1.0 IDENTIFICATION

1.1 Maindec 702 Revised

1.2 PDP-7 Extended Checkerboard

1.3 May 16, 1966

2. ABSTRACT

The PDP-7 Extended Checkerboard verifies the performance of from 4096 to 32,768 words of core memory and its associated logic. Memory is exercised in 4096 word increments with four test patterns and their complement patterns. The program resides in the lower addresses of the first 4K of memory and relocates to enable exercising of that region of memory.

3. REQUIREMENTS

3.1 Storage

The program, when initially loaded, occupies locations 0020 to 0474.

3.2 Subprograms (None)

3.3 Equipment

Standard PDP-7 with 4096 to 32,768 words of memory.

3.4 Miscellaneous (Not Applicable)

4. USAGE

4.1 Loading

The binary tape is punched in HRI mode.

AC sw Set the AC switches to 000000_8.

Address sw Set the Address switches to 00020_8.

If extended memory is to be tested set the EXTEND mode switch up; otherwise the switch must be down.

Place the HRI mode binary tape in the reader.

Press READ-IN.

The binary tape is read in HRI mode. At the completion of program load, the test will determine memory size and initiate memory (see section 4.4 for Start-Up and/or Entry).

NOTE: With the EXTEND mode switch down, the PDP-7 Extended Checkerboard may be loaded into any of the extended 8K areas and only that area will be tested. At the time of load, with the EXTEND mode switch down, the ADDRESS switches may be set to any of the following octal settings.

HRI LOAD ADDRESS

00020	TEST	00000 to 17777
20020	TEST	20000 to 37777
40020	TEST	40000 to 57777
60020	TEST	60000 to 77777

With the EXTEND mode switch up, the only address the program may be loaded at is 00020.

4.2 Calling Sequence (Not Applicable)

4.3 Switch Settings

4.3.1 ADDRESS SWITCHES

Loading Address 00020 (or see section 4.1 for loading into extended memory and not in EXTEND mode).

4.3.2 EXTEND Switch

Up - to test extended memory with Checkerboard in the first 4K.

Down - for all other cases.

4.3.3 AC Switches

Up - Suppress writing the corresponding bit as a 1 in memory and do not test that bit for failure.

Down - Include the corresponding bit in generating patterns and test it for failure.

4.4 Start-Up and/or Entry

When the Checkerboard program is read in, it is automatically started at address 0020 by the memory sizing routine. The first 4K of memory tested is addresses 10000 to 17777 (or 0400 to 7777 if only a 4K machine). The program then relocates to starting address 10020 (or 7420 if only a 4K machine) and tests the lowest 4K of memory, addressing 00000 to 07777 (or 0000 to 7377 if only a 4K machine). The Checkerboard will have one of three addresses for restart, depending on machine size and where the program was located at the time it was halted.

Set ADDRESS switches to 00020
7420
or 10020

AC switches - down - to test all bits
- up - to suppress testing

Press START

NOTE: The above addresses are also relative to the 8K of extended memory into which the program is loaded.

4.5 Errors in Usage

The following halts occur in succession for each error:

HALT 1

C(MA) = 00225, 07625 or 10225

C(AC) = contents of failed register

At least one bit in the memory cell just examined was not in its proper state. Each such bit in error appears as the complement of the majority of bits. Suppressed bits appear as 0s even though they might be 1s in the memory location.

For example, if the contents of the AC = 777767, one of two conclusions may be drawn:

Bit 14 was dropped on the first read/complement.

Bit 14 was picked up on the second read/complement.

Or, the contents of the AC = 000010 bit 14 could have been picked up on the first read/complement or dropped on the second.

Press CONTINUE:

HALT 2

C(MA) = 00227, 07627, or 10227

C(AC) = address of the register causing the previous HALT.

Press CONTINUE:

HALT 3

C(MA) = 00231, 07631, or 10231

C(AC) = control word used to generate the current memory pattern.

