

8/16E MEMORY PROTECT TEST

Consists of:

| | |
|------------------------------|------------------------|
| Program Description | 06-223M95R01A15 |
| Program Listing | 06-223M96R01A13 |
| Bootstrap Object Tape | 06-223M17R01 |

PERKIN-ELMER

Computer Systems Division
2 Crescent Place
Oceanport, N.J. 07757

8/16 E MEMORY PROTECT TEST PROGRAM DESCRIPTION

1. 8/16 E MEMORY PROTECT TEST

1.1 RELATED DOCUMENTS

| | |
|--|---------------|
| Test Program Listing | 06-223M96 R01 |
| Test Program Paper Tape | 06-223M17 R01 |
| 8/16 E Memory Protect Controller (MPC) Programming Manual | 29-622 |

1.2 Test Program

| | |
|--------------------------------|-------------------|
| Memory Test | 06-221 |
| Processor Test | 06-211 and 06-212 |
| Teletype Basic Confidence Test | 06-004 |
| CRT Test | 06-146 |

2. PROGRAM

2.1 PURPOSE

The 8/16 E Memory Protect Test Program tests the 8/16 E Memory Protect Controller (02-524) operating in an 8/16 E, 16-bit extended memory system. The program ensures that the 8/16 E Memory Protect Controller fulfills the requirements of the design specification. Malfunctions that exist in the controller are detected and the results are displayed on the console device.

2.2 TESTS

The program consists of seven tests:

1. TEST 1

This test ensures that the memory protect controller is inactive when PSW 7 = 0 or when an OFF command is issued. It also ensures that interrupts can be disarmed.

2. TEST 2

This test ensures that memory can be write protected and write/read protected.

3. TEST 3

This test ensures that program memory can be write protected and write/read protected.

4. TEST 4

This test ensures that memory can be execute protected.

5. TEST 5

This test ensures that memory can be write, read and execute protected.

6. TEST 6

This test ensures that program memory can be execute protected.

7. TEST 7

This test ensures that memory can be protected with PSW 8 = 1.

2.3 Test Design Specification

The integrity of the memory protect controller is demonstrated by protecting one or more blocks of memory at a given time. These blocks may be protected from write, read and/or execute operations. Subsequent to the setting of the protection, all blocks of memory are tested for protection. Protect violations create an I/O interrupt and update the status of the controller.

2.4 Test Options

Two types of strap options are available for the memory protect controller: (1) size of memory blocks, 512, 1024 or 2048 bytes, and (2) model type, 8/16 E or 70. The Model 70 option is used when compatibility with Model 70 software is required. This option can protect memory from write operations only and it is used normally within 64KB of memory. The Model 8/16 E option can protect memory from write, read and execute operations and it is used with the full memory range of an 8/16 E system.

2.5 Test Limitation

The extended memory address lines in the controller are exercised completely in a system containing 256KB of memory. For the Model 70 option, these address lines are not exercised since Model 70 programming precludes the use of extended memory.

3. MINIMUM HARDWARE REQUIRED

The hardware required, as a minimum, to perform this test is:

1. Processor - Model 8/16 E or equivalent
2. Minimum Memory - 16KB (Test 7 requires 48 KB)
3. Console Input Device - Teletype or CRT on PASLA (See Appendix A).
4. Paper Tape Reader - Teletype or High Speed Paper Tape Reader.

4. REQUIREMENTS OF MACHINE UNDER TEST

This program assumes that the programs listed in Section 1.2 have been run without detecting an error.

4.1 Device Specification

The address of the memory protect controller, the size of the memory block and the model type (8/16 E) must be specified through the option commands (Appendices B and D). Alternatively, the program uses the default values of these options.

5. LOADING PROCEDURES

5.1 Test Tape Format

Absolute, non-zoned object tape (M17) with front-end bootloader.

5.2 Memory Occupied

The test program occupies memory from X'A00' through '2287'.

5.3 Normal Loading Procedures

1. Manually enter the X'50' sequence shown below into memory.

| | <u>LOCATION</u> | <u>CONTENTS</u> |
|-------------|-----------------|-----------------|
| | X'30' | X'0000' |
| | X'32' | X'0000' |
| | X'34' | X'0000' |
| | X'36' | X'0050' |
| | X'50' | X'D500' |
| | X'52' | X'00CF' |
| | X'54' | X'4300' |
| | X'56' | X'0080' |
| For TTY | X'78' | X'0294' |
| For HSPTR | X'78' | X'0399' |
| For HSPTR/P | X'79' | X'1399' |

2. Place the program in the paper tape reader.
3. Execute at address X'30'.
4. When the processor halts, observe console display registers D1 and D2. If they are ZERO, loading is complete; otherwise, repeat loading procedure.

5. Refer to Appendix A and set up the address for the console device.
6. Address memory location X'A00'.
7. Start program execution. Observe that the following title is displayed on the console device:

8/16 E MEMORY PROTECT TEST 06-223R01

TOP OF MEMORY Z XXXX

The program calculates the last memory location and displays this location in hexadecimal value below the title of the test. This is an 18 bit address which reflects up to 256KB of memory.

Z = The two most-significant bits of the address.

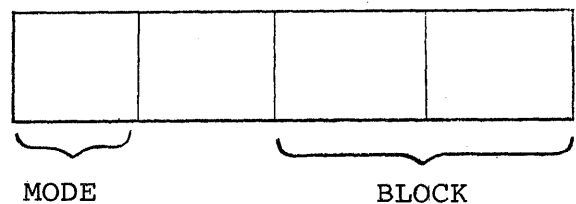
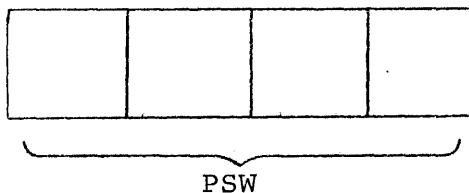
XXXX = The remaining 16 bits of address.

See Appendix F if the top of memory is to be modified.

6. OPERATING PROCEDURES

6.1 Normal Testing

1. When the asterisk is printed, enter the desired options via the console device. Refer to Appendix B for the console device command structure. Refer to Appendix D for the option explanation.
2. Enter the RUN command via the console device.
3. Each test selected is executed. If no error are detected, the message NO ERROR is printed. Should an error occur, refer to Section 6.3 for the appropriate section.
4. Any test or tests may be executed by entering the appropriate test number via the TEST option followed by the RUN command. (For Model 70 use, only Test 1, 2, and 3 are to be executed).
5. During the execution of the tests, the following information is displayed continuously on the display panel.



- A) The first four hexadecimal characters represent the current PSW.
- B) The fifth character, the mode of the controller:
 - 0 = write protect
 - F = write and read protect
- C) The last two characters, the block number of the first protected block in the map.

6.2 Optional Testing

1. Certain test options may be modified for further testing. See Appendix D for available options.
2. To inhibit all printouts and run the selected tests continuously, the console device (Teletype only) can be turned OFF. When this is done, the program counts the number of times the test is repeated in the memory location labeled TOTAL. If an error is detected, the count in the memory location labeled TOTALERR is incremented. The contents of TOTAL are continuously copied into the console panel display.

6.3 Error Procedures

6.3.1 Error Messages

When the program detects an error, an error message is displayed on the console device. The error message consists of an error number as shown below:

ERROR XXYY

where: XX is the test number in which the error is detected.

YY is the error number.

In addition to the test and error numbers, some additional data such as device code and status, may also be displayed depending on the error encountered. (See Appendix E.) The largest error number is 26.

After an error is encountered, it may be useful to know the current protect status of each block of memory. This information is contained in the memory protect map which may be displayed on the console device by issuing the MAP command.

6.3.2 Machine Malfunction

If a machine malfunction interrupt is generated, the following printout results:

MACHINE MALFUNCTION

XXYY

where: X = the condition code, CVGL, when the interrupt occurs.

YYYY = the location at which the interrupt occurred. Upon completion of this message the processor is placed in the wait state.

If the console device (Teletype only) is OFF when the interrupt is generated, X'XXXX' is written on the display and the processor is placed in the wait state. To continue test execution, depress the RUN (EXECUTE) switch on the display.

6.3.3 Illegal Instruction

Two types of illegal instruction interrupts used in this program are: the interrupt caused by an execute protect violation, which results in the status of the memory protect controller being sensed and the execute interrupt counter being incremented. The program continues normal execution in this case.

The second type of interrupt is caused by an actual illegal instruction encountered in the program. In this case, the following message is displayed:

ILLEGAL INSTRUCTION

XXXX XXXX

where: XXXX XXXX = the PSW when the interrupt occurred. Upon completion of the message, the processor is placed in the wait state.

If the console device (Teletype only) is OFF when the interrupt is generated, X'5555' is written on the display and the processor is placed in the wait state.

To continue test execution, depress the RUN switch on the display.

7. PROGRAMMING NOTES

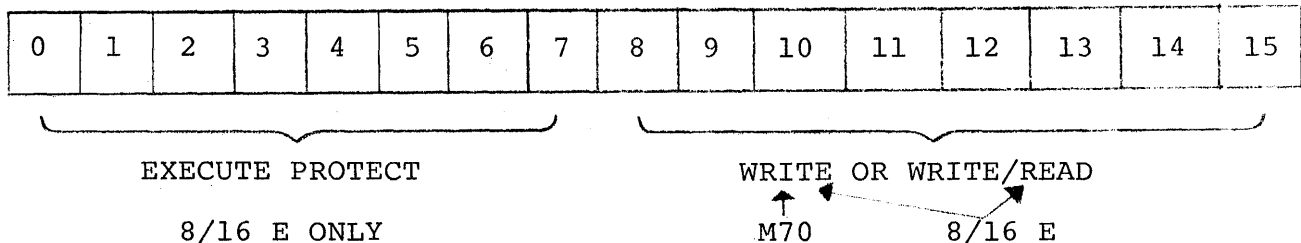
7.1 MEMORY PROTECTION

Memory is protected in blocks of 512, 1024, or 2048 bytes. Appendix G shows the relationship between block number and memory address. Each block of memory can be write protected or write and read protected depending on the protection mode of the controller. The controller is in

the write and read protect mode when the W/R bit of the command byte is set; otherwise, it is in the write protect mode. (See Appendix J.) In addition, the blocks can be execute protected, an independent protection mode. To protect a block of memory, the bit corresponding to the particular block number must be set in the halfword of the memory protect map.

7.2 MEMORY PROTECT MAP

The memory protect map consists of eight halfwords. Bits 0 - 7 of each halfword are used to set execute protection. Bits 8 - 15 are used to set write protection or write-read protection.



Appendix I shows the relationship between bit positions in the map and the block number.

The memory protect map is loaded in the controller by issuing eight write halfword instructions. For Model 70 use, only the write protect bits are loaded in the controller by eight write data instructions.

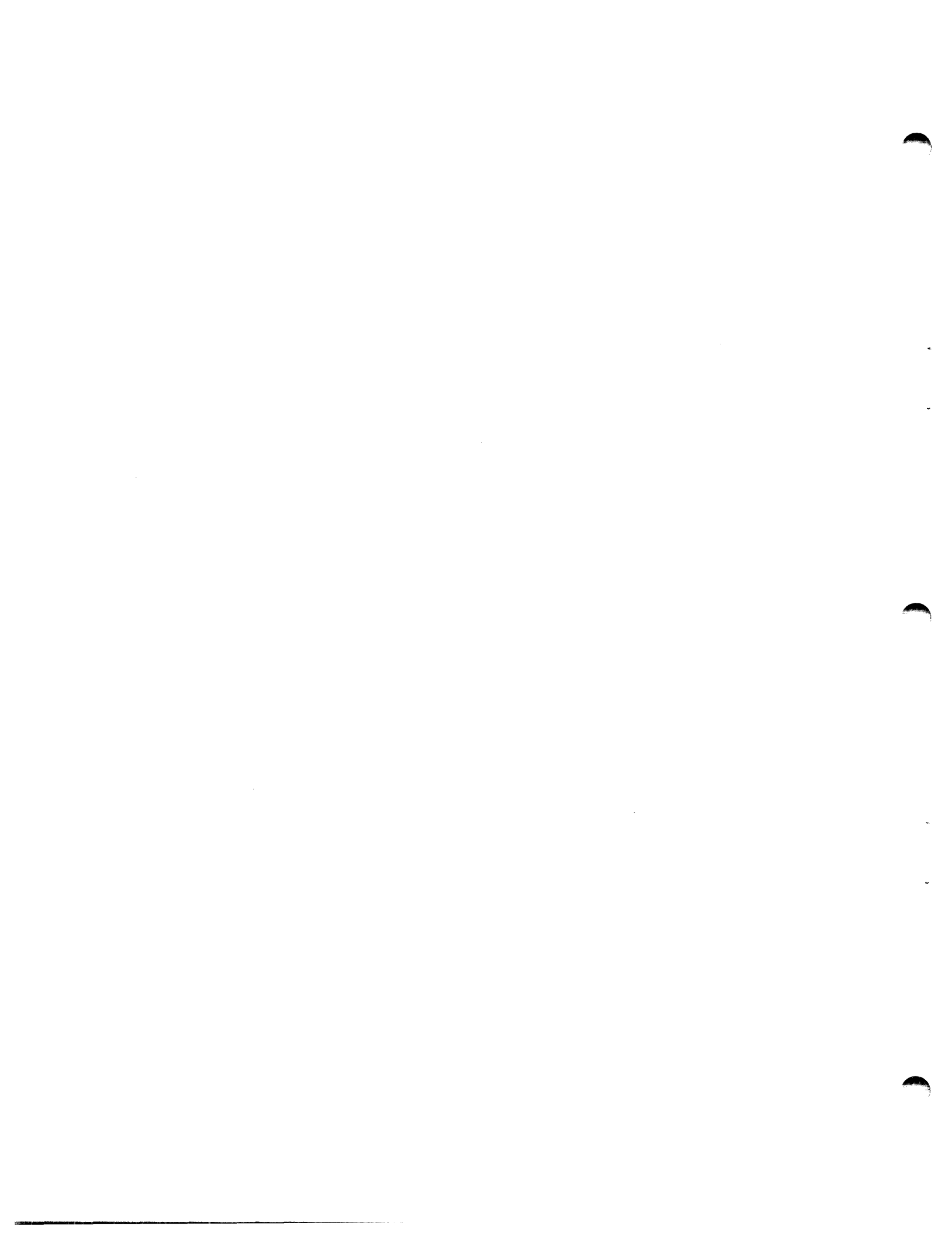
7.3 EXTENDED MEMORY

A 16-bit, extended memory system may contain up to 256KB of memory and it requires 18 bits to address. The memory is composed of a maximum of four segments, each containing 64KB of memory, with the exception of the last segment which may contain 32KB of memory.

