89
0413  01740 025745        JMP MPYE      SWITCH 9 UP - ERROR BREAKOUT
0414*
0415  01741 014610  MPY1  JSB C813
0416  01742 025710        JMP MPYR      SWITCH 13 UP: REPEAT - SAME DATA
0417  01743 014657        JSB CIA       CHECK INDIRECT ADDRESSING
0418  01744 025705        JMP MPYC        DO FOR NEXT LEVEL
0419  01745 014636  MPYE  JSB EOT
0420  01746 102076        HLT 76B       END OF TEST HALT: A=B=3
0421  01747 014571        JSB C88
0422  01750 025677        JMP MPYL      SWITCH 8 UP - LOOP ON THIS TEST
0423*
0424  01751 025777        JMP PG1
0425  01777               ORG 1777B
0426  01777         PG1   EQU *
```

```
0426  01777 000204  TST4  LDA .4
0429  02000 070142        STA TST#      TEST 4
0430  02001 014377  DIVL  JSB RNG       CHECK AND PERHAPS GET NEW ARG.
0431  02002 015239        JSB .DIV      SOFTWARE ROUTINE - GET EXPECTED
0432  02003 000240        LDA RNMA      ADDRESS OF RNM
0433  02004 070147        STA DAD       SAVE THE DIRECT ADDRESS
0434  02005 002400        CLA           START WITH 0
0435  02006 070120        STA IL#         INDIRECT ADDRESSING LEVELS
0436  02007 040146  DIVC  ADA APT       FORM THE IN LINE ADDRESS
0437  02010 100000        LDA A,I
0438  02011 072015        STA DIVA        PUT IT IN LINE
0439  02012 014663  DIVR  JSB IREG      INITIALIZE THE REG'S
0440*
0441  02013 000000        NOP           ** FOR PATCHING **
0442  02014 100400        DEF DIV,I     DO THE DIV MACRO CALL
0443  02015 000000  DIVA  NOP             (ADDRESS)
0444  02016 000000        NOP           ** FOR PATCHING **
0445*
0446  02017 014705        JSB CHECK     SAVE THE REGISTERS & CHECK
0447  02020 002003        SZA,RSS       A=0 INDICATES NO ERRORS
0448  02021 026045        JMP DIV1        NO ERROR
0449*
0450  02022 014507        JSB CEM       CHECK FOR ERROR MESSAGE
0451  02023 026041        JMP DIV2        NO MESSAGE WANTED
0452  02024 015045        JSB PIL#      PAK # INDIRECT LEVELS
0453  02025 000157        LDA .16
0454  02026 064232        LDB BAMA
0455  02027 014770        JSB MOVE      PAK "B,A,M WAS "
0456  02030 064113        LDB RNB
0457  02031 015015        JSB CUTO6     CNVT & PAK RNB & SPACE
0458  02032 064112        LDB RNA
0459  02033 015015        JSB CUTO6     CNVT & PAK RNA & SPACE
0460  02034 064115        LDB RNM
0461  02035 015015        JSB CUTO6     CNVT & PAK RNM & SPACE
0462  02036 015067        JSB PBA       LIST ANY B & A ERRORS
0463  02037 015116        JSB PEO       LIST ANY E OR OV REG ERRORS
0464  02040 014745        JSB PBUF      PRINT THE ERROR MESSAGE
0465*
0466  02041 014616  DIV2  JSB CEHLT     CHECK FOR ERROR HALT
0467  02042 102004        HLT 4           ERROR HALT: REGISTERS RESTORED
0468  02043 014602        JSB C89
0469  02044 026051        JMP DIVE      SWITCH 9 UP = ERROR BREAKOUT
0470*
0471  02045 014610  DIV1  JSB C813
0472  02046 026012        JMP DIVR      SWITCH 13 UP: REPEAT = SAME DATA
0473  02047 014557        JSB CIA       CHECK INDIRECT ADDRESSING
0474  02050 026007        JMP DIVC        DO FOR NEXT LEVEL
0475  02051 014636  DIVE  JSB EOT
0476  02052 102076        HLT 76B       END OF TEST HALT: A=B=4
0477  02053 014571        JSB C88
0478  02054 026001        JMP DIVL      SWITCH 8 UP = LOOP ON THIS TEST
```