4.6 Recovery from Such Errors

HALT 1: Press CONTINUE for next HALT

HALT 2: Press CONTINUE for next HALT

HALT 3: Press CONTINUE to resume testing

5. RESTRICTIONS

The EXTEND mode switch must be down if the Checkerboard is to be loaded and run outside the first 8K of memory.

Maindec 703 Memory Address Test should run successfully before this test is attempted on an unknown machine.

6. DESCRIPTION

6.1 General

The PDP-7 Extended Checkerboard is designed to worst-case test core memory in 4K increments with the minimum possibility that the test itself be destroyed by a memory malfunction, and to test memory from minimum to maximum configuration. HRI binary is supplied as an aid to loading the program with a minimum functioning hardware requirement.

At the completion of program load, the memory sizing routine determines if the area the program has been loaded into is 4K or 8K, and adjusts addressing constants if only 4K and Checkerboard is started. If 8K, EXTEND mode is tested and if a 0, the Checkerboard is started. If EXTEND mode is on, the program determines extended memory size and makes two JMP modifications to the basic Checkerboard so that the extended memory test will be moved out of the way and back again while the first 4K of memory is being tested.

Memory is exercised in 4096 word increments. Each 4K of memory is fully tested before the test proceeds to the next 4K. A total of four patterns and their complement patterns are generated, exercised, and tested in each 4K before proceeding to the next 4K.

6.1.1 Pattern Description

The Extended Checkerboard generates and exercises four basic patterns and their complement patterns. These patterns would appear in a bit plane as follows:

Pattern 1:

Y \ X	0	1	2	3
0	1	0	0	1
1	0	1	1	0
2	0	1	1	0
3	1	0	0	1

Y \ X	0	1	2	3
0	0	1	1	0
1	1	0	0	1
2	1	0	0	1
3	0	1	1	0

Pattern 2:

Y \ X	0	1	2	3
0	1	0	0	1
1	1	0	0	1
2	0	1	1	0
3	0	1	1	0

Y \ X	0	1	2	3
0	0	1	1	0
1	0	1	1	0
2	1	0	0	1
3	1	0	0	1

Pattern 3:

Y \ X	0	1	2	3
0	1	1	0	0
1	1	1	0	0
2	0	0	1	1
3	0	0	1	1

Y \ X	0	1	2	3
0	0	0	1	1
1	0	0	1	1
2	1	1	0	0
3	1	1	0	0

Pattern 4:

Y \ X	0	1	2	3
0	1	1	0	0
1	0	0	1	1
2	0	0	1	1
3	1	1	0	0

Y \ X	0	1	2	3
0	0	0	1	1
1	1	1	0	0
2	1	1	0	0
3	0	0	1	1

These patterns are generated by a common routine that produces them by using a different 18-bit control word for each pattern. Bit 0 to bit 15 of the control word generate 16 consecutive pattern words. If a bit is a 0, a word of all 0s is written into the corresponding memory location; if a 1, a word of all 1s is written. In bit 17 of the control word, a 1 indicates that the pattern complements on X addresses 100, 300, 500, and 700 (see patterns 1 and 2). In bit 17 of the control word, a 0 indicates that the pattern complements on X addresses 000, 200, 400, and 600 (see patterns 3 and 4).

The control words used to generate the patterns are as follows:

Pattern 1: 463145

Pattern 2: 631461

Pattern 3: 631460

Pattern 4: 463144

6.1.2 Test Description

Each of the four patterns and its complement pattern is exercised in the same manner. First, the pattern is generated in the memory area being tested. Then, each memory location is individually read, complemented, read, and recomplemented to its origianl value so that the area being tested still contains the entire pattern. The results of the second read are tested for all 1s or all 0s. Then, the pattern is regenerated and each memory register is read; bit 17 is complemented, read, and recomplemented, and the results of the second read are tested for all 0s or all 1s. The pattern is regenerated and bit 16 is tested, then bit 15, and the process is repeated until bit 0 has been tested.