Appendix H shows the translation between program address and physical address.

7.4 INTERRUPT

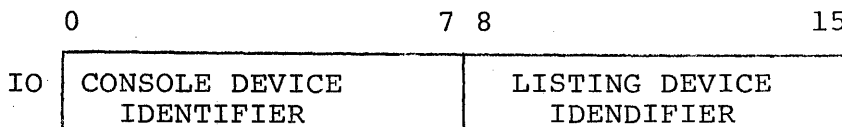
A write or read protect violation generates an I/O interrupt and sets the corresponding protect violation bits in the status byte. An execute protect violation creates an illegal instruction interrupt followed by an I/O interrupt. When an interrupt occurs during the test, the status of the controller is sensed, the interrupt acknowledged and the appropriate interrupt counter incremented. External interrupts are enabled during testing.



APPENDIX A

USER DEVICE DEFINITION

The first byte of memory labeled IO (see the listing) has a default value for the Teletype as an input/output console device. The second byte has a default value for no separate listing device. If the set-up is different, it must be changed as follows:



| CONSOLE DEVICE IDENTIFIER | EXPLANATION |
|------------------------------|--|
| X'01' | GDT/CRT on PASLA/PALM interface, strapped for FDX and the highest baud rate. |
| X'02' | TTY on TTY interface GDT/CRT on current loop interface. HDX |
| X'05' | Micro I/O bus interface. |
| 0, X'03', X'04', X'06'-X'FF' | Reserved. The program defaults it to 2. |

| LISTING DEVICE IDENTIFIER | EXPLANATION |
|---------------------------|--|
| X'00' | No separate listing device. |
| X'01' | GDT/CRT on PASLA/PALM interface, strapped for FDX and the highest baud rate. |
| X'02' | TTY on TTY interface GDT/CRT on current loop interface. HDX |
| X'03' | Line Printer |
| X'05' | Micro I/O bus interface. |
| X'04', X'06'-X'FF' | Reserved. The program defaults it to 2. |

The GDT (Graphic Display Terminal) or CRT, if used on the PASLA/PALM interface should be strapped for a device address of X'10' and X'11' for the receiving and transmitting sides, respectively. If it is different, the first byte of label PASLADR (see the listing) must be changed to the address of the receiving side of the console device and/or the first byte of the label PASLADR2 must be changed to the address of the receiving side of the listing device.

APPENDIX A (Continued)

The teletype or current loop interface, if used, should be strapped for device address of X'02'. If it is different, the first byte of label TTYADR must be changed to the address of the console device and/or the second byte must be changed to the address of the listing device.

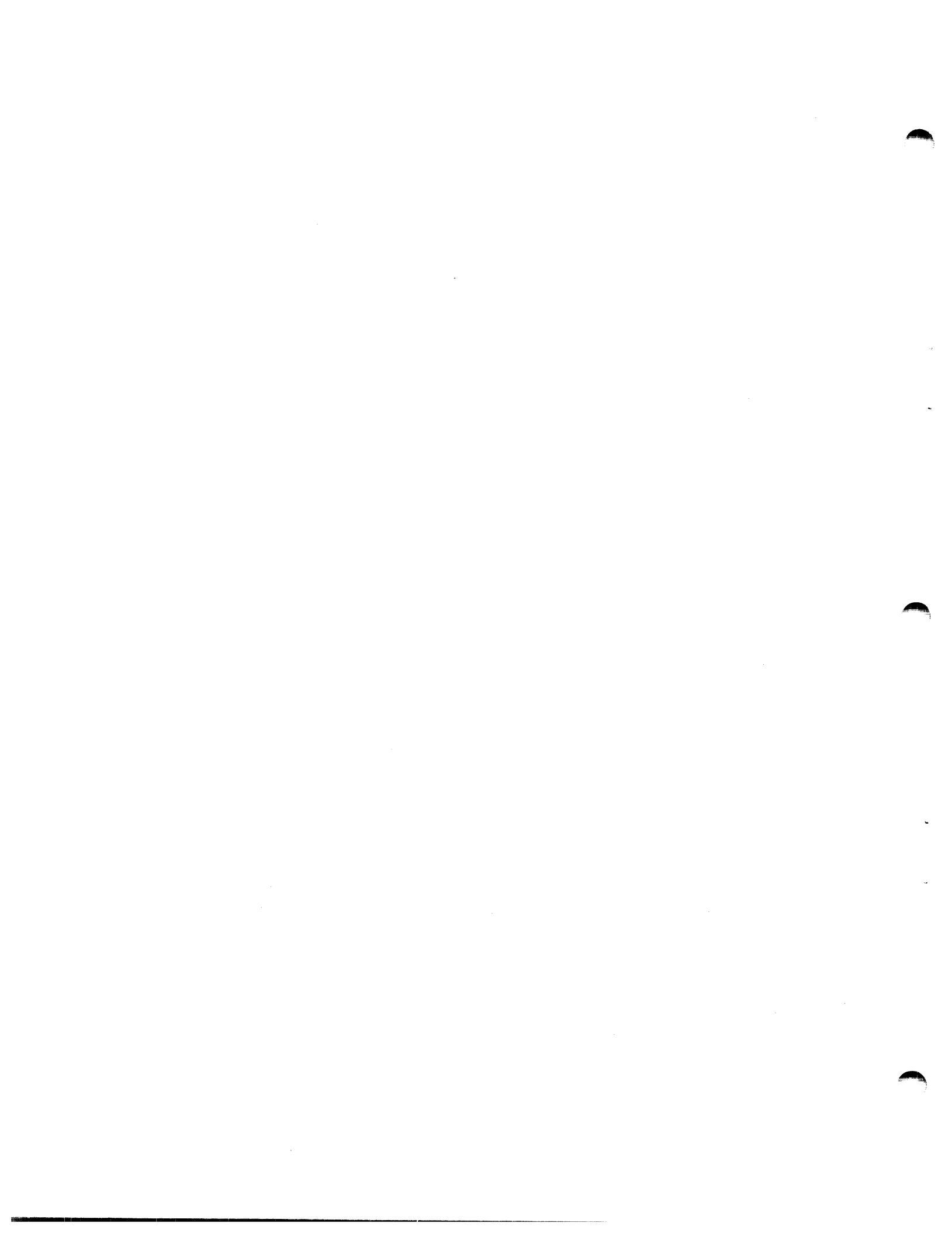
The line printer, if used, should be strapped for a device address of X'62'. If it is different, the second byte of label LNPADR must be changed accordingly.

The Micro I/O bus interface, if used, should be strapped for device address X'CO'. If different, the location 'MICADR' should be changed accordingly.

APPENDIX B

OPTION/COMMAND INPUT STRUCTURE

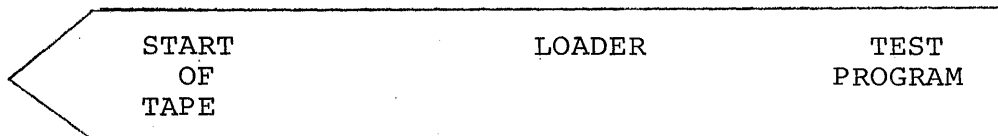
An asterisk (*) is printed on the console device to indicate that the program is awaiting an option input. Any option may then be typed in from the console input device followed by a space and the desired hex value; an exception is the test option which accepts arguments separated by commas. A carriage return (CR) is issued to terminate every option input. An invalid command or value causes a "?" followed by a carriage return (CR), line feed (LF) and an asterisk (*) to occur.



APPENDIX C

LOADER

The loader must be loaded using the 50 sequence as described in Section 5.3. The loader resides in memory from X'80' to X'CF' and loads the test program starting at location X'A00'. While reading the program tape, each data byte location is output to the display panel. While loading the test into memory, it performs an exclusive OR of each instruction to verify that the test loaded correctly. If the test loads correctly, the loader zeros display registers D1 and D2 and halts the processor. The loading procedures in Section 5.3 must be repeated if the test did not load correctly.



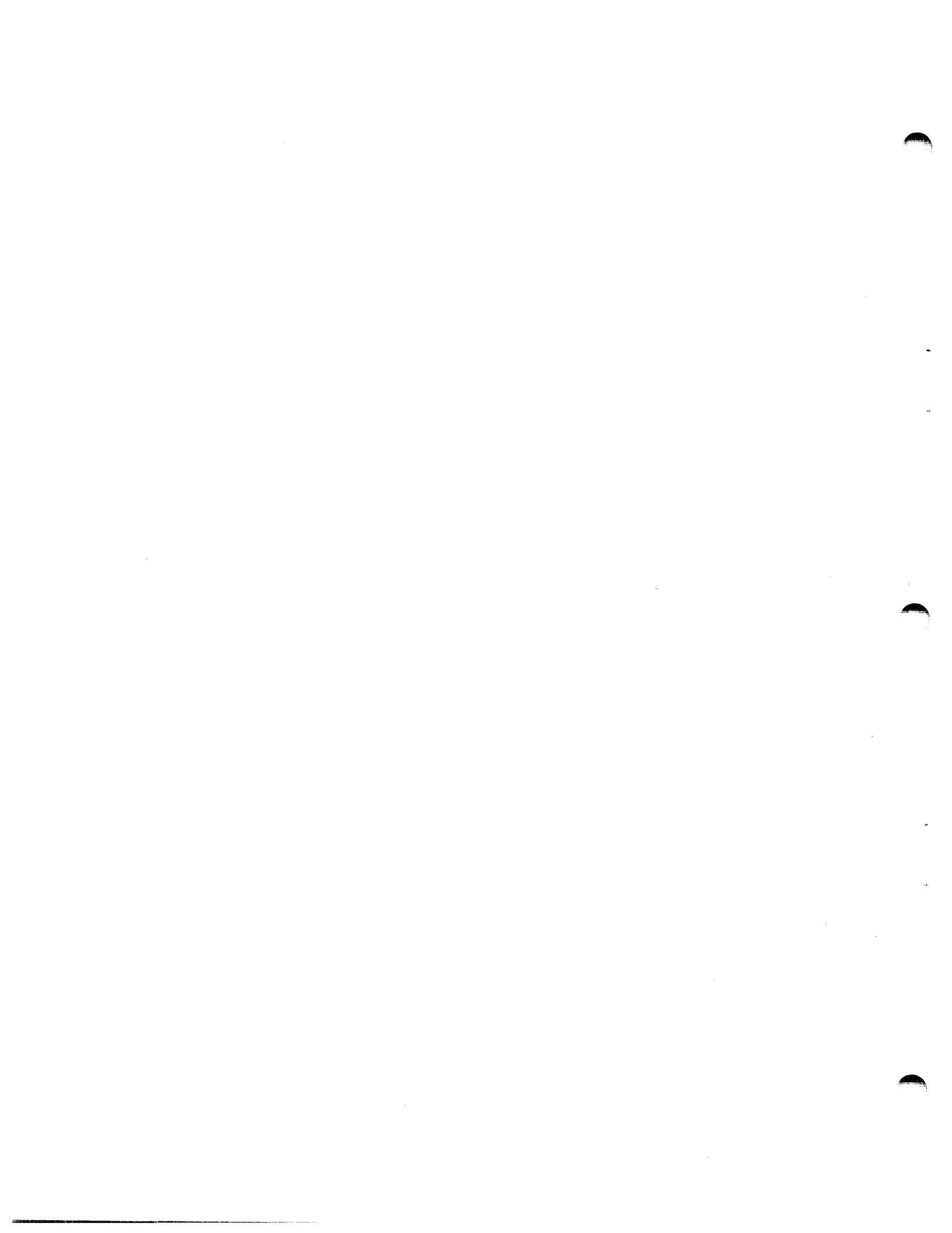
TAPE FORMAT



APPENDIX D

OPTION TABLE

| OPTION | DEFAULT VALUE | DESCRIPTION |
|--------|---------------|--|
| TEST | 1-6 | Selects the test or tests (1-7) to be executed. |
| NOMSG | 0 | Determines whether all messages or only error messages are printed. 0 = all messages 1 = error messages only |
| CONTIN | 0 | Enables the user to run all tests selected continuously until the break key returns the program to the command mode. 0 = normal execution 1 = continuous execution |
| OPTION | N/A | Displays the option labels and values |
| BLSIZE | X'400' | Specifies the size in hexadecimal of the memory block. X'200' for 512 byte blocks X'400' for 1024 byte blocks X'800' for 2048 byte blocks |
| MOD70 | 0 | Specifies the model type strap option 0 = tested with the Model 8/16 E option 1 = tested with the Model 70 option |
| DEVADR | X'AE' | Specifies device address of memory protect controller. |
| MAP | N/A | Displays the memory protect map that was issued last to the controller. 0 = block unprotected. 1 = block protected. The map consists of 8 halfwords with each bit representing a particular block. (See Appendix 7.) Bits 0-7 of each halfword represent execute protection, and bits 8-15 represent write-read protection or write protection. For Model 70 use, bits 0 - are not valid. |



APPENDIX E
ERROR MESSAGES

- TT01 - An interrupt occurred from an unexpected write violation.
(Note 1)
- TT02 - An interrupt occurred from an unexpected read violation,
(Note 1)
- TT03 - An interrupt occurred from an unexpected execute violation.
(Note 1)
- TT04 - The proper number of interrupts did not occur from write vio-
lations. (Note 1)
- TT05 - The proper number of interrupts did not occur from read vio-
lations. (Note 1)
- TT06 - The proper number of interrupts did not occur from execute
violations. (Note 1)
- TT07 - Controller not in an "OFF" status after a protect off command
is issued. (Note 2)
- TT08 - Controller not in an "ON" status (PON bit is set) after a
protect on command is issued. (Note 2)
- TT09 - A status of X'34' or X'64' is expected after a write or read
violation respectively. (Note 2)
- TT10 - A status of X'04' is not received after acknowledging no pending
interrupts. (Note 2)
- TT11 - A status of X'E4' is expected after an execute violation.
(Controller in the Write-Read Mode and the block is both read
and execute protected.) (Note 2)
- TT12 - A status of X'A4' is expected after an execute violation. (Note 2)
- TT13 - An external I/O interrupt is not caused by the Memory Protect
Controller. (Note 2)
- TT14 - An illegal instruction is not generated by the Memory Protect
Controller. (Note 2)
- TT15 - The first address of the block does not contain the proper data,
X'030F'. (Note 3)
- TT16 - The last address of the block does not contain the proper data,
X'030F'. (Note 3)
- TT17 - A device code other than X'00' is received after acknowledging
no pending interrupts. (Note 2)
- TT18 - A write of read protect violation occurred from an unprotected
block. (Note 1)

APPENDIX E (Continued)

- TT19 - An execute protect violation occurred from an unprotected block.
(Note 1)
- TT20 - Excess number of interrupts. (Note 1)
- TT21 - First address of the block does not contain the proper data,
X'FFFF'. (Note 3)
- TT22 - Last address of the block does not contain the proper data,
X'FFFF'. (Note 3)
- TT23 - First address of the block does not contain the proper data,
the address of the location. (Note 3)
- TT24 - Last address of the block does not contain the proper data,
the address of the location. (Note 3)
- TT25 - After an execute violation, the status sensed by acknowledging
the I/O interrupt is not X'E4' or X'A4'. (Note 2)
- TT26 - The "PON" status bit is not set following a write, read or
execute violation. (Note 2)

NOTE 1 - ERROR TTEE
BLOCK XX
WW RR EE

TT = Test number¹
EE = Error number
XX = Block number under test
WW = Number of interrupts from write violations.
RR = Number of interrupts from read violations.
EE = Number of interrupts from execute violations.