```
0480   02055 000205   TST5   LDA .5
0481   02056 070142         STA TST#      TEST 5
0482   02057 015432   ASRL  JSB ISRT      INITIALIZE SHIFT/ROTATE TEST
0483   02060 015312   ASRC  JSB .ASR      SOFTWARE ROUTINE - GET EXPECTED
0484   02061 000117         LDA SHFT      FORM THE INSTRUCTION
0485   02062 030223         IOR ASRB
0486   02063 072066         STA ASRI        PUT IT IN LINE
0487   02064 014663   ASRR  JSB IREG      INITIALIZE THE REGISTERS
0488*
0489   02065 000000         NOP           ** FOR PATCHING **
0490   02066 101020   ASRI  DEF ASR,I     ASR (XX) INSTRUCTION
0491   02067 000000         NOP           ** FOR PATCHING **
0492*
0493   02070 014705         JSB CHECK     SAVE THE REGISTERS & CHECK THEM
0494   02071 002003         SZA,RSS       A=0 INDICATES NO ERRORS
0495   02072 026100         JMP ASR1      NO ERRORS
0496   02073 015447         JSB SRMP      SHIFT/ROTATE MESSAGE PROCESSING
0497   02074 014616         JSB CEHLT     CHECK FOR ERROR HALT
0498   02075 102005         HLT 5           ERROR HALT: REGISTERS RESTORED
0499   02076 014602         JSB C89
0500   02077 026104         JMP ASRE      SWITCH 9 UP - ERROR BREAKOUT
0501*
0502   02100 014610   ASR1  JSB C813
0503   02101 026064         JMP ASRR      SWITCH 13 UP: REPEAT - SAME DATA
0504   02102 015465         JSB CSHFT     CHECK THE SHIFT COUNT
0505   02103 026060         JMP ASRC        DO FOR THE NEXT VALUE
0506   02104 014636   ASRE  JSB EOT
0507   02105 102076         HLT 76B       END OF TEST HALT: A=B=5
0508   02106 014571         JSB C86
0509   02107 026057         JMP ASRL      SWITCH 6 UP - LOOP ON THIS TEST
```

```
0511   02110 060206   TST6   LDA  .6
0512   02111 070142          STA  TST#      TEST 6
0513   02112 015432   ASLL  JSB  ISRT      INITIALIZE SHIFT ROTATE TEST
0514   02113 015326   ASLC  JSB  .ASL      SOFTWARE ROUTINE - GET EXPECTED
0515   02114 060117          LDA  SHFT      FORM THE INSTRUCTION
0516   02115 030224          IOR  ASL0
0517   02116 072121          STA  ASLI        STORE IN LINE
0518   02117 014663   ASLR  JSB  IREG      INITIALIZE THE REG'S
0519*
0520   02120 000000          NOP            ** FOR PATCHING **
0521   02121 100020   ASLI  DEF  ASL,I     ASL (XX) INSTRUCTION
0522   02122 000000          NOP            ** FOR PATCHING **
0523*
0524   02123 014705          JSB  CHECK     SAVE THE REGISTERS & CHECK THEM
0525   02124 002003          SZA,RSS        A=0 INDICATES NO ERRORS
0526   02125 026133          JMP  ASL1        NO ERRORS
0527*
0528   02126 015447          JSB  SRMP      SHIFT/ROTATE MESSAGE PROCESSING
0529   02127 014616          JSB  CEHLT     CHECK FOR ERROR HALT
0530   02130 102006          HLT  6           ERROR HALT: REGISTERS RESTORED
0531   02131 014602          JSB  C89
0532   02132 026137          JMP  ASLE      SWITCH 9 UP - ERROR BREAKOUT
0533*
0534   02133 014610   ASL1  JSB  C813
0535   02134 026117          JMP  ASLR      SWITCH 13 UP: REPEAT - SAME DATA
0536   02135 015465          JSB  CSHFT     CHECK THE SHIFT COUNT
0537   02136 026113          JMP  ASLC        DO FOR NEXT VALUE
0538   02137 014636   ASLE  JSB  EOT
0539   02140 102076          HLT  76B       END OF TEST HALT: A=B=6
0540   02141 014571          JSB  C88
0541   02142 026112          JMP  ASLL      SWITCH 8 UP - LOOP ON THIS TEST
```