6.2 Applications

The Extended Checkerboard is designed for a minimum of manual intervention in completely exercising all available memory. Most other applications require manual intervention and the changing of the contents of specific memory locations.

Some of these applications can be accomplished as follows:

To Select a Single Pattern:

To exercise memeory with only one of the four patterns it is necessary to change four memory locations so that they contain the same control word, i.e., all four memory locations should equal 463145 to exercise pattern 1 only.

Addresses	Current Contents	Pattern
0271	463145	One
0272	631461	Two
0273	631460	Three
0274	463144	Four

To Prevent Program from Relocating:

At times it may be desirable to exercise a single 4K memory module. In the case of a PDP-7 with only 4K of memory, one may want to only exercise addresses 0400 to 7777 or only addresses 0000 to 7377. This may be accomplished by using one of the following two procedures.

NOTE: The first 4K of memory tested after program load is addresses 10000 to 17777, (or address 0400 to 7777 if only a 4K machine).

To lock the program into exercising only upper addresses:

Press STOP (immediately after loading).

Set ADDRESS switches to 0046, 20046, 40046, or 60046.

Set AC switches to 600020.

Press DEPOSIT (up).

Set AC switches to 000000.

Press START.

After testing the first area, the program relocates itself to starting address 10020 (or 7420 if only 4K). This move can be detected by watching the PROGRAM COUNTER indicators. Bit 2 of the PROGRAM COUNTER will light after the program has been relocated (or bits 3, 4, and 5 if only a 4K machine).

To lock the program into exercising address 00000 to 07777 (or to 7377 if the machine has only 4K)

Load the Extended Checkerboard with all AC switches up.

Watch the PROGRAM COUNTER until the test has relocated itself.

Press STOP.

Set the ADDRESS switches to 10046, 30046, 50046, or 70046 (or if only a 4K machine to 07446).

Set the AC switches to 610020 (or if only a 4K machine to 607420).

Press DEPOSIT (up).

Set AC switches to 000000.

Press START.

Since the Extended Checkerboard can be loaded into and run from extended memory by loading with the EXTEND switch down, these procedures may be used to test any 4K of extended memory.

7. METHODS (Not Applicable)

8. FORMAT (Not Applicable)

9. EXECUTION TIME

Approximately 53 seconds for each 4096 words of memory.