NOTE 2 - ERROR TTEE
BLOCK XX
STATUS SS
DEVICE DD

TT = Test number¹
EE = Error number
XX = Block number under test
SS = Status of device
DD = Device code

¹Test number of "XX" signifies that the error did not occur during the execution of a test.

APPENDIX E (Continued)

NOTE 3 - ERROR TTEE
BLOCK XX
A BBBB

TT = Test number¹

EE = Error number

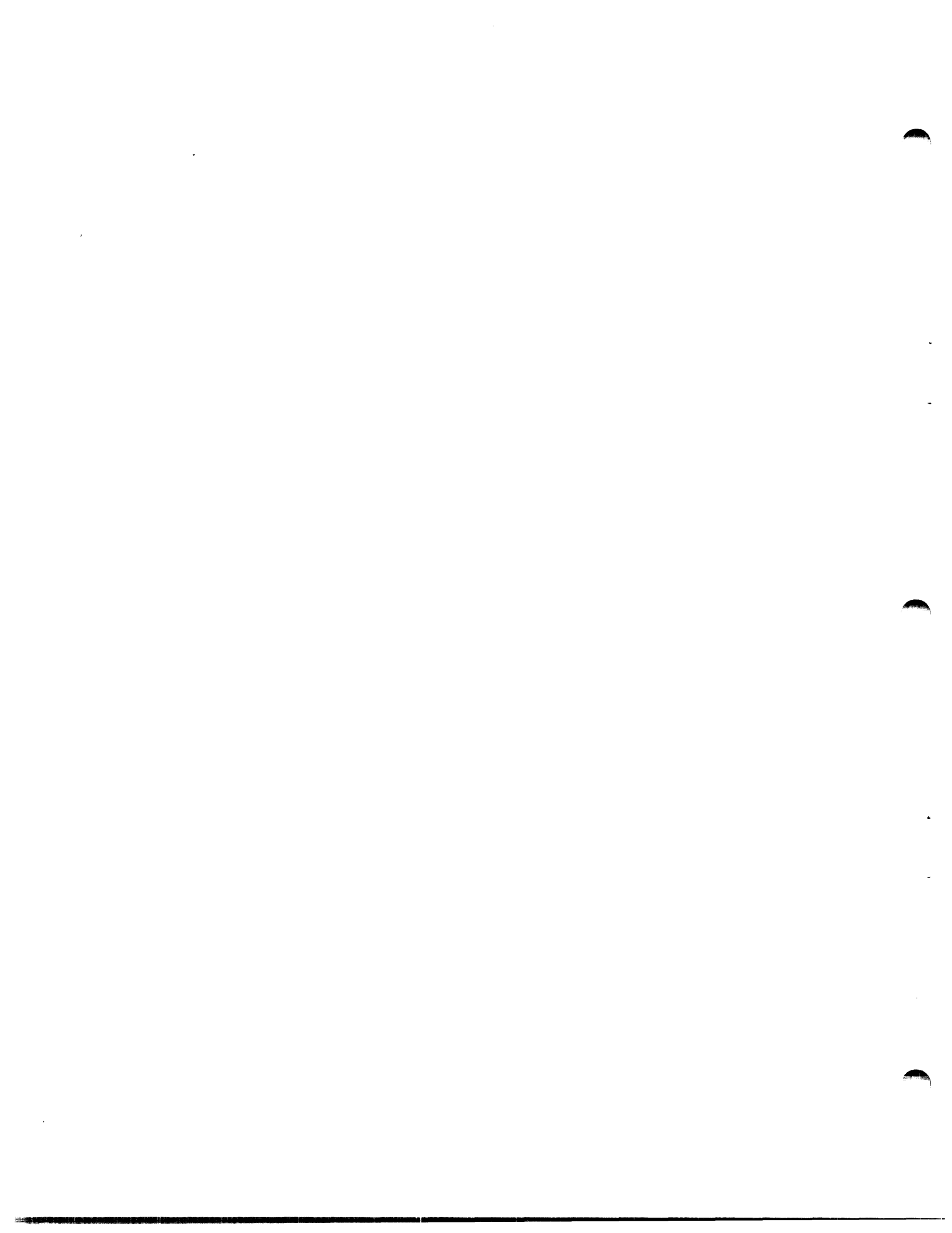
A = Most-significant two bits of two actual address

BBBB = The least-significant 16 bits of the actual address.

(All parameters except TT and EE are expressed in hexadecimal).

Errors created by improper entry into the interrupt service pointer table are also defined.

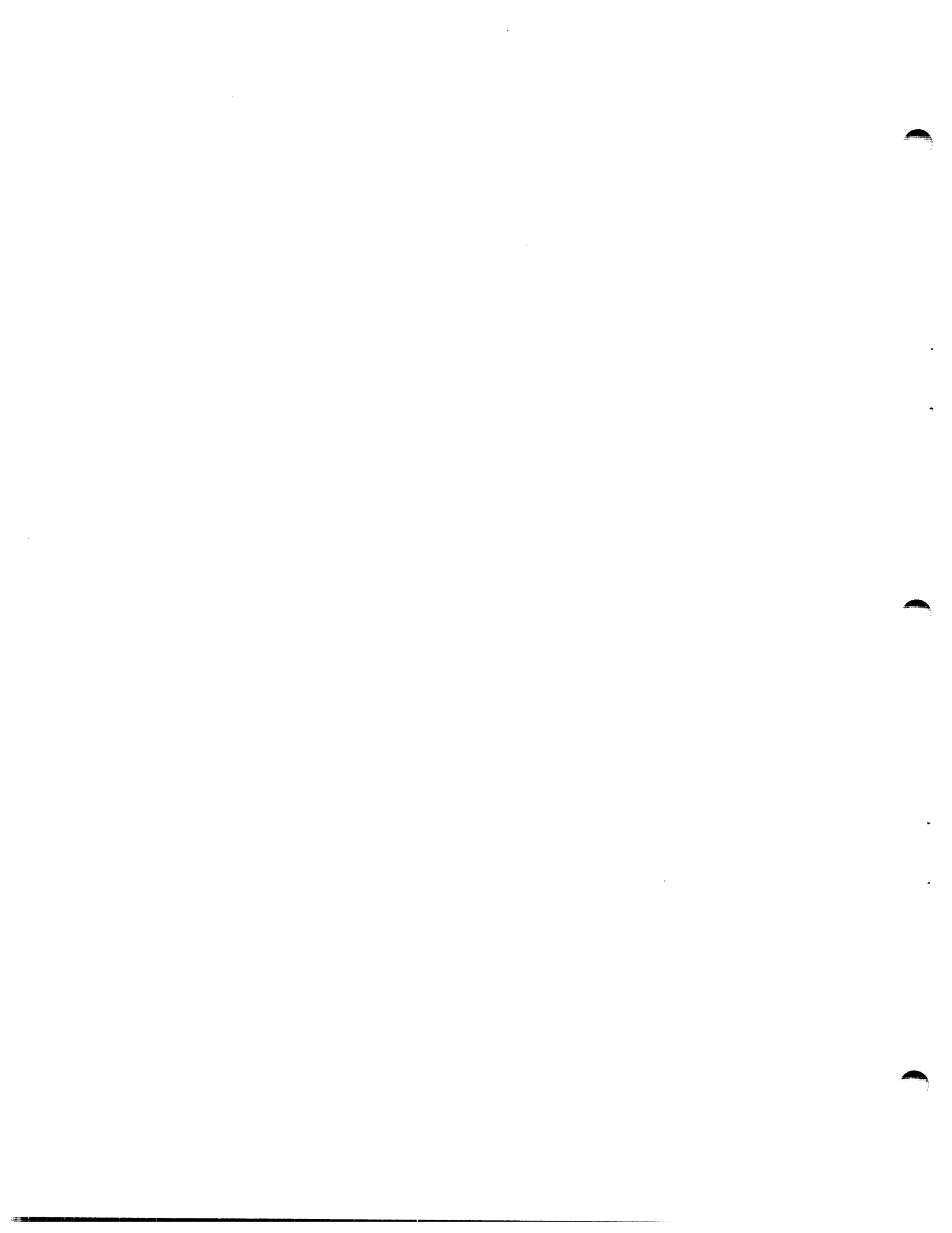
| | |
|---------|---------------------------|
| EE = F0 | SVC Error |
| F1 | Fixed Point Error |
| F2 | System Queue Error |
| F3 | Floating Point Error |
| F4 | Immediate Interrupt Error |



APPENDIX F
TOP OF MEMORY MODIFICATION

The TOP OF MEMORY value displayed at the start of the program is the top of memory calculated by the test program. This value is the last memory location that the program thinks exists in the system.

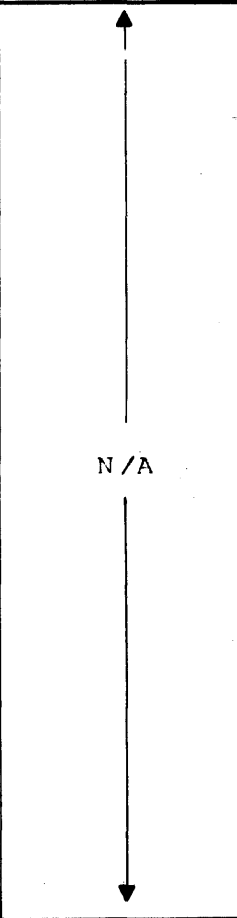
If the test program's TOP OF MEMORY value requires modification, change the contents of the memory labeled ACTTOCMS and ACTTOCLS. The contents of ACTTOCMS should equal the two most significant bits of the 18 bit address required for 256KB of memory. ACTTOCLS should contain the value of the last 16 bits of address. To maintain this modified TOP OF MEMORY value, restart the program at the memory location labeled PRT TOC. All values are to be expressed in hexadecimal.



APPENDIX G
TABLE OF MEMORY ADDRESSES VS. BLOCKS

| BLOCK | LOAD MASK DATA | | MEMORY ADDRESSES (HEX) | | |
|-------|----------------|-----|------------------------|-----------|---------------|
| | BYTE | BIT | .5K BYTE | 1K BYTE | 2K BYTE |
| 1 | 0 | 0 | 0000-01FF | 0000-03FF | 0000-07FF |
| 2 | 0 | 1 | 0200-03FF | 0400-07FF | 0800-0FFF |
| 3 | 0 | 2 | 0400-05FF | 0800-0BFF | 1000-17FF |
| 4 | 0 | 3 | 0600-07FF | 0C00-0FFF | 1800-1FFF |
| 5 | 0 | 4 | 0800-09FF | 1000-13FF | 2000-27FF |
| 6 | 0 | 5 | 0A00-0BFF | 1400-17FF | 2800-2FFF |
| 7 | 0 | 6 | 0C00-0DFF | 1800-1BFF | 3000-37FF |
| 8 | 0 | 7 | 0E00-0FFF | 1C00-1FFF | 3800-3FFF |
| 9 | 1 | 0 | 1000-11FF | 2000-23FF | 4000-47FF |
| 10 | 1 | 1 | 1200-13FF | 2400-27FF | 4800-4FFF |
| 11 | 1 | 2 | 1400-15FF | 2800-2BFF | 5000-57FF |
| 12 | 1 | 3 | 1600-17FF | 2C00-2FFF | 5800-5FFF |
| 13 | 1 | 4 | 1800-19FF | 3000-33FF | 6000-67FF |
| 14 | 1 | 5 | 1A00-1BFF | 3400-37FF | 6800-6FFF |
| 15 | 1 | 6 | 1C00-1DFF | 3800-3BFF | 7000-77FF |
| 16 | 1 | 7 | 1E00-1FFF | 3C00-3FFF | 7800-7FFF |
| 17 | 2 | 0 | 2000-21FF | 4000-43FF | 8000-87FF |
| 18 | 2 | 1 | 2200-23FF | 4400-47FF | 8800-8FFF |
| 19 | 2 | 2 | 2400-25FF | 4800-4BFF | 9000-97FF |
| 20 | 2 | 3 | 2600-27FF | 4C00-4FFF | 9800-9FFF |
| 21 | 2 | 4 | 2800-29FF | 5000-53FF | A000-A7FF |
| 22 | 2 | 5 | 2A00-2BFF | 5400-57FF | A800-AFFF |
| 23 | 2 | 6 | 2C00-2DFF | 5800-5BFF | B000-B7FF |
| 24 | 2 | 7 | 2E00-2FFF | 5C00-5FFF | B800-BFFF |
| 25 | 3 | 0 | 3000-31FF | 6000-63FF | C000-C7FF |
| 26 | 3 | 1 | 3200-33FF | 6400-67FF | C800-CFFF |
| 27 | 3 | 2 | 3400-35FF | 6800-6BFF | D000-D7FF |
| 28 | 3 | 3 | 3600-37FF | 6C00-6FFF | D800-DFFF |
| 29 | 3 | 4 | 3800-39FF | 7000-73FF | E000-E7FF |
| 30 | 3 | 5 | 3A00-3BFF | 7400-77FF | E800-EFFF |
| 31 | 3 | 6 | 3C00-3DFF | 7800-7BFF | F000-F7FF |
| 32 | 3 | 7 | 3E00-3FFF | 7C00-7FFF | F800-FFFF |
| 33 | 4 | 0 | 4000-41FF | 8000-83FF | ↑ N/A ↓ |
| 34 | 4 | 1 | 4200-43FF | 8400-87FF | |
| 35 | 4 | 2 | 4400-45FF | 8800-8BFF | |
| 36 | 4 | 3 | 4600-47FF | 8C00-8FFF | |

APPENDIX G (Continued)

| | | | | | |
|----|---|---|-----------|-----------|---|
| 37 | 4 | 4 | 4800-49FF | 9000-93FF |  N/A |
| 38 | 4 | 5 | 4A00-4BFF | 9400-97FF | |
| 39 | 4 | 6 | 4C00-4DFF | 9800-9BFF | |
| 40 | 4 | 7 | 4E00-4FFF | 9C00-9FFF | |
| 41 | 5 | 0 | 5000-51FF | A000-A3FF | |
| 42 | 5 | 1 | 5200-53FF | A400-A7FF | |
| 43 | 5 | 2 | 5400-55FF | A800-ABFF | |
| 44 | 5 | 3 | 5600-57FF | AC00-AFFF | |
| 45 | 5 | 4 | 5800-59FF | B000-B3FF | |
| 46 | 5 | 5 | 5A00-5BFF | B400-B7FF | |
| 47 | 5 | 6 | 5C00-5DFF | B800-BBFF | |
| 48 | 5 | 7 | 5E00-5FFF | BC00-BFFF | |
| 49 | 6 | 0 | 6000-61FF | C000-C3FF | |
| 50 | 6 | 1 | 6200-63FF | C400-C7FF | |
| 51 | 6 | 2 | 6400-65FF | C800-CBFF | |
| 52 | 6 | 3 | 6600-67FF | CC00-CFFF | |
| 53 | 6 | 4 | 6800-69FF | D000-D3FF | |
| 54 | 6 | 5 | 6A00-6BFF | D400-D7FF | |
| 55 | 6 | 6 | 6C00-6DFF | D800-DBFF | |
| 56 | 6 | 7 | 6E00-6FFF | DC00-DFFF | |
| 57 | 7 | 0 | 7000-71FF | E000-E3FF | |
| 58 | 7 | 1 | 7200-73FF | E400-E7FF | |
| 59 | 7 | 2 | 7400-75FF | E800-EBFF | |
| 60 | 7 | 3 | 7600-77FF | EC00-EFFF | |
| 61 | 7 | 4 | 7800-79FF | F000-F3FF | |
| 62 | 7 | 5 | 7A00-7BFF | F400-F7FF | |
| 63 | 7 | 6 | 7C00-7DFF | F800-FBFF | |
| 64 | 7 | 7 | 7E00-7FFF | FC00-FFFF | |

* If more than 32K bytes of memory exists, and the 512 byte block option is used, the pattern defined for the first 32K repeats itself for the second 32K.