```
0543   02143 000154  TST7   LDA .7
0544   02144 070142         STA TST#      TEST 7
0545   02145 015432  LSRL   JSB ISRT      INITIALIZE SHIFT/ROTATE TEST
0546   02146 015364  LSRC   JSB .LSR      SOFTWARE ROUTINE - GET EXPECTED
0547   02147 000117         LDA SHFT      FORM  THE INSTRUCTION
0548   02150 030225         IOR LSR0
0549   02151 072154         STA LSRI        PUT IT IN LINE
0550   02152 014677  LSRR   JSB IREGO     INITIALIZE THE REGISTERS
0551*
0552   02153 000000         NOP           ** FOR PATCHING **
0553   02154 101040  LSRI   DEF LSR,I     LSR (XX) INSTRUCTION
0554   02155 000000         NOP           ** FOR PATCHING **
0555*
0556   02156 014705         JSB CHECK     SAVE THE REGISTERS & CHECK THEM
0557   02157 002003         SZA,RSS       A=0 INDICATES NO ERROR
0558   02160 026166         JMP LSR1        NO ERRORS
0559*
0560   02161 015447         JSB SRMP      SHIFT/ROTATE MESSAGE PROCESSING
0561   02162 014616         JSB CEHLT     CHECK FOR ERROR HALT
0562   02163 102007         HLT 7           ERROR HALT - REG'S RESTORED
0563   02164 014662         JSB C89
0564   02165 026172         JMP LSRE      SWITCH 9 UP - ERROR BREAKOUT
0565*
0566   02166 014610  LSR1   JSB C813
0567   02167 026152         JMP LSRR      SWITCH 13 UP: REPEAT - SAME DATA
0568   02170 015465         JSB CSHFT     CHECK THE SHIFT COUNT
0569   02171 026146         JMP LSRC        DO FOR NEXT VALUE
0570   02172 014636  LSRE   JSB EOT
0571   02173 102076         HLT 76B       END OF TEST HALT: A=B=7
0572   02174 014671         JSB C88
0573   02175 026145         JMP LSRL      SWITCH 8 UP - LOOP ON THIS TEST
```

```
0575   02176 000155  TST10 LDA B10
0576   02177 070142         STA TST#       TEST 10
0577   02200 015432  LSLL   JSB IRT        INITIALIZE SHIFT/ROTATE TEST
0578   02201 015366  LSLC   JSB .LSL       SOFTWARE ROUTINE - GET EXPECTED
0579   02202 000117         LDA SHFT       FORM THE INSTRUCTION
0580   02203 030226         IOR LSL0
0581   02204 072207         STA LSLI         PUT IT IN LINE
0582   02205 014077  LSLR   JSB IREGS      INITIALIZE THE REGISTERS
0583*
0584   02206 000000         NOP            ** FOR PATCHING **
0585   02207 100040  LSLI   DEF LSL,I      LSL (XX) INSTRUCTION
0586   02210 000000         NOP            ** FOR PATCHING **
0587*
0588   02211 014105         JSB CHECK      SAVE THE REGISTERS & CHECK THEM
0589   02212 002003         SZA,RSS        A=0 INDICATES NO ERRORS
0590   02213 026221         JMP LSL1         NO ERRORS
0591*
0592   02214 015447         JSB SRMP       SHIFT/ROTATE MESSAGE PROCESSING
0593   02215 014016         JSB CEHLT      CHECK FOR ERROR HALT
0594   02216 102010         HLT 10B          ERROR HALT - REG'S RESTORED
0595   02217 014002         JSB C$9
0596   02220 026225         JMP LSLE       SWITCH 9 UP - ERROR BREAKOUT
0597*
0598   02221 014010  LSL1   JSB C$13
0599   02222 026205         JMP LSLR       SWITCH 13 UP: REPEAT - SAME DATA
0600   02223 015465         JSB CSHFT      CHECK THE SHIFT COUNT
0601   02224 026201         JMP LSLC         DO FOR NEXT VALUE
0602   02225 014036  LSLE   JSB EOT
0603   02226 102076         HLT 76B        END OF TEST HALT: A=B=10
0604   02227 014071         JSB C$8
0605   02230 026200         JMP LSLL       SWITCH 8 UP - LOOP ON THIS TEST
```