10. PROGRAM

10.1 Core Map (None)

10.2 Dimension List (None)

10.3 Macro, Parameter, and Variable Lists (None)

10.4 Program Listing

```
/PDP-7 MEMORY CHECKERBOARD 4K OR 8K AND EXTENDED MEMORY
```

```
BITCON  266
CHKRBD  20
CHKRLP  32
CNTROL  247
COFINI  232
COLOOP  207
COMEMO  202
DMPEND  474
EGHTKX  460
ENDLTH  364
FOURKX  457
GENPAT  130
GETPAT  260
GNLOOP  152
KCNTR   466
KCOUNT  465
KLENTH  323
K100    262
K10000  467
K16     251
K200    252
K377    253
K400    322
K7400   324
K7777   470
LASTKX  464
LWR     263
MADDRS  242
MIN4    243
MLENTH  267
MOVBAK  442
MOVEK1  471
MOVEK2  472
MOVEK3  473
MOVEUP  427
MSKOUT  245
MSTRPT  271
MVJMPS  455
MVJMP1  100
MVJMP2  103
MVLOOP  46
NDEX    254
NDLOOP  65
NOXTND  375
ONE     265
OPRCON  270
PASSK   244
PATGEN  261
PATLOC  236
SETUP   277
STRTAD  276
SVADRS  255
SVLGTH  256
SVMSTR  240
SXTNKX  462
TEST    104
TSTEND  60
TSTLWR  246
TSTUPR  257
TSXTND  325
TWLVKX  461
TWTYKX  463
UPR     264
UPRPRO  250
UPRSTR  241
XIOTST  377
XINDLP  416
CHKRBD  20
CHKRLP  32
MVLOOP  46
TSTEND  60
NDLOOP  65
MVJMP1  100
MVJMP2  103
TEST    104
GENPAT  130
GNLOOP  152
COMEMO  202
COLOOP  207
COFINI  232
PATLOC  236
SVMSTR  240
UPRSTR  241
MADDRS  242
MIN4    243
PASSK   244
MSKOUT  245
TSTLWR  246
CNTROL  247
UPRPRO  250
K16     251
K200    252
K377    253
NDEX    254
SVADRS  255
SVLGTH  256
TSTUPR  257
GETPAT  260
PATGEN  261
K100    262
LWR     263
UPR     264
ONE     265
BITCON  266
MLENTH  267
OPRCON  270
MSTRPT  271
STRTAD  276
SETUP   277
K400    322
KLENTH  323
K7400   324
TSXTND  325
ENDLTH  364
NOXTND  375
XIOTST  377
XINDLP  416
MOVEUP  427
MOVBAK  442
MVJMPS  455
FOURKX  457
EGHTKX  460
TWLVKX  461
SXTNKX  462
TWTYKX  463
LASTKX  464
KCOUNT  465
```

```
KCNTR     466
K10000    467
K7777     470
MOVEK1    471
MOVEK2    472
MOVEK3    473
DMPEND    474


/PDP-7 MEMORY CHECKERBOARD 4K OR 8K AND EXTENDED MEMORY

20/

CHKRBD,         LAC .
                LAC STRTAD              /GET REGION S.A.
                DAC MADDRS              /TO START TESTING
                LAS
                CMA
                DAC MSKOUT
                LAC MIN4                /-4
                DAC PASSK                       /TO COUNT PATTERNS
                LAC PATLOC
                DAC GETPAT              /TO GET PATTERNS

CHKRLP,         LAC I GETPAT            /GET NEXT CHECKERBOARD
                DAC CNTROL              /FOR GENERATING
                JMS TEST                /TEST CURRENT REGION
                ISZ GETPAT
                ISZ PASSK                       /DONE ALL 4 CHECKERBOARDS
                JMP CHKRLP              /NO, DO NEXT
                DZM MSKOUT
                DZM NDEX
                LAC CHKRBD              /GET SA OF THIS PROGRAM
                DAC GETPAT              /FOR INDIRECTS

/MOVE THE PROGRAM TO TEST MEMORY NOT TESTED ON THE LAST PASS
                XOR UPRPRO              /CHANGE S.A. TO NEXT TEST
                DAC SVADRS              /FOR INDIRECTS
MVLOOP,         LAC I GETPAT            /GET NEXT INSTR
                AND OPRCON              /740000
                SAD OPRCON              /IS IT AN OPERATIVE INSTR
                JMP TSTEND              /MAYBE
                LAC I GETPAT
                XOR UPRPRO              /CHANGE ADDRESS OF REF.

                DAC I SVADRS
                ISZ SVADRS              /ADVANCE
                ISZ GETPAT              /ADDRESSES
                JMP MVLOOP
TSTEND,         LAC I GETPAT
                DAC I SVADRS            /STORE OPERATIVE GROUP
                CMA
                SZA                     /DELIMITER
                JMP TSTEND-3            /NO
```

```
NDLOOP,         ISZ GETPAT
                ISZ SVADRS
                LAC I GETPAT            /GET NEXT CONSTANT