APPENDIX H
8/16E MEMORY PROTECT CONTROLLER (MPC)
PHYSICAL ADDRESSES

| PSW BITS 8:11 | PROGRAM ADDRESS | XMA PHYSICAL ADDRESS |
|---------------|----------------------------|--|
| * 0 or F | 0000 - 7FFF 8000 - FFFF | 00000 - 07FFF (L) 08000 - 0FFFF (U) |
| 1 | 0000 - 7FFF 8000 - FFFF | 00000 - 07FFF (L) 10000 - 17FFF (U) |
| 2 | 0000 - 7FFF 8000 - FFFF | 00000 - 07FFF (L) 18000 - 1FFFF (U) |
| 3 | 0000 - 7FFF 8000 - FFFF | 00000 - 07FFF (L) 20000 - 27FFF (U) |
| 4 | 0000 - 7FFF 8000 - FFFF | 00000 - 07FFF (L) 28000 - 2FFFF (U) |
| 5 | 0000 - 7FFF 8000 - FFFF | 00000 - 07FFF (L) 30000 - 37FFF (U) |
| 6 | 0000 - 7FFF 8000 - FFFF | 00000 - 07FFF (L) 38000 - 3FFFF (U) |
| **7 | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (U) 00000 - 07FFF (L) |
| ***8 | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (L) 08000 - 0FFFF (L) |
| 9 | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (L) 10000 - 17FFF (U) |
| A | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (L) 18000 - 1FFFF (U) |
| B | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (L) 20000 - 27FFF (U) |
| C | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (L) 28000 - 2FFFF (U) |
| D | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (L) 30000 - 37FFF (U) |
| E | 0000 - 7FFF 8000 - FFFF | 08000 - 0FFFF (L) 38000 - 3FFFF (U) |

*PSW 0 or F have the same lower 64KB addressing scheme.

**The upper and lower 32KB are reversed.

***The upper and lower 32KB are one of the same.

(L) = Lower 32KB (U) = Upper 32KB XMA = XMA140 x MA150



APPENDIX I
8/16E MEMORY PROTECT CONTROLLER
CROSS REFERENCE

| BLOCK | | HALFWORD | BIT | | PROGRAM ADDRESS STRAP OPTIONS | | |
|-------|-----|----------|-----|-----|-------------------------------|-----------|-----------|
| W/R | EXC | | W/R | EXC | .5K BYTE | 1K BYTE | 2K BYTE |
| 1 | 65 | 1 | 8 | 0 | 0000-01FF | 0000-03FF | 0000-07FF |
| 2 | 66 | 1 | 9 | 1 | 0200-03FF | 0400-07FF | 0800-0FFF |
| 3 | 67 | 1 | 10 | 2 | 0400-05FF | 0800-0BFF | 1000-17FF |
| 4 | 68 | 1 | 11 | 3 | 0600-07FF | 0C00-0FFF | 1800-1FFF |
| 5 | 69 | 1 | 12 | 4 | 0800-0900 | 1000-13FF | 2000-27FF |
| 6 | 70 | 1 | 13 | 5 | 0A00-0BFF | 1400-17FF | 2800-2FFF |
| 7 | 71 | 1 | 14 | 6 | 0C00-0DFF | 1800-1BFF | 3000-37FF |
| 8 | 72 | 1 | 15 | 7 | 0E00-0FFF | 1C00-1FFF | 3800-3FFF |
| 9 | 73 | 2 | 8 | 0 | 1000-11FF | 2000-23FF | 4000-47FF |
| 10 | 74 | 2 | 9 | 1 | 1200-13FF | 2400-27FF | 4800-4FFF |
| 11 | 75 | 2 | 10 | 2 | 1400-15FF | 2800-28FF | 5000-57FF |
| 12 | 76 | 2 | 11 | 3 | 1600-17FF | 2C00-2FFF | 5800-5FFF |
| 13 | 77 | 2 | 12 | 4 | 1800-19FF | 3000-33FF | 6000-67FF |
| 14 | 78 | 2 | 13 | 5 | 1A00-1BFF | 3400-37FF | 6800-6FFF |
| 15 | 79 | 2 | 14 | 6 | 1C00-1DFF | 3800-3BFF | 7000-77FF |
| 16 | 80 | 2 | 15 | 7 | 1E00-1FFF | 3C00-3FFF | 7800-7FFF |
| 17 | 81 | 3 | 8 | 0 | 2000-21FF | 4000-43FF | 8000-87FF |
| 18 | 82 | 3 | 9 | 1 | 2200-23FF | 4400-47FF | 8800-8FFF |
| 19 | 83 | 3 | 10 | 2 | 2400-25FF | 4800-4BFF | 9000-97FF |
| 20 | 84 | 3 | 11 | 3 | 2600-27FF | 4C00-4FFF | 9800-9FFF |
| 21 | 85 | 3 | 12 | 4 | 2800-29FF | 5000-53FF | A000-A7FF |
| 22 | 86 | 3 | 13 | 5 | 2A00-2BFF | 5400-57FF | A800-AFFF |
| 23 | 87 | 3 | 14 | 6 | 2C00-2DFF | 5800-5BFF | B000-B7FF |
| 24 | 88 | 3 | 15 | 7 | 2E00-2FFF | 5C00-5FFF | B800-BFFF |
| 25 | 89 | 4 | 8 | 0 | 3000-31FF | 6000-63FF | C000-C7FF |
| 26 | 90 | 4 | 9 | 1 | 3200-33FF | 6400-67FF | C800-CFFF |
| 27 | 91 | 4 | 10 | 2 | 3400-35FF | 6800-6BFF | D000-D7FF |
| 28 | 92 | 4 | 11 | 3 | 3600-37FF | 6C00-6FFF | D800-DFFF |
| 29 | 93 | 4 | 12 | 4 | 3800-39FF | 7000-73FF | E000-E7FF |
| 30 | 94 | 4 | 13 | 5 | 3A00-3BFF | 7400-77FF | E800-EFFF |
| 31 | 95 | 4 | 14 | 6 | 3C00-3DFF | 7800-7BFF | F000-F7FF |
| 32 | 96 | 4 | 15 | 7 | 3E00-3FFF | 7C00-7FFF | F800-FFFF |
| 33 | 97 | 5 | 8 | 0 | 4000-41FF | 8000-83FF | |
| 34 | 98 | 5 | 9 | 1 | 4200-43FF | 8400-87FF | |
| 35 | 99 | 5 | 10 | 2 | 4400-45FF | 8800-8BFF | N/A |
| 36 | 100 | 5 | 11 | 3 | 4600-47FF | 8C00-8FFF | |
| 37 | 101 | 5 | 12 | 4 | 4800-49FF | 9000-93FF | |

APPENDIX I (Continued)

| | | | | | | | |
|----|-----|---|----|---|-----------|-----------|---|
| 38 | 102 | 5 | 13 | 5 | 4A00-4BFF | 9400-97FF |  N/A |
| 39 | 103 | 5 | 14 | 6 | 4C00-4DFF | 9800-9BFF | |
| 40 | 104 | 5 | 15 | 7 | 4E00-4FFF | 9C00-9FFF | |
| 41 | 105 | 6 | 8 | 0 | 5000-51FF | A000-A3FF | |
| 42 | 106 | 6 | 9 | 1 | 5200-53FF | A400-A7FF | |
| 43 | 107 | 6 | 10 | 2 | 5400-55FF | A800-ABFF | |
| 44 | 108 | 6 | 11 | 3 | 5600-57FF | AC00-AFFF | |
| 45 | 109 | 6 | 12 | 4 | 5800-59FF | B000-B3FF | |
| 46 | 110 | 6 | 13 | 5 | 5A00-5BFF | B400-B7FF | |
| 47 | 111 | 6 | 14 | 6 | 5C00-5DFF | B800-BBFF | |
| 48 | 112 | 6 | 15 | 7 | 5E00-5FFF | BC00-BFFF | |
| 49 | 113 | 7 | 8 | 0 | 6000-61FF | C000-C3FF | |
| 50 | 114 | 7 | 9 | 1 | 6200-63FF | C400-C7FF | |
| 51 | 115 | 7 | 10 | 2 | 6400-65FF | C800-CBFF | |
| 52 | 116 | 7 | 11 | 3 | 6600-67FF | CC00-CFFF | |
| 53 | 117 | 7 | 12 | 4 | 6800-69FF | D000-D3FF | |
| 54 | 118 | 7 | 13 | 5 | 6A00-6BFF | D400-D7FF | |
| 55 | 119 | 7 | 14 | 6 | 6C00-6DFF | D800-DBFF | |
| 56 | 120 | 7 | 15 | 7 | 6E00-6FFF | DC00-DFFF | |
| 57 | 121 | 8 | 8 | 0 | 7000-71FF | E000-E3FF | |
| 58 | 122 | 8 | 9 | 1 | 7200-73FF | E400-E7FF | |
| 59 | 123 | 8 | 10 | 2 | 7400-75FF | E800-EBFF | |
| 60 | 124 | 8 | 11 | 3 | 7600-77FF | EC00-EFFF | |
| 61 | 125 | 8 | 12 | 4 | 7800-79FF | F000-F3FF | |
| 62 | 126 | 8 | 13 | 5 | 7A00-7BFF | F400-F7FF | |
| 63 | 127 | 8 | 14 | 6 | 7C00-7DFF | F800-FBFF | |
| 64 | 128 | 8 | 15 | 7 | 7E00-7FFF | FC00-FFFF | |

APPENDIX J
STATUS AND COMMAND BYTES

| BIT NO. | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|--------------|---------|------|------|-------|-----|-----|---|---|
| STATUS BYTE | PEF | PRF | *PON | *PWF | | *EX | | |
| COMMAND BYTE | *DISARM | *ARM | *PON | *POFF | W/R | | | |

Strap option
for Model
8/16E

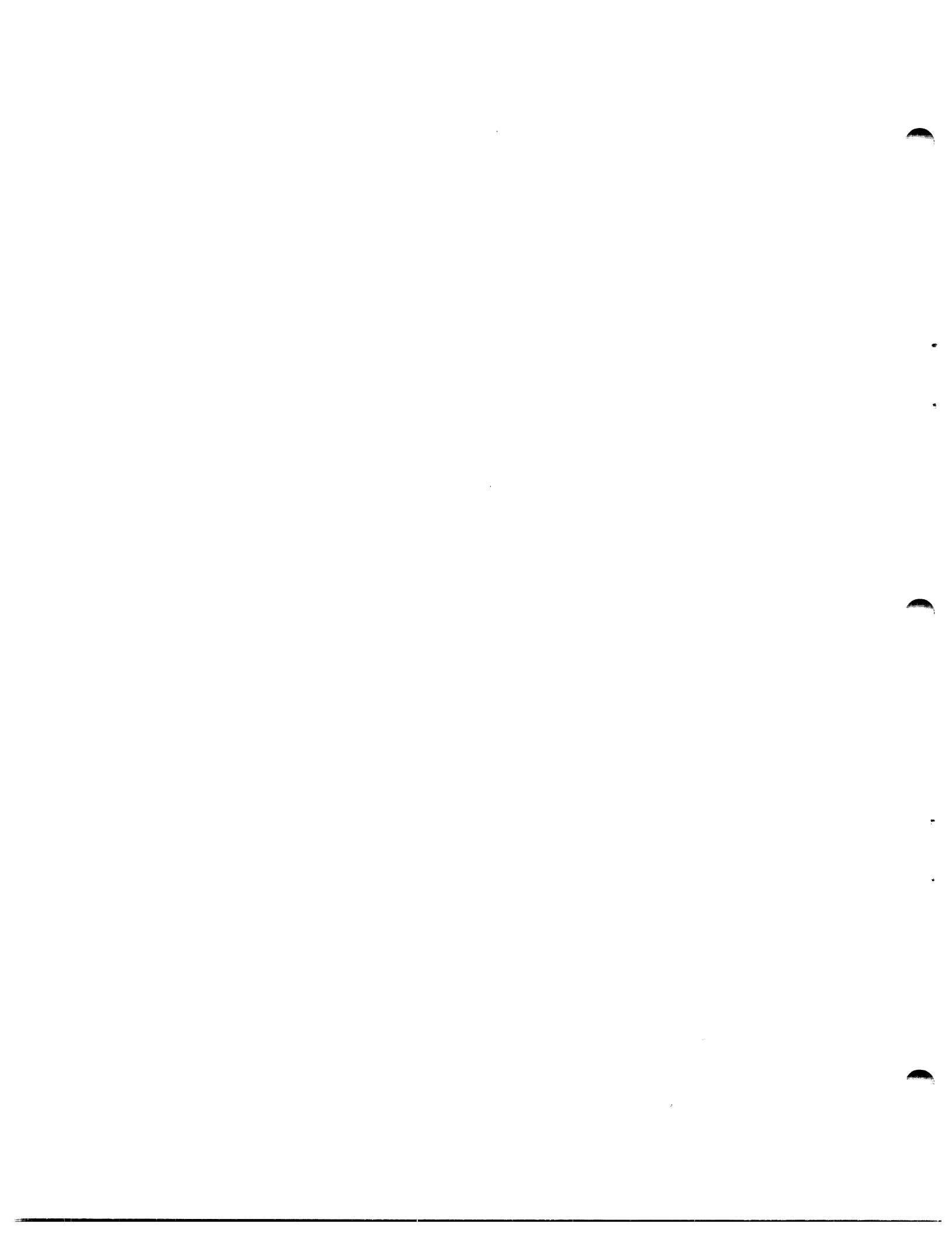
*ONLY THESE BITS ARE USED IN THE MODEL 70 MODE

Status

- PEF Indicates an attempt to execute a protected memory area. An output command (OC or OCR) instruction resets this bit. (Model 8/16E mode only.)
- PRF Indicates an attempt to read from a protected memory area. An output command (OC or OCR) instruction resets this bit. (Model 8/16E mode only.)
- PON Indicates protect is enabled.
- PWF Indicates an attempt to write into a protected memory area. An output command (OR or OCR) instruction resets this bit. For Model 70 mode only, an acknowledge interrupt (AI or AIR) instruction resets this bit.
- EX For Model 8/16E mode, when PWF, PRF, or PEF is set, examine is set. For Model 70 mode, when PWF is set, examine is set.