```
0607   02231 000156  TST11 LDA B11
0608   02232 070142        STA TST#      TEST 11
0609   02233 015432  RRRL  JSB .RRR      INITIALIZE SHIFT/ROTATE TEST
0610   02234 015376  RRRC  JSB .RRR      SOFTWARE ROUTINE - SET EXPECTED
0611   02235 000117        LDA SHFT      FORM THE INSTRUCTION
0612   02236 030227        IOR RRR0
0613   02237 072242        STA RRRI        PUT IT IN LINE
0614   02240 014077  RRRR  JSB IRG80      INITIALIZE THE REGISTERS
0615*
0616   02241 000000        NOP           ** FOR PATCHING **
0617   02242 101100  RRRI  DEF RRR,I     RRR (XX) INSTRUCTION
0618   02243 000000        NOP           ** FOR PATCHING **
0619*
0620   02244 014706        JSB CHECK     SAVE THE REGISTERS & CHECK THEM
0621   02245 002003        SZA,RSS       A=0 INDICATES NO ERROR
0622   02246 020254        JMP RRR1        NO ERRORS
0623*
0624   02247 015447        JSB SRMP      SHIFT/ROTATE MESSAGE PROCESSING
0625   02250 014616        JSB CEHLT     CHECK FOR ERROR HALT
0626   02251 102011        HLT 11B         ERROR HALT - REG'S RESTORED
0627   02252 014002        JSB C$0
0628   02253 020260        JMP RRRE      SWITCH 0 UP - ERROR BREAKOUT
0629*
0630   02254 014613  RRR1  JSB C$13
0631   02255 020240        JMP RRRR      SWITCH 13 UP: REPEAT - SAME DATA
0632   02256 015466        JSB CSHFT     CHECK THE SHIFT COUNT
0633   02257 020234        JMP RRRC        DO FOR NEXT VALUE
0634   02260 014636  RRRE  JSB EOT
0635   02261 102076        HLT 76B       END OF TEST HALT: A=B=11
0636   02262 014571        JSB C$8
0637   02263 020233        JMP RRRL      SWITCH 8 UP - LOOP ON THIS TEST
```