                DAC I SVADRS
                CMA
                SZA CLA-OPR             /2ND DELIMITER
                JMP NDLOOP              /NO

                ISZ SVADRS              /ADVANCE FOR STORE S.A.
                DAC I SVADRS            /MAKE SA=LWR START
                SAD MADDRS              /THIS TEST TESTING LWR
                SKP                     /YES, NEXT PASS TEST UPR
                                        /HAVE EXTENDED MEMORY
MVJMP1,         JMP I TSTLWR            /GO TEST LWR MEMORY OR JMP MOVEUP
                LAC UPRSTR
                DAC I SVADRS            /MAKE S.A. = UPR START

MVJMP2,         JMP I TSTUPR            /GO TEST UPPER MEMORY OR JMP MOVBAK
                                        /MOVE EXTENDED CHECKERBD BACK

/CONNECT SUBROUTINES TO FORM TEST

TEST,           JMP .
                JMS GENPAT                      /GENERATE PATTERN
                CLA CMA-OPR
                DAC BITCON                      /COMPLIMENT WHOLE WORDS
                JMS COMEMO                      /COMPLIMENT AND COMPARE
                LAC ONE
                DAC BITCON                      /COMPLIMENT SINGLE BITS
                JMS GENPAT                      /GENERATE
                JMS COMEMO                      /COMPLIMENT AND COMPARE
                LAC BITCON
                CLL RAL-OPR                     /NEXT BIT
                DAC BITCON
                SZA                     /DONE ALL
                JMP .-6                 /TEST NEXT BIT
                LAC MIN4
                XOR I GETPAT
                SAD CNTROL                      /DONE COMPLIMENT PATTERN
                JMP I TEST                      /YES
                DAC CNTROL
                JMP TEST+1

/PATTERN GENERATION VARIABLE LENGTH
/AND PATTERN CONTROLLED BY INITIAL
/CONTENTS OF CONTROL

GENPAT,         JMP .
                LAC CNTROL              /GET MASTER PATTERN WORD
                DAC SVMSTR
                RAR
                LAC K100
                SNL
                CLA
                DAC LWR
                TAD K200
                DAC UPR
```

```
                LAC MADDRS
                DAC SVADRS          /SAVE STARTING ADDRESS
                LAC MLENTH
                DAC SVLGTH          /NUMBER OF WORDS

                LAC K16             /-16 DECIMAL
                DAC NDEX
                LAC SVMSTR
                DAC PATGEN          /RESET FOR PATTERN
GNLOOP,         LAC PATGEN
                CLL RAL-OPR         /NEXT BIT TO LINK
                DAC PATGEN          /SAVE REST
                SZL CLA-OPR         /SHOULD NEXT WORD = 0

                CMA                 /NO
                AND MSKOUT          /CLR NOT SELECT
                DAC I SVADRS
                ISZ SVADRS          /GENERATE NEXT ADDRESS
                ISZ SVLGTH          /DONE WHOLE PATTERN
                SKP
                JMP I GENPAT        /EXIT
                ISZ NDEX            /DONE 16 WORDS
                JMP GNLOOP          /NO

                LAC SVADRS
                AND K377            /CLR NEXT ADDRS TO LWR 8
                SAD LWR
                JMP .+4             /IF = 0
                SAD UPR             /OR = 200
                SKP                 /COMPLIMENT MASTER PAT
                JMP GNLOOP-4
                LAC SVMSTR
                CMA
                DAC SVMSTR
                JMP GNLOOP-4        /DO NEXT 200

/COMPARE MEMORY FOR FAILURE

COMEMO,         JMP .
                LAC MADDRS
                DAC SVADRS          /SAVE STARTING ADDRESS
                LAC MLENTH
                DAC SVLGTH          /SAVE LENGTH
COLOOP,         LAC BITCON
                XOR I SVADRS
                DAC I SVADRS
                LAC BITCON
                XOR I SVADRS