Command

- DISARM Disables interrupts. They are not queued.
- ARM Enables interrupts.
- PON Indicates protect is enabled.
- POFF Disables the protect function at the controller.
- W/R If this bit is set, the blocks of protected memory are in the write/read protect. (Model 8/16E mode only.)



PROG= MEMPT ASSEMBLED BY CAL 03-066R07-00 (32-BIT)

```

1  **06223
2      CROSS
3      ERLST
4      TARGT 16
5      NORX3
6      WIDTH 120
7  MEMPT  PROG 8/16E MEMORY PROTECT TEST 06-223R01A13
8  *****
9  *
10 *
11 *
12 * THIS PROGRAM TESTS THE 8/16E (SERIES SIXTEEN) MEMORY
13 * PROTECT CONTROLLER.
14 *
15 * SEVEN TESTS ARE PROVIDED IN THIS PROGRAM
16 *
17 * TEST 1      TO INSURE THAT THE MEMORY PROTECT CONTROLLER IS
18 *             INACTIVE WHEN PSW7 IS ZERO OR WHEN AN OFF COMMAND
19 *             IS ISSUED. ALSO, TO INSURE THAT INTERRUPTS CAN BE
20 *             DISARMED.
21 *
22 * TEST 2      TO INSURE THAT MEMORY CAN BE WRITE AND WRITE-READ
23 *             PROTECTED.
24 *
25 * TEST 3      TO INSURE THAT PROGRAM MEMORY CAN BE WRITE AND
26 *             WRITE-READ PROTECTED.
27 *
28 * TEST 4      TO INSURE THAT MEMORY CAN BE EXECUTE PROTECTED.
29 *
30 * TEST 5      TO INSURE THAT MEMORY CAN BE WRITE, READ AND
31 *             EXECUTE PROTECTED.
32 *
33 * TEST 6      TO INSURE THAT PROGRAM MEMORY CAN BE EXECUTE
34 *             PROTECTED.
35 *
36 * TEST 7      TO INSURE THAT MEMORY CAN BE WRITE, READ AND
37 *             EXECUTE PROTECT WITH PSW8 SET.
38 *
39 * TESTS 1,2,4,5 & 7 TEST ALL BLOCKS OF MEMORY EXCEPT THOSE BLOCKS
40 * WHICH CONTAIN THE PROGRAM. TEST 3 & 6 TESTS THE BLOCKS OF MEMORY
41 * THAT DO CONTAIN THE PROGRAM.
42 *
43 * FOR THE MODEL 70 OPTION, ONLY TEST 1,2 & 3 ARE APPLICABLE
44 *
45 * THIS MEMORY PROTECT CONTROLLER CAN BE STRAPPED FOR BLOCCK SIZES
46 * OF 512, 1K AND 2K BYTE SIZE. ALSO IT CAN BE STRAPPED FOR MODEL 70
47 * USE ,FOR 8-16E USE OR SERIES SIXTEEN SERIES.
48 *
49 *
50 * NOTE: A DISTINCTION IS MADE BETWEEN PROGRAM MEMORY WHICH IS THAT
51 * PART OF MEMORY CONTAINING THIS PROGRAM AND SIMPLY MEMORY WHICH IS
52 * THE TOTAL MEMORY MINUS THE PROGRAM MEMORY.
53 * THIS IS DONE FOR PROGRAMMING REASONS. THE PROGRAM OPERATES MORE

```

```

MPC00010
MPC00020
MPC00030
MPC00040
MPC00050
MPC00060
MPC00070
MPC00080
MPC00090
MPC00100
MPC00110
MPC00120
MPC00130
MPC00140
MPC00150
MPC00160
MPC00170
MPC00180
MPC00190
MPC00200
MPC00210
MPC00220
MPC00230
MPC00240
MPC00250
MPC00260
MPC00270
MPC00280
MPC00290
MPC00300
MPC00310
MPC00320
MPC00330
MPC00340
MPC00350
MPC00360
MPC00370
MPC00380
MPC00390
MPC00400
MPC00410
MPC00420
MPC00430
MPC00440
MPC00450
MPC00460
MPC00470
MPC00480
MPC00490
MPC00500
MPC00510
MPC00520
MPC00530

```


54 * EFFICIENTLY WITH THE PROGRAM MEMORY TESTED SEPARATLY FROM THE * MPC00540
 55 * REST OF MEMORY. * MPC00550
 56 * THE KEY ELEMENT USED IN THIS PROGRAM TO MAKE THE DISTINCTION IS * MPC00560
 57 * THE REFERENCE BLOCK (BLOCKREF) WHICH IS THE FIRST BLOCK OF MEMORY * MPC00570
 58 * CONTAINING NONE OF THE PROGRAM. THIS IS THE 6TH, 10TH OR 19TH * MPC00580
 59 * BLOCK OF MEMORY FOR THE 2K, 1K OR 512 BYTE OPTION RESPECTIVELY. * MPC00590
 60 * * MPC00600
 61 * TO EXECUTE THE TESTS: * MPC00610
 62 * ENTER THE DESIRED OPTION INFORMATION VIA THE CONSOLE DEVICE.* MPC00620
 63 * THEN ENTER THE DESIRED TEST NUMBER(S), FOLLOWED BY THE "RUN" * MPC00630
 64 * COMMAND. * MPC00640
 65 * REFER TO APPENDIX 2 OF 06-223M95R01A15 FOR THE OPTION * MPC00650
 66 * COMMAND INPUT STRUCTURE. * MPC00660
 67 * * MPC00670
 68 * * MPC00680
 69 * ASSUMPTIONS: * MPC00690
 70 * THIS PROGRAM ASSUMES THAT THE PROCESSOR TEST AND THE * MPC00700
 71 * MEMORY TEST HAVE RUN WITHOUT DETECTING AN ERROR. * MPC00710
 72 * WITHIN THE PROGRAM THE TESTS SHOULD BE PERFORMED IN * MPC00720
 73 * SEQUENCE; A TEST SHOULD NOT BE PERFORMED UNLESS THE PREVIOUS * MPC00730
 74 * TEST FINDS NO ERRORS. * MPC00740
 75 * * MPC00750
 76 * * MPC00760
 77 * ***** MPC00770

0000 0000
 0000 0001
 0000 0002
 0000 0003
 0000 0004
 0000 0005
 0000 0006
 0000 0007
 0000 0008
 0000 0009
 0000 000A
 0000 000B
 0000 000C
 0000 000D
 0000 000E
 0000 000F
 0000 0003
 0000 0004
 0000 0005
 0000 0006
 0000 0007
 0000 0008
 0000 0009
 0000 000A
 0000 000E

78 R0 EQU 0
 79 R1 EQU 1
 80 R2 EQU 2
 81 R3 EQU 3
 82 R4 EQU 4
 83 R5 EQU 5
 84 R6 EQU 6
 85 R7 EQU 7
 86 R8 EQU 8
 87 R9 EQU 9
 88 R10 EQU 10
 89 R11 EQU 11
 90 R12 EQU 12
 91 R13 EQU 13
 92 R14 EQU 14
 93 R15 EQU 15
 94 RET1 EQU R3
 95 NO EQU R4
 96 BLK EQU R5
 97 INTCR EQU R6
 98 INTCW EQU R7
 99 INTCE EQU R8
 100 INCQR EQU R9
 101 EXPT EQU R10
 102 RET2 EQU R14
 103 *
 104 *
 105 *
 106 *
 107 *
 108 *
 109 *
 110 *
 111 *
 112 *
 113 *
 114 *
 115 *
 116 *
 117 *
 118 *
 119 *
 120 *
 121 *
 122 *
 123 *
 124 *
 125 *
 126 *
 127 *
 128 *
 129 *
 130 *
 131 *
 132 *
 133 *
 134 *
 135 *
 136 *
 137 *
 138 *
 139 *
 140 *
 141 *
 142 *
 143 *
 144 *
 145 *
 146 *
 147 *
 148 *
 149 *
 150 *
 151 *
 152 *
 153 *
 154 *
 155 *
 156 *
 157 *
 158 *
 159 *
 160 *
 161 *
 162 *
 163 *
 164 *
 165 *
 166 *
 167 *
 168 *
 169 *
 170 *
 171 *
 172 *
 173 *
 174 *
 175 *
 176 *
 177 *
 178 *
 179 *
 180 *
 181 *
 182 *
 183 *
 184 *
 185 *
 186 *
 187 *
 188 *
 189 *
 190 *
 191 *
 192 *
 193 *
 194 *
 195 *
 196 *
 197 *
 198 *
 199 *
 200 *

0000R
 0080 C810 0A00
 0084 2421

LOAD START ADRS OF PROGRAM
 LOAD INCREMENT VALUE

MPC00780
 MPC00790
 MPC00800
 MPC00810
 MPC00820
 MPC00830
 MPC00840
 MPC00850
 MPC00860
 MPC00870
 MPC00880
 MPC00890
 MPC00900
 MPC00910
 MPC00920
 MPC00930
 MPC00940
 MPC00950
 MPC00960
 MPC00970
 MPC00980
 MPC00990
 MPC01000
 MPC01010
 MPC01020
 MPC01030
 MPC01040
 MPC01050
 MPC01060
 MPC01070
 MPC01080

| | | | | | | | |
|------|------|------|-----|----------|-------------------|--------------------------------------|----------|
| 0086 | C830 | 2287 | 109 | LHI | R3,PROGEND | LOAD END ADRS OF PROGRAM | MPC01090 |
| 008A | D340 | 0078 | 110 | LB | R4,X'78' | LOAD INPUT DEVICE ADRS | MPC01100 |
| 008E | DE40 | 0079 | 111 | OC | R4,X'79' | ISSUE OUTPUT COMMAND TO INPUT DEVICE | MPC01110 |
| 0092 | 0788 | | 112 | XHR | R8,R8 | ZERO REGISTER 8 | MPC01120 |
| 0094 | 9D45 | | 113 | STATUS1 | SSR R4,R5 | SENSE STATUS OF INPUT DEVICE | MPC01130 |
| 0096 | 20D1 | | 114 | BTBS | X'D',1 | WAIT FOR GOOD STATUS | MPC01140 |
| 0098 | 9B46 | | 115 | RDR | R4,R6 | READ DATA BYTE FROM TAPE | MPC01150 |
| 009A | 0866 | | 116 | LHR | R6,R6 | IS DATA BYTE READ = ZERO ? | MPC01160 |
| 009C | 2234 | | 117 | BZS | STATUS1 | YES, READ NEXT BYTE | MPC01170 |
| 009E | D261 | 0000 | 118 | STR | STB R6,0(R1) | NO, STORE BYTE IN MEMORY | MPC01180 |
| 00A2 | 9471 | | 119 | EXBR | R7,R1 | EXCHANGE BYTES OF CURRENT ADRS | MPC01190 |
| 00A4 | 9827 | | 120 | WHR | R2,R7 | WRITE CURRENT ADRS TO DISPLAY | MPC01200 |
| 00A6 | D371 | 0000 | 121 | LB | R7,0(R1) | LOAD DATA BYTE FROM MEMORY | MPC01210 |
| 00AA | 0787 | | 122 | XHR | R8,R7 | EXCLUSIVE OR DATA BYTE INTO REGISTER | MPC01220 |
| 00AC | 9D45 | | 123 | SSR | R4,R5 | SENSE STATUS OF INPUT DEVICE | MPC01230 |
| 00AE | 20D1 | | 124 | BTBS | X'D',1 | WAIT FOR GOOD STATUS | MPC01240 |
| 00B0 | 9B46 | | 125 | RDR | R4,R6 | READ DATA BYTE FROM TAPE | MPC01250 |
| 00B2 | C110 | 009E | 126 | BXLE | R1,STR | REPEAT UNTIL ENTIRE PROGRAM LOADED | MPC01260 |
| 00B6 | C580 | 00D3 | 127 | MN | CLHI R8,X'D3' | IS CHECKSUM CORRECT ? | MPC01270 |
| 00BA | 2135 | | 128 | BNES | PSWHALT2 | NO, HALT PROCESSOR | MPC01280 |
| 00BC | 0777 | | 129 | XHR | R7,R7 | YES, ZERO REGISTER 7 | MPC01290 |
| 00BE | 9827 | | 130 | WHR | R2,R7 | WRITE ZERO TO DISPLAY | MPC01300 |
| 00C0 | C200 | 00C8 | 131 | LPSW | HALT1 | HALT PROCESSOR AND SET LOC TO X'A00' | MPC01310 |
| 00C4 | C200 | 00CC | 132 | PSWHALT2 | LPSW HALT2 | HALT PROCESSOR AND SET LOC TO X'50' | MPC01320 |
| 00C8 | 8000 | | 133 | HALT1 | DC X'8000',X'A00' | | MPC01330 |
| 00CA | 0A00 | | | | | | |
| 00CC | 8000 | | 134 | HALT2 | DC X'8000',X'50' | | MPC01340 |
| 00CE | 0050 | | | | | | |
| | | | 135 | * | | | MPC01350 |
| | | | 136 | * | | | MPC01360 |
| | | | 137 | * | | | MPC01370 |
| 00D0 | | | 138 | ORG | X'A00' | | MPC01380 |
| 0A00 | 4300 | 0A3A | 139 | BADST | B STARTO | | MPC01390 |
| 0A04 | 4300 | 0A3A | 140 | | B STARTO | | MPC01400 |
| 0A08 | 4300 | 0A3A | 141 | | B STARTO | | MPC01410 |
| 0A0C | 4300 | 0A3A | 142 | | B STARTO | | MPC01420 |
| 0A10 | 0200 | | 143 | IO | DC X'0200' | TTY, NO LIST DEVICE | MPC01430 |
| 0A12 | 0202 | | 144 | TTYADR | DC X'0202' | TTY ADDRESSES | MPC01440 |
| 0A14 | 1011 | | 145 | PASLADR | DC X'1011' | FIRST PASLA ADDR | MPC01450 |
| 0A16 | 1011 | | 146 | PASLADR2 | DC X'1011' | SECOND PASLA ADDR | MPC01460 |
| 0A18 | 6262 | | 147 | LNPADR | DC X'6262' | LINE PRINTER ADDRESS | MPC01470 |
| 0A1A | C0C0 | | 148 | MICADR | DC X'C0C0' | MICRO I/O BUS | MPC01480 |
| 0A1C | 0202 | | 149 | ADDRESS | DC X'0202' | DEVICE ADDRESSES USED IN PROG | MPC01490 |
| 0A1E | F8F8 | | 150 | PASLCMD | DC X'F8F8' | PASLA COMMAND | MPC01500 |
| 0A20 | 2C2C | | 151 | COMMA | DC C',,,' | COMMA | MPC01510 |
| 0A22 | 0000 | | 152 | WRTCMD | DC X'0' | WRITE COMMANDS | MPC01520 |
| 0A24 | 0000 | | 153 | PASLFLG | DC X'0' | PASLA FLAG | MPC01530 |
| 0A26 | 0000 | | 154 | MICFLAG | DC X'0' | MICRO I/O FLAG | MPC01540 |
| 0A28 | 0000 | | 155 | TTYFLG | DC X'0' | TTY FLAG | MPC01550 |
| 0A2A | 80 | | 156 | LNPWRT | DB X'80' | LINE PRINTER WRITE | MPC01560 |
| 0A2B | A3 | | 157 | PASLWRT | DB X'A3' | PASLA WRITE | MPC01570 |
| 0A2C | B1 | | 158 | PASLRD | DB X'B1' | PASLA READ | MPC01580 |
| 0A2D | 82 | | 159 | MICRD | DB X'82' | READ COMMAND | MPC01590 |
| 0A2E | 02 | | 160 | MICWRT | DB X'02' | WRITE COMMAND | MPC01600 |
| 0A2F | 00 | | 161 | SUBTST | DB 0 | | MPC01610 |