```
0639   02264 000157   TST12 LDA B12
0640   02265 070142         STA TST#       TEST 12
0641   02266 015432   RRLL  JSB ISRT       INITIALIZE SHIFT/ROTATE TEST
0642   02267 015410   RRLC  JSB .RRL       SOFTWARE ROUTINE - GET EXPECTED
0643   02270 000117         LDA SHFT       FORM THE INSTRUCTION
0644   02271 030230         IOR RRL0
0645   02272 072275         STA RRLI         PUT IT IN LINE
0646   02273 014677   RRLR  JSB IREGO      INITIALIZE THE REGISTERS
0647*
0648   02274 000000         NOP            ** FOR PATCHING **
0649   02275 100100   RRLI  DEF RRL,I      RRL (XX) INSTRUCTION
0650   02276 000000         NOP            ** FOR PATCHING **
0651*
0652   02277 014705         JSB CHECK      SAVE THE REGISTERS & CHECK THEM
0653   02300 002003         SZA,RSS        A=B INDICATES AN ERROR
0654   02301 026307         JMP RRL1         NO ERRORS
0655*
0656   02302 015447         JSB SRMP       SHIFT/ROTATE MESSAGE PROCESSING
0657   02303 014616         JSB CEHLT      CHECK FOR ERROR HALT
0658   02304 102012         HLT 12B          ERROR HALT - REG'S RESTORED
0659   02305 014602         JSB C89
0660   02306 026313         JMP RRLE       SWITCH 9 UP - ERROR BREAKOUT
0661*
0662   02307 014610   RRL1  JSB C813
0663   02310 026273         JMP RRLR       SWITCH 13 UP: REPEAT - SAME DATA
0664   02311 015465         JSB CSHFT      CHECK THE SHIFT COUNT
0665   02312 026267         JMP RRLC         DO FOR NEXT VALUE
0666   02313 014636   RRLE  JSB EOT
0667   02314 102076         HLT 76B        END OF TEST HALT: A=B=12
0668   02315 014671         JSB C88
0669   02316 026266         JMP RRLL       SWITCH 8 UP - LOOP ON THIS TEST
```

```
0671   02317 014377          JSB RNG
0672   02320 060134          LDA FL66
0673   02321 002002          SZA
0674   02322 025527          JMP AGAIN      SW 6 WAS UP DURING LOOP
0675   02323 034136          ISZ LCNT         WAS ALWAYS DOWN - STEP COUNT
0676   02324 060136          LDA LCNT
0677   02325 050212          CPA .1500
0678   02326 026330          JMP EOP        END OF PASS IF 1500 LOOPS
0679   02327 025527          JMP AGAIN        NOT END OF PASS - DO NEXT LOOP
0680*
0681* UPDATE END OF PASS COUNT
0682*
0683   02330 064236  EOP     LDB MA77B      ADDR OF LEAST SIGNF CHARS IN MSG
0684   02331 160001  NEXT2   LDA B,I        NEXT 2 ASCII CHARACTERS
0685   02332 002004          INA            STEP LEAST SIGNIFICANT DIGIT
0686   02333 030161          IOR B20        MAKE SURE BASE IS 0, NOT SPACE
0687   02334 170001          STA B,I        PUT BACK IN MSG, TEMPORARILY
0688   02335 010165          AND B77        MASK OFF LOWER CHARACTER
0689   02336 050164          CPA B72        IS IT 10 ?
0690   02337 002001          RSS            YES
0691   02340 026355          JMP DONE       NO - FINISHED
0692   02341 160001          LDA B,I        RESTORE 2 CHARS FROM MESSAGE
0693   02342 040166          ADA B366       STEP UPPER CHARACTER,MAKE LOWER
0694   02343 030174          IOR BIT12      MAKE SURE UPPER BASE IS 0
0695   02344 170001          STA B,I        PUT BACK IN MESSAGE TEMPORARILY
0696   02345 010220          AND D7400      MASK OFF UPPER CHARACTER
0697   02346 050217          CPA D5000      IS IT 10?
0698   02347 002001          RSS            YES
0699   02350 026355          JMP DONE       NO - FINISHED
0700   02351 060213          LDA A60
0701   02352 170001          STA B,I        REPLACE WITH 2 ASCII 0'S
0702   02353 044176          ADB M1         GET ADDR. OF NEXT CHAR. PAIR
0703   02354 026331          JMP NEXT2      STEP AND TEST NEXT CHARACTERS
0704*
0705   02355 014551  DONE    JSB CNEM       CHECK FOR NON-ERROR MESSAGE
0706   02356 026362          JMP EOP1         NO MESSAGE WANTED
0707   02357 060210          LDA .20          PRINT END OF PASS MESSAGE
0708   02360 064235          LDB MA77
0709   02361 114102          JSB LIST,I
0710   02362 124107  EOP1    JMP EXEC,I     CALL DIAGNOSTIC MONITOR
0711   02363 014502  EXRTN   JSB SSR        CHECK SWITCH 12
0712   02364 001710          ALF,SLA
0713   02365 102077          HLT 77B          HALT 77 IF UP
0714   02366 025525          JMP SOP        DO NEXT PASS
0715*
0716   02367         END     EQU *
0717                         END
**  NO ERRORS*
```