                DAC I SVADRS

                AND MSKOUT          /CLR NOT SELECTED BITS
                SZA                 /IF -

                CMA                 /MAKE +
                AND MSKOUT          /CLR AGAIN
                SNA                 /RESULT SHOULD BE 0
                JMP COFINI
```

```
                CMA
                AND MSKOUT
                HLT
                LAC SVADRS                      /DISPLAY FAILED BIT 8S?
                HLT
                LAC CNTROL
                HLT                             /DISPLAY PATTERN CON.WORD
COFINI,         ISZ SVADRS
                ISZ SVLGTH                      /DO ALL
                JMP COLOOP                      /NO
                JMP I COMEMO


PATLOC,         MSTRPT

/CONSTANTS NOT CHANGED BY MOVING THE PROGRAM
/AND TEMP STORE REGISTERS

                LAM                             /DELIMITER

SVMSTR,         0                               /TO GENERATE PATTERNS
UPRSTR,         10000                           /S.A. TO TEST UPR MEM
MADDRS,         0                               /TEMP STOR CURRENT REGION
MIN4,           777774                          /-4
PASSK,          0                               /TO COUNT PATTERNS
MSKOUT,         0                               /TO NOT COMPARE BITS
LSTLWR,         10000+CHKRBD                    /TO START TEST ON LOWER
CNTROL,         0                               /TO STORE CURRENT PATTERN
UPRPRO,         10000                           /MASK TO MOVE PROGRAM
K16,            777760                          /TO COUNT WORDS IN GENPAT
K200,           200                             /TO FORM 200 OR 300
K377,           377                             /TO MASK ADDRESSES
NDEX,           0                               /FOR COUNTING
SVADRS,         0                               /FOR INDIRECTS
SVLGTH,         0                               /TO COUNT MEM LENGTH
LSTUPR,         CHKRBD                          /TO START TEST ON UPR
GETPAT,         0                               /FOR INDIRECTS
PATGEN,         0                               /USED IN GENERATING
K100,           100                             /TO FORM 100 AND 300
LWR,            0                               /= 0 OR 100
UPR,            0                               /= 200 OR 300
ONE,            1
BITCON,         0
MLENTH,         770000                          /LENGTH OF MEM TO TEST
UPRCON,         740000
MSTRPT,         463145                          /GENERATE PATTERN 1
                631461                          /GENERATE PATTERN 2
                631460                          /GENERATE PATTERN 3
                463144                          /GENERATE PATTERN 4

                LAM                             /DELIMITER

STRTAD,         10000
```

```
/TEST MEMORY LENGTH AND ADJUST
/TEST CONSTANTS IF ONLY 4K

SETUP,          CAF
                DZM 7777
                CLA CMA-OPR
                DAC 17777
                SAD 7777                /IS MEM 4K OR 8K
                JMP .+4                 /4K
                SEM
                JMP CHKRBD
                JMP TSXTND                      /8K, SEE IF MEMORY IS EXTENDED
                LAC K400                /CHANGE CONSTANTS

                DAC STRTAD              /TO TEST 4K MACHINE
                DAC UPRSTR

                LAC KLENTH
                DAC MLENTH

                LAC K7400
                DAC UPRPRO
                TAD TSTUPR
                DAC TSTLWR

                JMP CHKRBD
K400,           400
KLENTH,         770400                  /4K - 400
K7400,          7400                    /MOVE

/EXTENDED MEMORY CHECKERBOARD
/DETERMINE MEMORY LENGTH

TSXTND,         CLA
                DAC 7777
                DAC 17777