| | | | | | | | |
|------|-----------|-----|--------|-----|--------|--------------|----------|
| 0A30 | A4 | 162 | RDCMD | DB | X'A4' | READ COMMAND | MPC01620 |
| 0A31 | 98 | 163 | TTYWRT | DB | X'98' | | MPC01630 |
| 0A32 | A4 | 164 | TTYRD | DB | X'A4' | | MPC01640 |
| 0A33 | 80 | 165 | NORM | DB | X'80' | | MPC01650 |
| 0A34 | 40 | 166 | INCRMT | DB | X'40' | | MPC01660 |
| 0A35 | 60 | 167 | WTONLY | DB | X'60' | | MPC01670 |
| | 0000 0A35 | 168 | PON | EQU | WTONLY | | MPC01680 |
| 0A36 | 68 | 169 | WRTRD | DB | X'68' | | MPC01690 |
| 0A37 | 50 | 170 | POFF | DB | X'50' | | MPC01700 |
| 0A38 | 54 | 171 | POFFWR | DB | X'54' | | MPC01710 |
| 0A39 | A8 | 172 | DISARM | DB | X'A8' | | MPC01720 |

| | | | | | | | | |
|------|------|------|-----|--------|------|------------|----------------------------------|----------|
| 0A3A | C200 | 206E | 174 | STARTO | LPSW | SET1 | | MPC01740 |
| 0A3E | C810 | 21EC | 175 | EXEC | LHI | R1,RSAVE | REG. SAVE POINTER | MPC01750 |
| 0A42 | 4010 | 0022 | 176 | | STH | R1,X'22' | | MPC01760 |
| 0A46 | 2400 | | 177 | | LIS | R0,0 | | MPC01770 |
| 0A48 | 4000 | 002C | 178 | | STH | R0,X'2C' | FLPT NEW PSW | MPC01780 |
| 0A4C | C810 | 1C50 | 179 | | LHI | R1,FLPT | | MPC01790 |
| 0A50 | 4010 | 002E | 180 | | STH | R1,X'2E' | | MPC01800 |
| 0A54 | 4000 | 0034 | 181 | | STH | R0,X'34' | ILL INST NEW PSW | MPC01810 |
| 0A58 | C810 | 1CA2 | 182 | | LHI | R1,ILGINT | | MPC01820 |
| 0A5C | 4010 | 0036 | 183 | | STH | R1,X'36' | | MPC01830 |
| 0A60 | 4000 | 003C | 184 | | STH | R0,X'3C' | MAC MAL NEW PSW | MPC01840 |
| 0A64 | C810 | 1D52 | 185 | | LHI | R1,MALFTN | | MPC01850 |
| 0A68 | 4010 | 003E | 186 | | STH | R1,X'3E' | | MPC01860 |
| 0A6C | 4000 | 0044 | 187 | | STH | R0,X'44' | EXT INT NEW PSW | MPC01870 |
| 0A70 | C810 | 1566 | 188 | | LHI | R1,INTPRT | | MPC01880 |
| 0A74 | 4010 | 0046 | 189 | | STH | R1,X'46' | | MPC01890 |
| 0A78 | 4000 | 004C | 190 | | STH | R0,X'4C' | FIX POINT DIV. NEW PSW | MPC01900 |
| 0A7C | C810 | 1C3C | 191 | | LHI | R1,FXPT | | MPC01910 |
| 0A80 | 4010 | 004E | 192 | | STH | R1,X'4E' | | MPC01920 |
| 0A84 | C810 | 21C8 | 193 | | LHI | R1,TABLE | SYSTEM Q POINTER | MPC01930 |
| 0A88 | 4010 | 0080 | 194 | | STH | R1,X'80' | | MPC01940 |
| 0A8C | C810 | 1C62 | 195 | | LHI | R1,SYSQ | SYSTEM Q | MPC01950 |
| 0A90 | 4010 | 0088 | 196 | | STH | R1,X'88' | AUTO I-O ,SYSTEM QUEUE | MPC01960 |
| 0A94 | 4010 | 0092 | 197 | | STH | R1,X'92' | SYSTEM Q | MPC01970 |
| 0A98 | 4000 | 0086 | 198 | | STH | R0,X'86' | | MPC01980 |
| 0A9C | 4000 | 0090 | 199 | | STH | R0,X'90' | | MPC01990 |
| 0AA0 | 4000 | 009A | 200 | | STH | R0,X'9A' | SUPERVISOR CALL | MPC02000 |
| 0AA4 | C800 | 14C6 | 201 | | LHI | R0,NOPROT | SVC 0 PSW7 0 NO PROTECT | MPC02010 |
| 0AA8 | 4000 | 009C | 202 | | STH | R0,X'9C' | | MPC02020 |
| 0AAC | C800 | 14D6 | 203 | | LHI | R0,DISPROT | SVC 1 DISABLE &PSW 7 0 | MPC02030 |
| 0AB0 | 4000 | 009E | 204 | | STH | R0,X'9E' | | MPC02040 |
| 0AB4 | C800 | 14EA | 205 | | LHI | R0,SET8 | SVC 2 SET PSW 8 | MPC02050 |
| 0AB8 | 4000 | 00A0 | 206 | | STH | R0,X'A0' | | MPC02060 |
| 0ABC | C800 | 14FA | 207 | | LHI | R0,CLEAR8 | SVC 3 CLEAR PSW 8 | MPC02070 |
| 0ACO | 4000 | 00A2 | 208 | | STH | R0,X'A2' | | MPC02080 |
| 0AC4 | C800 | 150A | 209 | | LHI | R0,PROT | SVC 4 SET PSW7 PROTECT | MPC02090 |
| 0AC8 | 4000 | 00A4 | 210 | | STH | R0,X'A4' | | MPC02100 |
| 0ACC | C800 | 151A | 211 | | LHI | R0,ONPROT | SVC 5 PROTECT ON COMMAND | MPC02110 |
| 0ADO | 4000 | 00A6 | 212 | | STH | R0,X'A6' | | MPC02120 |
| 0AD4 | C800 | 1530 | 213 | | LHI | R0,OFFPROT | SVC 6 PROTECT OFF COMMAND | MPC02130 |
| 0AD8 | 4000 | 00A8 | 214 | | STH | R0,X'A8' | | MPC02140 |
| 0ADC | C800 | 1544 | 215 | | LHI | R0,WRPROT | SVC 7 WR-RD COMMAND | MPC02150 |
| 0AE0 | 4000 | 00AA | 216 | | STH | R0,X'AA' | | MPC02160 |
| 0AE4 | C800 | 155A | 217 | | LHI | R0,SENSEZ | SVC 8 SENSE STATUS WITH PSW 7 1 | MPC02170 |
| 0AE8 | 4000 | 00AC | 218 | | STH | R0,X'AC' | | MPC02180 |
| 0AEC | C800 | 1C28 | 219 | | LHI | R0,SVCERR | CLEAR THE REST OF THE SVC RTABLE | MPC02190 |
| 0AFO | C810 | 00AE | 220 | | LHI | R1,X'AE' | | MPC02200 |
| 0AF4 | 2422 | | 221 | | LIS | R2,2 | | MPC02210 |
| 0AF6 | C830 | 00BB | 222 | | LHI | R3,X'BB' | | MPC02220 |
| 0AFA | 4001 | 0000 | 223 | SCC | STH | R0,0(R1) | | MPC02230 |
| 0AFE | C110 | 0AFA | 224 | | BXLE | R1,SCC | AUTO I/O SERVICE TABLE | MPC02240 |
| 0B02 | C800 | 1C88 | 225 | | LHI | R0,EXTINT1 | | MPC02250 |
| 0B06 | C810 | 00D0 | 226 | | LHI | R1,X'D0' | | MPC02260 |
| 0B0A | 2422 | | 227 | | LIS | R2,2 | | MPC02270 |
| 0B0C | C830 | 02CE | 228 | | LHI | R3,X'2CE' | | MPC02280 |

| | | | | | | | |
|------|-----------|-----|----------|------|-------------|--------------------------------|----------|
| OB10 | 4031 0000 | 229 | XCC | STH | R3,0(R1) | | MPC02290 |
| OB14 | C110 OB10 | 230 | | BXLE | R1,XCC | | MPC02300 |
| | | 231 | * | | | | MPC02310 |
| | | 232 | * | | | | MPC02320 |
| OB18 | C8A0 AAAA | 233 | | LHI | R10,X'AAAA' | LOAD PATTERN | MPC02330 |
| OB1C | C830 6666 | 234 | | LHI | R3,X'6666' | | MPC02340 |
| OB20 | 2410 | 235 | | LIS | R1,0 | START ADR = X'0000 20000' | MPC02350 |
| OB22 | C8E0 2000 | 236 | | LHI | R11,X'2000' | | MPC02360 |
| OB26 | 4010 2284 | 237 | FINDTOC | STH | R1,ACTADUP | STORE MOST SIG BITS | MPC02370 |
| OB2A | 41F0 18C8 | 238 | | BAL | R15,ADRTRAN | CONVERT TO PROGRAM ADR | MPC02380 |
| OB2E | 084C | 239 | | LHR | R4,R12 | | MPC02390 |
| OB30 | 2642 | 240 | | AIS | R4,2 | NEXT ADDRESS LOCATION | MPC02400 |
| OB32 | 489C 0000 | 241 | | LH | R9,0(R12) | SAVE CONTENTS OF MEMORY | MPC02410 |
| OB36 | 4854 0000 | 242 | | LH | R5,0(R4) | LOCATIONS | MPC02420 |
| OB3A | C5C0 8000 | 243 | | CLHI | R12,X'8000' | TEST FOR 16 BIT SELCH | MPC02430 |
| OB3E | 4230 0B88 | 244 | | BNE | FINDTOC2 | ADR 8000 NO, CONTINUE | MPC02440 |
| OB42 | 95FF | 245 | | EPSR | R15,R15 | GET CURRENT PSW | MPC02450 |
| OB44 | C4F0 00F0 | 246 | | NHI | R15,X'00F0' | TEST FOR MODULE 1 | MPC02460 |
| OB48 | C5F0 0010 | 247 | | CLHI | R15,X'10' | | MPC02470 |
| OB4C | 4230 0B88 | 248 | | BNE | FINDTOC2 | IF NOT 1 0000 ADR CONTINUE | MPC02480 |
| OB50 | 95FF | 249 | | EPSR | R15,R15 | GET PSW AGAIN | MPC02490 |
| OB52 | 40F0 225C | 250 | | STH | R15,SPSW | SAVE IT | MPC02500 |
| OB56 | 24E0 | 251 | | LIS | R14,0 | | MPC02510 |
| OB58 | 95FE | 252 | | EPSR | R15,R14 | | MPC02520 |
| OB5A | 40E0 8000 | 253 | | STH | R14,X'8000' | | MPC02530 |
| OB5E | 40E0 8002 | 254 | | STH | R14,X'8002' | | MPC02540 |
| OB62 | 95EF | 255 | | EPSR | R14,R15 | | MPC02550 |
| OB64 | 4030 8000 | 256 | | STH | R3,X'8000' | STORE 6666 IN 1 0000 | MPC02560 |
| OB68 | 40A0 8002 | 257 | | STH | R10,X'8002' | STORE AAAA IN 1 0002 | MPC02570 |
| OB6C | 24E0 | 258 | | LIS | R14,0 | MEMORY MODULE 0 | MPC02580 |
| OB6E | 95FE | 259 | | EPSR | R15,R14 | SET PSW | MPC02590 |
| OB70 | 4880 8000 | 260 | | LH | R8,X'8000' | GET CONTENTS OF 0 8000 | MPC02600 |
| OB74 | 4860 8002 | 261 | | LH | R6,X'8002' | GET CONTENTS OF 0 8002 | MPC02610 |
| OB78 | 48E0 225C | 262 | | LH | R14,SPSW | GET CURRENT PSW | MPC02620 |
| OB7C | 95FE | 263 | | EPSR | R15,R14 | RESTORE IT | MPC02630 |
| OB7E | 0583 | 264 | | CLHR | R8,R3 | SAME DATA ? | MPC02640 |
| OB80 | 2134 | 265 | | BNES | FINDTOC2 | NO, CONTINUE | MPC02650 |
| OB82 | 05A6 | 266 | | CLHR | R10,R6 | SAME DATA ? | MPC02660 |
| OB84 | 4330 0BBE | 267 | | BE | FINDTOC1 | YES THAN FOUND TOP | MPC02670 |
| OB88 | 40AC 0000 | 268 | FINDTOC2 | STH | R10,0(R12) | STORE PATTERN | MPC02680 |
| OB8C | 4034 0000 | 269 | | STH | R3,0(R4) | | MPC02690 |
| OB90 | 08FF | 270 | | LHR | R15,R15 | | MPC02700 |
| OB92 | 488C 0000 | 271 | | LH | R8,0(R12) | RETRIEVE | MPC02710 |
| OB96 | 4864 0000 | 272 | | LH | R6,0(R4) | | MPC02720 |
| OB9A | 409C 0000 | 273 | | STH | R9,0(R12) | RESTORE MEMORY | MPC02730 |
| OB9E | 4054 0000 | 274 | | STH | R5,0(R4) | | MPC02740 |
| OBA2 | 058A | 275 | | CLHR | R8,R10 | | MPC02750 |
| OBA4 | 213D | 276 | | BNES | FINDTOC1 | | MPC02760 |
| OBA6 | 0536 | 277 | | CLHR | R3,R6 | | MPC02770 |
| OBA8 | 213B | 278 | | BNES | FINDTOC1 | | MPC02780 |
| OBAA | C8E0 2000 | 279 | | AHI | R11,X'2000' | INC MEM BY X'2000' | MPC02790 |
| OBAA | 4380 0B26 | 280 | | BNC | FINDTOC | IF NO CARRY TRY THIS LOCATION | MPC02800 |
| OBBA | 2611 | 281 | | AIS | R1,1 | INC MS PART OF ACTUAL ADR BY 1 | MPC02810 |
| OBBA | 07EB | 282 | | XHR | R11,R11 | ZERO LS PART OF ACTUAL ADR | MPC02820 |
| OBBA | C510 0005 | 283 | | CLHI | R1,5 | IS MEMORY LESS THAN 256KB | MPC02830 |