```
**ORG   ******** 00003/01 00068/01 00074/01 00077/01 00080/01 00425/02
..B     00061/01 00546/01 00550/01
.1      00146/01 00274/01
.10     00154/01 00453/02
.11     00155/01 00345/02
.1500   00160/01 00677/02
.16     00156/01 00344/01 00228/02 00253/02
.2      00147/01 00281/01 00320/01 00606/01 00660/01 00318/02
.20     00157/01 00707/02
.3      00148/01 00632/01 00645/01 00657/01 00702/01 00342/02 00374/02
.38     00158/01 00660/01
.4      00149/01 00336/01 00452/01 00516/01 00662/01 00663/01 00426/02
.5      00150/01 00388/01 00480/02
.6      00151/01 00511/02
.63     00159/01 00686/01 00700/01
.7      00152/01 00543/02
.8      00153/01 00523/01 00697/01 00205/02 00399/02
.ASL    00107/02 00120/02 00514/02
.ASL1   00109/02 00113/02
.ASL2   00117/02 00124/02
.ASR    00094/02 00105/02 00483/02
.ASR1   00096/02 00102/02
.DIV    00040/02 00066/02 00431/02
.DIV1   00067/02 00079/02
.LSL    00140/02 00148/02 00578/02
.LSL1   00142/02 00145/02
.LSR    00130/02 00138/02 00546/02
.LSR1   00132/02 00135/02
```

```
DLDL    00264/02 00308/02

DLDR    00276/02 00302/02

DONE    00705/02 00691/02 00699/02

DSA     00102/01 00313/02

DST     00053/01 00328/02

DST1    00365/02 00336/02

DST2    00360/02 00339/02

DST3    00357/02 00344/02

DSTA    00329/02 00319/02

DSTC    00317/02 00368/02

DSTE    00369/02 00363/02

DSTL    00312/02 00372/02

DSTR    00323/02 00366/02

EA      00091/01 00598/01 00672/01 00033/02 00049/02 00071/02 00076/02
        00081/02 00103/02 00114/02 00136/02 00146/02 00157/02 00166/02
        00266/02 00324/02

EB      00092/01 00594/01 00670/01 00034/02 00048/02 00056/02 00061/02
        00064/02 00067/02 00075/02 00104/02 00115/02 00137/02 00147/02
        00158/02 00169/02 00268/02 00325/02

EDM     00355/01 00354/01

EDMA    00354/01 00329/01

EE      00093/01 00479/01 00514/01 00694/01

END     00716/02 00075/01

EO      00094/01 00491/01 00521/01 00708/01 00021/02 00085/02 00096/02
        00127/02

EOP     00683/02 00678/02

EOP1    00710/02 00706/02

EOT     00443/01 00447/01 00448/01 00468/01 00385/02 00369/02 00419/02
        00475/02 00506/02 00538/02 00570/02 00602/02 00634/02 00666/02