                DAC KCOUNT
                CMA
                EEM
                DAC I FOURKX
                SAD 7777
                JMP NOXTND
                ISZ KCOUNT              /INDICATE EXTENDED 4K
                DAC I EGHTKX
                SAD 17777
                JMP FNDLTH              /12K MACHINE (EXTRA 4K)
                ISZ KCOUNT              /INDICATE EXTENDED 8K
                DAC I TWLVKX
                SAD 7777

                JMP FNDLTH              /16K MACHINE (EXTRA 8K)
                ISZ KCOUNT              /INDICATE EXTENDED 12K
                DAC I SXTNKX
                SAD 17777
                JMP FNDLTH              /20K MACHINE (EXTRA 12K)
                CLA
                ISZ KCOUNT              /INDICATE EXTENDED 16K
```

```
                DZM I TWTYKX
                SAD I FOURKX
                JMP FNDLTH          /24K MACHINE (EXTRA 16K)
                ISZ KCOUNT
                DZM I LASIKX
                SAD I EGHTKX
                JMP FNDLTH          /28K MACHINE (EXTRA 20K)
                ISZ KCOUNT          /INDICATE 32K OF CORE

FNDLTH,         CLA CMA-OPR
                TAD KCOUNT
                CMA                 /MAKE 2'S COMPLIMENT
                DAC KCOUNT          /OF 4K EXTENDED MODULES
                LAC MVJMPS
                DAC MVJMP1
                LAC MVJMPS+1
                DAC MVJMP2
                JMP CHKRBD
NOXTND,         LEM                 /CLEAR EXTEND MODE
                JMP CHKRBD                  /NO EXTENDED MEMORY
XTDTST,         LAC KCOUNT
                DAC KCNTR
                LAC K10000
                DAC MADDRS
                LAC MADDRS
                TAD K7777
                DAC MADDRS          /GENERATE S.A. FOR
                ISZ MADDRS          /NEXT 4K MODULE
                LAC MIN4            /TO TEST
                DAC PASSK
                LAC PATLOC
                DAC GETPAT
                LAS
                CMA
                DAC MSKOUT
XTNDLP,         LAC I GETPAT
                DAC CNTROL
                JMS TEST
                ISZ GETPAT
                ISZ PASSK
                JMP XTNDLP
                ISZ KCNTR                   /DONE ALL EXTENDED
                JMP XTDTST+4        /NO, DO NEXT 4K
                JMP CHKRBD          /YES, TEST 10000 TO 17777
/MOVE THE EXTENDED MEMORY CHECKERBOARD
/OUT OF THE WAY TO TEST BLOCK 0000 TO 7777

MOVEUP,         LAC MOVEK1
                DAC 10
                LAC MOVEK2
                DAC 11
                LAC MOVEK3
                DAC SVLGTH
                LAC I 10
                DAC I 11
                ISZ SVLGTH
                JMP .-3
                JMP I TSTLWR
```

```
/MOVE THE EXTENDED MEMORY CHECKERBOARD
/BACK TO FIRST 4K

MOVBAK,         LAC MOVEK2+10000        /THIS ROUTINE IS
                DAC 10                  /EXECUTED OUT
                LAC MOVEK1+10000        /OF 2ND 4K OF
                DAC 11                  /NOT EXTENDED
                LAC MOVEK3+10000        /MEMORY
                DAC SVLGTH+10000
                LAC I 10
                DAC I 11
                ISZ SVLGTH+10000
                JMP .-3+10000
                JMP XTDTST

MVJMPS,         JMP MOVEUP
                JMP MOVBAK

FOURKX,         27777                   /TO GET TO FIRST 4K EXTENDED
EGHTKX,         37777                   /DITTO  2ND 4K EXTENDED
TWLVKX,         47777                   /3RD
SXTNKX,         57777                   /4TH
TWTYKX,         67777                   /5TH
LASTKX,         77777                   /6TH, OR 32K
KCOUNT,         0
KCNTR,          0

K10000,         10000
K7777,          7777
MOVEK1,         XTDTST-1
MOVEK2,         XTDTST+7777
MOVEK3,         -MOVEK3+XTDTST


DMPEND,         JMP SETUP

START
```