| | | | | | | | |
|------|------|-----------|-----|----------|-----------------------|----------------------------------|----------|
| OBBA | 4280 | 0B26 | 284 | BL | FINDTOC | YES, TRY THIS LOCATION | MPC02840 |
| | | | 285 | * | | | MPC02850 |
| OBBE | 48E0 | 2284 | 286 | FINDTOC1 | LH R14,ACTADUP | FOUND TOC | MPC02860 |
| OBC2 | 27B1 | | 287 | SIS | R11,1 | | MPC02870 |
| OBC4 | 4FE0 | 2190 | 288 | SCH | R14,ZERO | SUBTRACT A HW | MPC02880 |
| OBC8 | 95DD | | 289 | EPSR | R13,R13 | ZERO PSW 8-11 BITS | MPC02890 |
| OBCA | C4D0 | FF0F | 290 | NHI | R13,X'FF0F' | | MPC02900 |
| OBCE | 950D | | 291 | EPSR | RO,R13 | | MPC02910 |
| OBDO | 40E0 | 2280 | 292 | STH | R14,ACTTOCMS | STORE PARAMETERS | MPC02920 |
| OBDA | 40B0 | 2282 | 293 | STH | R11,ACTTOCLS | | MPC02930 |
| OBDA | 40E0 | 2284 | 294 | STH | R14,ACTADUP | | MPC02940 |
| OBDC | 41F0 | 18C8 | 295 | BAL | R15,ADRTRAN | TRANSLATE | MPC02950 |
| OBEO | 95FF | | 296 | EPSR | R15,R15 | GET CURRENT PSW | MPC02960 |
| OBE2 | C4F0 | 0070 | 297 | NHI | R15,X'0070' | GET BITS 7-11 | MPC02970 |
| OBE6 | 40F0 | 2276 | 298 | STH | R15,PSW811 | SAVE | MPC02980 |
| OBEA | 081E | | 299 | LHR | R1,R14 | | MPC02990 |
| OBEC | 41E0 | 1DF2 | 300 | BAL | R14,CONVERT | CONVERT MS BITS TO ASCII | MPC03000 |
| OBFO | 0000 | | 301 | DC | X'0' | | MPC03010 |
| OBF2 | 2186 | | 302 | DC | Z(TOCHS) | | MPC03020 |
| OBF4 | 4810 | 2282 | 303 | LH | R1,ACTTOCLS | CONVERT LS BITS TO ASCII | MPC03030 |
| OBFB | 41E0 | 1DF2 | 304 | BAL | R14,CONVERT | | MPC03040 |
| OBFC | 000C | | 305 | DC | X'C' | | MPC03050 |
| OBFE | 218A | | 306 | DC | Z(TOCLS) | | MPC03060 |
| OC00 | 41F0 | 1EA4 | 307 | PRTTITLE | BAL R15,PRINT | 8-16E MEMORY PROTECT TEST 06-223 | MPC03070 |
| OC04 | 2078 | | 308 | DC | Z(TITLE) | START ADRS OF MESSAGE | MPC03080 |
| OC06 | 209F | | 309 | DC | Z(ENDDF) | END ADRS OF MESSAGE | MPC03090 |
| OC08 | 41F0 | 1EA4 | 310 | PRTTOC | BAL R15,PRINT | PRINT TOC PARAMETERS | MPC03100 |
| OC0C | 2174 | | 311 | DC | Z(TOCHESG) | PRINT TOC PARAMETERS | MPC03110 |
| OC0E | 218D | | 312 | DC | Z(TOCHESGE) | | MPC03120 |
| OC10 | 2302 | | 313 | BS | ORG | | MPC03130 |
| OC12 | | | 314 | BADR6 | DS 2 | | MPC03140 |
| | | | 315 | * | | | MPC03150 |
| | | | 316 | * | | | MPC03160 |
| | | | 317 | * | | | MPC03170 |
| | | | 318 | * | O P T I O N T A B L E | | MPC03180 |
| | | | 319 | * | | | MPC03190 |
| | | | 320 | * | | | MPC03200 |
| | | | 321 | * | | | MPC03210 |
| OC14 | 4300 | 0C7C | 322 | ORG | B TTYIN | | MPC03220 |
| OC18 | FC00 | | 323 | TEST | DC X'FC00',C'TEST | | MPC03230 |
| OC1A | 5445 | 5354 2020 | | | | | |
| OC20 | 0000 | | 324 | NOMSG | DC X'0',C'NOMSG' | | MPC03240 |
| OC22 | 4E4F | 4D53 4720 | | | | | |
| OC28 | 0000 | | 325 | CONTIN | DC X'0',C'CONTIN' | | MPC03250 |
| OC2A | 434F | 4E54 494E | | | | | |
| OC30 | 00AE | | 326 | PRTADR | DC X'AE',C'DEVADR' | | MPC03260 |
| OC32 | 4445 | 5641 4452 | | | | | |
| OC38 | 0400 | | 327 | BLOCKSZ | DC X'400',C'BLSIZE' | | MPC03270 |
| OC3A | 424C | 5349 5A45 | | | | | |
| OC40 | 0000 | | 328 | MODEL70 | DC X'0',C'MOD70' | | MPC03280 |
| OC42 | 4D4F | 4437 3020 | | | | | |
| OC48 | 0000 | | 329 | MAP | DC X'0',C'MAP' | | MPC03290 |
| OC4A | 4D41 | 5020 2020 | | | | | |
| | 0000 | 0C50 | 330 | OPTEND | EQU * | | MPC03300 |
| OC50 | 0000 | | 331 | MESS | DC X'0',C'OPTION' | | MPC03310 |

| | | | | | | | |
|------|----------------|-----|----------|------|---------------|------------------------------------|----------|
| OC52 | 4F50 5449 4F4E | | | | | | |
| OC58 | 0000 | 332 | CON | DC | X'0',C'CON | | MPC03320 |
| OC5A | 434F 4E20 2020 | | | | | | |
| OC60 | 0000 | 333 | RUN | DC | X'0',C'RUN | ' ,X'0',X'FFFF' | MPC03330 |
| OC62 | 5255 4E20 2020 | | | | | | |
| OC68 | 00C0 | | | | | | |
| OC6A | FFFF | | | | | | |
| | | 334 | * | | | | MPC03340 |
| | | 335 | * | | | | MPC03350 |
| | 0000 OC6C | 336 | STOP.TST | EQU | * | | MPC03360 |
| OC6C | 8800 | 337 | | DCX | 8800 | BREAK POINT | MPC03370 |
| OC6E | 4300 0A3A | 338 | | B | STARTO | | MPC03380 |
| | | 339 | * | | | | MPC03390 |
| | | 340 | * | | | | MPC03400 |
| | | 341 | * | | | | MPC03410 |
| | | 342 | * | | | | MPC03420 |
| OC72 | C8E0 1EA4 | 343 | QUESTNZ | LHI | R14,PRINT | | MPC03430 |
| OC76 | 01FE | 344 | QUESTN | BALR | R15,R14 | OUTPUT A CR, LF, ?, CR, LF | MPC03440 |
| OC78 | 20FE | 345 | | DC | Z(QMARK) | | MPC03450 |
| OC7A | 2103 | 346 | | DC | Z(QEND) | | MPC03460 |
| | | 347 | * | | | | MPC03470 |
| | | 348 | * | | | | MPC03480 |
| | | 349 | * | | | | MPC03490 |
| OC7C | C8E0 1EA4 | 350 | TTYIN | LHI | R14,PRINT | SETUP R14 FOR PRINT ROUTINE | MPC03500 |
| OC80 | C890 OC76 | 351 | | LHI | R9,QUESTN | SETUP R9 FOR ERROR ROUTINE | MPC03510 |
| OC84 | 01FE | 352 | LF | BALR | R15,R14 | OUTPUT AN * TO INDICATE | MPC03520 |
| OC86 | 2104 | 353 | | DC | Z(ASTERISK) | WE ARE READY FOR INPUT | MPC03530 |
| OC88 | 2109 | 354 | | DC | Z(ENDAST) | | MPC03540 |
| OC8A | C880 2000 | 355 | | LHI | R8,X'2000' | DISABLE INTERRUPTS ENSURED | MPC03550 |
| OC8E | 9518 | 356 | | EPSR | R1,R8 | | MPC03560 |
| OC90 | C800 2020 | 357 | | LHI | R0,X'2020' | BLANK OUT TTY BUFFER | MPC03570 |
| OC94 | 4000 227A | 358 | | STH | R0,TTYBUF | | MPC03580 |
| OC98 | 4000 227C | 359 | | STH | R0,TTYBUF+2 | | MPC03590 |
| OC9C | 4000 227E | 360 | | STH | R0,TTYBUF+4 | | MPC03600 |
| OCA0 | DEB0 0A30 | 361 | | OC | R11,RDCMD | SET READ MODE | MPC03610 |
| OCA4 | 0711 | 362 | | XHR | R1,R1 | CLEAR TTY INDEX | MPC03620 |
| OCA6 | 41F0 1E8E | 363 | RDCHR | BAL | R15,GETCHR | GET A CHARACTER | MPC03630 |
| OCAA | C500 000D | 364 | | CLHI | R0,X'0D' | IS IT A CR ? | MPC03640 |
| OCAE | 233A | 365 | | BES | OKIN | YES TRY TO MATCH IT TO TABLE | MPC03650 |
| OCB0 | C500 0020 | 366 | | CLHI | R0,X'20' | IS IT A BLANK ? | MPC03660 |
| OCB4 | 2337 | 367 | | BES | OKIN | YES, TRY A MATCH | MPC03670 |
| OCB6 | D201 227A | 368 | | STB | R0,TTYBUF(R1) | NO, STORE THE CHARACTER | MPC03680 |
| OCBA | 2611 | 369 | | AIS | R1,1 | BUMP BUFFER INDEX | MPC03690 |
| OCBC | C510 0006 | 370 | | CLHI | R1,6 | HAVE WE REACHED 6 CHARACTERS ? | MPC03700 |
| OC00 | 203D | 371 | | BNES | RDCHR | NO, DO ANOTHER READ | MPC03710 |
| OCC2 | 0711 | 372 | OKIN | XHR | R1,R1 | MATCH ROUTINE - CLEAR TABLE INDEX | MPC03720 |
| OCC4 | 0733 | 373 | OKIN2 | XHR | R3,R3 | CLEAR TTYBUF INDEX | MPC03730 |
| OCC6 | 0841 | 374 | | LHR | R4,R1 | SET TABLE INDEX (NEW) | MPC03740 |
| OCC8 | 4854 OC1A | 375 | LOOKUP | LH | R5,ORG+6(R4) | GET HALFWORD FROM MEMORY | MPC03750 |
| OCCC | 0219 | 376 | | BMR | R9 | IF MINUS, THEN NO MATCH I.E. ERROR | MPC03760 |
| OCCE | 4553 227A | 377 | | CLH | R5,TTYBUF(R3) | COMPARE TO TTYBUF HALFWORD | MPC03770 |
| OC02 | 4230 0DDE | 378 | | BNE | NEXT | NO MATCH, BUMP TO NEXT TABLE ENTRY | MPC03780 |
| OC06 | 2642 | 379 | | AIS | R4,2 | IF EQUAL, TRY NEXT HALFWORD | MPC03790 |
| OC08 | 2632 | 380 | | AIS | R3,2 | | MPC03800 |
| OC0A | C530 0006 | 381 | | CLHI | R3,6 | HAVE WE FOUND 3 EQUAL HALFWORDS | MPC03810 |