EOT1    00449/01 00446/01

EOT2    00458/01 00460/01
```

```
LSLC    00578/02 00601/02

LSLE    00602/02 00596/02

LSLI    00585/02 00581/02

LSLL    00577/02 00605/02

LSLR    00582/02 00599/02

LSR     00058/01 00175/01 00553/02

LSR0    00175/01 00548/02

LSR1    00566/02 00558/02

LSRC    00546/02 00569/02

LSRE    00570/02 00564/02

LSRI    00553/02 00549/02

LSRL    00545/02 00573/02

LSRR    00550/02 00567/02

M1      00141/01 00702/02

M16     00144/01 00018/02 00059/02 00175/02

M5      00142/01 00339/01

M6      00143/01 00609/01

MA77    00184/01 00708/02

MA77B   00185/01 00683/02

MG77    00193/01 00184/01

MG77A   00194/01 00247/02 00248/02 00249/02

MG77B   00195/01 00185/01 00250/02

MI      00190/01 00183/01

MIA     00183/01 00254/02

MIND    00092/02 00053/02 00065/02

MOVE    00569/01 00330/01 00337/01 00454/01 00574/01 00634/01 00647/01
        00664/01 00690/01 00704/01 00207/02 00347/02 00401/02 00455/02

MOVEC   00573/01 00581/01

MPY     00054/01 00388/02
```

```
        00072/02 00177/02 00206/02 00267/02 00353/02 00456/02

RNBA    00296/01 00277/01 00304/01 00301/01

RNE     00086/01 00253/01 00477/01

RNG     00230/01 00242/01 00273/01 00288/01 00287/01 00294/01 00193/02
        00264/02 00312/02 00376/02 00430/02 00671/02

RNG1    00274/01 00287/01

RNG2    00281/01 00270/01

RNG3    00288/01 00283/01

RNGBC   00265/01 00260/01

RNM     00087/01 00187/01 00250/01 00254/01 00266/01 00279/01 00286/01
        00011/02 00050/02 00077/02 00404/02 00460/02

RNMA    00187/01 00376/02 00432/02

RNO     00086/01 00257/01 00474/01 00489/01

RRL     00061/01 00178/01 00649/02

RRL0    00178/01 00644/02

RRL1    00662/02 00654/02

RRLC    00642/02 00665/02

RRLE    00666/02 00660/02

RRLI    00649/02 00645/02

RRLL    00641/02 00669/02

RRLR    00646/02 00663/02

RRR     00060/01 00177/01 00617/02

RRR0    00177/01 00612/02

RRR1    00630/02 00622/02

RRRC    00610/02 00633/02

RRRE    00634/02 00626/02

RRRI    00617/02 00613/02

RRRL    00609/02 00637/02

RRRR    00614/02 00631/02
```

```
S6      00187/01 00189/02 00223/02

S6CNT   00188/01 00261/01 00263/01 00394/01

SA      00099/01 00102/01 00321/02 00332/02

SB      00100/01 00322/02 00333/02

SBM     00651/01 00656/01

SBMA    00650/01 00646/01

SHFT    00089/01 00342/01 00173/02 00192/02 00195/02 00197/02 00226/02
        00227/02 00484/02 00515/02 00547/02 00579/02 00611/02 00643/02

SOP     00257/02 00282/02 00714/02

SPC1    00585/01 00588/01 00626/01 00729/01 00730/01

SRMP    00282/02 00284/02 00215/02 00496/02 00528/02 00560/02 00592/02
        00624/02 00656/02

ST1     00240/02 00245/02

START   00237/02 00066/01

STRM    00202/01 00100/01

STRMA   00100/01 00346/02

T1      00110/01 00570/01 00573/01 00610/01 00621/01 00014/02 00023/02
        00091/02 00101/02 00112/02 00123/02 00134/02 00144/02 00155/02
        00166/02 00176/02

T2      00111/01 00571/01 00575/01 00580/01 00017/02 00028/02 00092/02

T3      00112/01 00019/02 00026/02

TST#    00109/01 00331/01 00333/01 00338/01 00455/01 00458/01 00459/01
        00263/02 00311/02 00375/02 00429/02 00481/02 00512/02 00544/02
        00576/02 00608/02 00640/02

*TST10  00575/02

*TST11  00607/02

*TST12  00639/02

*TST2   00310/02

*TST3   00374/02

*TST4   00428/02

*TST5   00480/02
```

*TST6 00511/02

*TST7 00543/02