10.5 Octal Dump

```
00000   TO 00006    000000
00007   010153
00010   TO 00017    000000
00020   200020
00021   200276
00022   040242
00023   750004
00024   740001
00025   040245
00026   200243
00027   040244
00030   200236
00031   040260
00032   220260
00033   040247
00034   100104
00035   440260
00036   440244
00037   600032
00040   140245
00041   140254
00042   200020
00043   040260
00044   240250
00045   040255
00046   220260
00047   500270
00050   540270
00051   600060
00052   220260
00053   240250
00054   060255
00055   440255
00056   440260
00057   600046
00060   220260
00061   060255
00062   740001
00063   740200
00064   600055
00065   440260
00066   440255
00067   220260
00070   060255
00071   740001
00072   750200
00073   600065
00074   440255
00075   060255
00076   540242
00077   741000
00100   620246
00101   200241
00102   060255
00103   620257
00104   600104 ~020035
00105   100130
00106   750001
00107   040266
00110   100202
00111   200265
00112   040266
00113   100130
00114   100202
00115   200266
00116   744010
00117   040266
00120   740200
00121   600113
00122   200243
00123   260260
00124   540247
00125   620104
00126   040247
00127   600105
00130   600131 ~020114
00131   200247
00132   040243
00133   740020
00134   200262
00135   740400
00136   750000
00137   040263
00140   340252
00141   040264
00142   200242
00143   040255
00144   200267
00145   040256
00146   200251
00147   040254
00150   200240
00151   040261
00152   200261
00153   744010
00154   040261
00155   751400
00156   740001
00157   500245
00160   060255
00161   440255
00162   440256
00163   741000
00164   620130
00165   440254
00166   600152
00167   200255
00170   500253
00171   540263
00172   600176
```

```
00173   540264
00174   741000
00175   600146
00176   200240
00177   740001
00200   040240
00201   600146
00202   600202
00203   200242
00204   040255
00205   200267
00206   040256
00207   200266
00210   260255
00211   060255
00212   200266
00213   260255
00214   060255
00215   500245
00216   740200
00217   740001
00220   500245
00221   741200
00222   600232
00223   740001
00224   500245
00225   740040
00226   200255
00227   740040
00230   200247
00231   740040
00232   440255
00233   440256
00234   600207
00235   620202
00236   000271
00237   777777
00240   000000
00241   010000
00242   000000
00243   777774
00244   TO 00245    000000
00246   010020
00247   000000
00250   010000
00251   777760
00252   000200
00253   000377
00254   TO 00256    000000
00257   000020
00260   TO 00261    000000
00262   000100
00263   TO 00264    000000
00265   000001
00266   000000
00267   770000
00270   740000
00271   463145
00272   631461
00273   631460
00274   463144
00275   777777
00276   010000
00277   703302
00300   147777
00301   750001
00302   057777
00303   547777
00304   600310
00305   707701
00306   600020
00307   600325
00310   200322
00311   040276
00312   040241
00313   200323
00314   040267
00315   200324
00316   040250
00317   340257
00320   040246
00321   600020
00322   000400
00323   770400
00324   007400
00325   750000
00326   047777
00327   057777
00330   040465
00331   740001
00332   707702
00333   060457
00334   547777
00335   600375
00336   440465
00337   060460
00340   557777
00341   600364
00342   440465
00343   060461
00344   547777
00345   600364
00346   440465
00347   060462
00350   557777
00351   600364
00352   750000
00353   440465
00354   160463
00355   560457
```

```
00356   600364
00357   440465
00360   160464
00361   560460
00362   600364
00363   440465
00364   750001
00365   340465
00366   740001
00367   040465
00370   200455
00371   040100
00372   200456
00373   040103
00374   600020
00375   707704
00376   600020
00377   200465
00400   040466
00401   200467
00402   040242
00403   200242
00404   340470
00405   040242
00406   440242
00407   200243
00410   040244
00411   200236
00412   040260
00413   750004
00414   740001
00415   040245
00416   220260
00417   040247
00420   100104
00421   440260
00422   440244
00423   600416
00424   440466
00425   600403
00426   600020
00427   200471
00430   040010
00431   200472
00432   040011
00433   200473
00434   040256
00435   220010
00436   060011
00437   440256
00440   600435
00441   620246
00442   210472
00443   040010
00444   210471
00445   040011
00446   210473
00447   050256
00450   220010
00451   060011
00452   450256
00453   610450
00454   600377
00455   600427
00456   600442
00457   027777
00460   037777
00461   047777
00462   057777
00463   067777
00464   077777
00465   TO 00466    000000
00467   010000
00470   007777
00471   000376
00472   010376
00473   777703
00474   600277
00475   TO 07777    002000
```