| | | | | | | | |
|------|-----------|-----|---------|------|------------------|---------------------------------|----------|
| OCDE | 203B | 382 | | BNES | LOOKUP | NO, LOOP | MPC03820 |
| OCE0 | C510 0048 | 383 | MATCH | CLHI | R1,RUN-ORG-4 | OPTION MATCH - CHECK IF RUN CMD | MPC03830 |
| OCE4 | 4330 0E26 | 384 | | BE | SELTS1 | YES, SELECT TEST | MPC03840 |
| OCE8 | C510 0040 | 385 | | CLHI | R1,CON-ORG-4 | CON OPTION | MPC03850 |
| OCEC | 4330 0C6C | 386 | | BE | STOP.TST | YES, BRANCH | MPC03860 |
| OCF0 | C510 0038 | 387 | MESSCHK | CLHI | R1,MESS-ORG-4 | IS IT THE MESSAGE OPTION? | MPC03870 |
| OCF4 | 2135 | 388 | | BNES | MAPCHK | NO CHECK NEXT OPTION | MPC03880 |
| OCF6 | 41F0 1940 | 389 | | BAL | R15,MESSOUT | OTHERWISE OUTPUT MESSAGE | MPC03890 |
| OCFA | 4300 0C84 | 390 | | B | LF | RETURN FOR ANOTHER ENTRY | MPC03900 |
| OCFE | C510 0030 | 391 | MAPCHK | CLHI | R1,MAP-ORG-4 | MAP OPTION? | MPC03910 |
| OD02 | 2136 | 392 | | BNES | BLKCHK | NO, CHECK BLOCK OPTION | MPC03920 |
| OD04 | 41E0 19D0 | 393 | | BAL | R14,MAPOUT | DISPLAY PROTECTION MAP | MPC03930 |
| OD08 | 4300 0C7C | 394 | | B | TTYIN | WAIT FOR ANOTHER OPTION | MPC03940 |
| OD0C | | 395 | BADR7 | DS | 2 | | MPC03950 |
| OD0E | C500 000D | 396 | BLKCHK | CLHI | R0,X'0D' | CR ? | MPC03960 |
| OD12 | 0339 | 397 | | BER | R9 | DISPLAY ? | MPC03970 |
| OD14 | C510 0020 | 398 | | CLHI | R1,BLOCKSZ-ORG-4 | BLOCK OPTION? | MPC03980 |
| OD18 | 4230 0D9C | 399 | | BNE | LOKAGN | NO, CHECK FOR TEST COMMAND | MPC03990 |
| OD1C | 41D0 0DE4 | 400 | | BAL | R13,HEXASC | | MPC04000 |
| OD20 | C960 0200 | 401 | | CHI | R6,X'200' | BLOCK SIZE X'200' ? | MPC04010 |
| OD24 | 4230 0D4E | 402 | | BNE | BLK1 | NO, CHECK FOR 400 OR 800 SIZE | MPC04020 |
| OD28 | 4061 0C18 | 403 | | STH | R6,ORG+4(R1) | YES, STORE BLOCK PARAMETERS | MPC04030 |
| OD2C | C860 0040 | 404 | | LHI | R6,64 | | MPC04040 |
| OD30 | 4060 2198 | 405 | | STH | R6,BLOCKMP | | MPC04050 |
| OD34 | C860 0080 | 406 | | LHI | R6,128 | | MPC04060 |
| OD38 | 4060 219A | 407 | | STH | R6,BLOCKMZ | | MPC04070 |
| OD3C | C860 0013 | 408 | | LHI | R6,19 | | MPC04080 |
| OD40 | 4060 219C | 409 | | STH | R6,BLOCKREF | | MPC04090 |
| OD44 | 246D | 410 | | LIS | R6,13 | | MPC04100 |
| OD46 | 4060 219E | 411 | | STH | R6,BLOCKFLG | | MPC04110 |
| OD4A | 4300 0C84 | 412 | | B | LF | | MPC04120 |
| OD4E | C960 0400 | 413 | BLK1 | CHI | R6,X'400' | BLOCK SIZE X'400'? | MPC04130 |
| OD52 | 4230 0D76 | 414 | | BNE | BLK2 | NO, CHECK FOR 800 SIZE | MPC04140 |
| OD56 | 4061 0C18 | 415 | | STH | R6,ORG+4(R1) | YES, STORE BLOCK PARAMETERS | MPC04150 |
| OD5A | C860 0040 | 416 | | LHI | R6,64 | | MPC04160 |
| OD5E | 4060 219A | 417 | | STH | R6,BLOCKMZ | | MPC04170 |
| OD62 | 4060 2198 | 418 | | STH | R6,BLOCKMP | | MPC04180 |
| OD66 | 246A | 419 | | LIS | R6,10 | | MPC04190 |
| OD68 | 4060 219C | 420 | | STH | R6,BLOCKREF | | MPC04200 |
| OD6C | 2467 | 421 | | LIS | R6,7 | | MPC04210 |
| OD6E | 4060 219E | 422 | | STH | R6,BLOCKFLG | | MPC04220 |
| OD72 | 4300 0C84 | 423 | | B | LF | | MPC04230 |
| OD76 | C960 0800 | 424 | BLK2 | CHI | R6,X'800' | BLOCK SIZE X'800'? | MPC04240 |
| OD7A | 0239 | 425 | | BNER | R9 | NO, DISPLAY ? | MPC04250 |
| OD7C | 4061 0C18 | 426 | | STH | R6,ORG+4(R1) | YES, STORE BLOCK PARAMETERS | MPC04260 |
| OD80 | C860 0020 | 427 | | LHI | R6,32 | | MPC04270 |
| OD84 | 4060 219A | 428 | | STH | R6,BLOCKMZ | | MPC04280 |
| OD88 | 4060 2198 | 429 | | STH | R6,BLOCKMP | | MPC04290 |
| OD8C | 2466 | 430 | | LIS | R6,6 | | MPC04300 |
| OD8E | 4060 219C | 431 | | STH | R6,BLOCKREF | | MPC04310 |
| OD92 | 2464 | 432 | | LIS | R6,4 | | MPC04320 |
| OD94 | 4060 219E | 433 | | STH | R6,BLOCKFLG | | MPC04330 |
| OD98 | 4300 0C84 | 434 | | B | LF | | MPC04340 |
| OD9C | C510 0000 | 435 | LOKAGN | CLHI | R1,TEST-ORG-4 | CHECK IF TEST CMD | MPC04350 |
| ODA0 | 2337 | 436 | | BES | TESTST | | MPC04360 |

| | | | | | | | |
|------|------|------|-----|--------|-------------------|-------------------------------------|----------|
| ODA2 | 41D0 | ODE4 | 437 | BAL | R13,HEXASC | GET HEX OPERAND | MPC04370 |
| ODA6 | 40E1 | OC18 | 438 | STH | R6,ORG+4(R1) | STORE IN OPTION TABLE HALFWORD | MPC04380 |
| ODAA | 4300 | OC84 | 439 | LF1 | B LF | GO TO BEGINNING | MPC04390 |
| ODAE | 24C0 | | 440 | TESTST | LIS R12,0 | | MPC04400 |
| ODBO | 41D0 | ODE4 | 441 | TST00 | BAL R13,HEXASC | GET HEX OPERAND | MPC04410 |
| ODB4 | C560 | 0008 | 442 | | CLHI R6,8 | 8 OR GREATER | MPC04420 |
| ODB8 | 0389 | | 443 | | BNLR R9 | YES, ERROR | MPC04430 |
| ODBA | 0866 | | 444 | | LHR R6,R6 | ZERO ...?? | MPC04440 |
| ODBC | 0339 | | 445 | | BZR R9 | YES, ERROR | MPC04450 |
| ODBE | 2431 | | 446 | | LIS R3,1 | CONVERT FROM BINARY TO | MPC04460 |
| ODCO | C560 | 0010 | 447 | TST01 | CLHI R6,16 | UNARY BIT PATTERN LEFT | MPC04470 |
| ODC4 | 2334 | | 448 | | BES TST2 | | MPC04480 |
| ODC6 | 0A33 | | 449 | | AHR R3,R3 | | MPC04490 |
| ODC8 | 2661 | | 450 | | AIS R6,1 | | MPC04500 |
| ODCA | 2205 | | 451 | | BS TST01 | | MPC04510 |
| ODCC | 06C3 | | 452 | TST2 | OHR R12,R3 | OR BIT PATTERN INTO OPTION REGISTER | MPC04520 |
| ODCE | C500 | 000D | 453 | | CLHI R0,X'0D' | WHERE WE TERMINATED BY CR ? | MPC04530 |
| ODD2 | 4230 | ODB0 | 454 | | BNE TST00 | NO, LOOK FOR ANOTHER HEX OPERAND | MPC04540 |
| ODD6 | 40C1 | OC18 | 455 | | STH R12,ORG+4(R1) | STORE OPTION | MPC04550 |
| ODDA | 4330 | OC84 | 456 | | BE LF | YES, GO TO BEGINNING | MPC04560 |
| ODDE | 2618 | | 457 | NEXT | AIS R1,8 | BUMP TABLE INDEX TO NEXT ENTRY | MPC04570 |
| ODE0 | 4300 | OCC4 | 458 | | B OKIN2 | RESUME LOOKUP | MPC04580 |
| ODE4 | 41F0 | 1E8E | 459 | HEXASC | BAL R15,GETCHR | HEX CONVERT ROUTINE | MPC04590 |
| ODE8 | 0766 | | 460 | | XHR R6,R6 | CLEAR BUFFER REGISTER | MPC04600 |
| ODEA | C500 | 0020 | 461 | | CLHI R0,X'20' | SKIP LEADING SPACES | MPC04610 |
| ODEE | 2235 | | 462 | | BES HEXASC | GET VALUE | MPC04620 |
| ODF0 | C500 | 0030 | 463 | HEXLP | CLHI R0,C'0' | CHECK IF VALID HEX CHAR | MPC04630 |
| ODF4 | 0289 | | 464 | | BLR R9 | NO, PRINT? | MPC04640 |
| ODF6 | C500 | 003A | 465 | | CLHI R0,X'3A' | IS IT A HEX NUMBER ? | MPC04650 |
| ODFA | 2188 | | 466 | | BLS HEX | YES, ADD CHAR TO BUFFER | MPC04660 |
| ODFC | C500 | 0041 | 467 | | CLHI R0,C'A' | NO, IS IT A HEX LETTER ? | MPC04670 |
| OE00 | 0289 | | 468 | | BLR R9 | | MPC04680 |
| OE02 | C500 | 0047 | 469 | | CLHI R0,X'47' | | MPC04690 |
| OE06 | 0389 | | 470 | | BNLR R9 | NO, PRINT A "?" | MPC04700 |
| OE08 | 2609 | | 471 | | AIS R0,9 | YES, ADJUST A-F TO 10-15 | MPC04710 |
| OE0A | C400 | 000F | 472 | HEX | NHI R0,15 | ISOLATE 4 BITS | MPC04720 |
| OE0E | 9164 | | 473 | | SLLS R6,4 | SHIFT LEFT 4 | MPC04730 |
| OE10 | 0660 | | 474 | | OHR R6,R0 | OR IN NEW CHARACTER | MPC04740 |
| OE12 | 41F0 | 1E8E | 475 | | BAL R15,GETCHR | GET NEXT CHARACTER | MPC04750 |
| OE16 | C500 | 000D | 476 | | CLHI R0,X'0D' | | MPC04760 |
| OE1A | 033D | | 477 | | BER R13 | EXIT IF CR | MPC04770 |
| OE1C | C500 | 002C | 478 | | CLHI R0,X'2C' | | MPC04780 |
| OE20 | 033D | | 479 | | BER R13 | OR COMMA | MPC04790 |
| OE22 | 4300 | ODF0 | 480 | | B HEXLP | LOOP TO PROCESS IT | MPC04800 |
| | | | 481 | * | | | MPC04810 |
| | | | 482 | * | | | MPC04820 |
| | | | 483 | * | | | MPC04830 |
| OE26 | 0788 | | 484 | SELTS1 | XHR R8,R8 | | MPC04840 |
| OE28 | 4080 | 0A28 | 485 | | STH R8,TTYFLG | ZERO TTY OFF FLAG | MPC04850 |
| OE2C | 4080 | 21D4 | 486 | | STH R8,TOTAL | ZERO PASS COUNT | MPC04860 |
| OE30 | 4080 | 21D6 | 487 | | STH R8,TOTALERR | ZERO ERRORS | MPC04870 |
| OE34 | 4080 | 20AA | 488 | | STH R8,ERRNUM | ZERO ERROR FLAG | MPC04880 |
| OE38 | D280 | 2284 | 489 | | STB R8,ACTADUP | ZERO ACTUAL UPPER ADDRESS | MPC04890 |
| | | | 490 | * | | | MPC04900 |
| | | | 491 | * | | | MPC04910 |

| | | | | | | | | |
|------|------|------|-----|--------|------|-------------|------------------------------------|----------|
| 0E3C | 4880 | 0C18 | 492 | SELTST | LH | R8,TEST | GET TEST OPTION | MPC04920 |
| 0E40 | 0711 | | 493 | | XHR | R1,R1 | ZERO TEST NUMBER | MPC04930 |
| 0E42 | 230D | | 494 | | BS | SHIFT | DECODE TEST OPTION | MPC04940 |
| 0E44 | 0711 | | 495 | TSTSEL | XHR | R1,R1 | ZERO TEST NUMBER | MPC04950 |
| 0E46 | 4010 | 20AA | 496 | | STH | R1,ERRNUM | ZERO ERROR FLAG | MPC04960 |
| 0E4A | 4880 | 21D8 | 497 | TSTSL2 | LH | R8,OPTSAV | LOAD CURRENT TEST OPTION | MPC04970 |
| 0E4E | D310 | 0A2F | 498 | | LB | R1,SUBTST | LOAD PREVIOUS TEST NUMBER | MPC04980 |
| 0E52 | 2611 | | 499 | BUMP | AIS | R1,1 | INCREMENT TEST NUMBER | MPC04990 |
| 0E54 | C510 | 0008 | 500 | | CLHI | R1,8 | HAVE WE REACHED MAX TEST ? | MPC05000 |
| 0E58 | 4380 | 0EAE | 501 | | BNL | OPTCHK | YES, CHECK FOR CONTIN OPTION | MPC05010 |
| 0E5C | 9181 | | 502 | SHIFT | SLLS | R8,1 | NO, IS NEXT TEST TO BE EXECUTED ? | MPC05020 |
| 0E5E | 2286 | | 503 | | BNCS | BUMP | NO, INCREMENT TEST NUMBER | MPC05030 |
| 0E60 | 4080 | 21D8 | 504 | | STH | R8,OPTSAV | YES, SAVE CURRENT TEST NUMBER | MPC05040 |
| 0E64 | D210 | 0A2F | 505 | | STB | R1,SUBTST | SAVE CURRENT TEST NUMBER | MPC05050 |
| 0E68 | 9111 | | 506 | | SLLS | R1,1 | ESTABLISH BRANCH INDEX | MPC05060 |
| 0E6A | C880 | 2000 | 507 | | LHI | R8,X'2000' | DISABLE INTERRUPTS | MPC05070 |
| 0E6E | 95A8 | | 508 | | EPSR | R10,R8 | | MPC05080 |
| 0E70 | D3B0 | 0A1C | 509 | | LB | R11,ADDRESS | GET CONSOLE ADR | MPC05090 |
| 0E74 | DEB0 | 0A30 | 510 | | OC | R11,RDCMD | READ TTY | MPC05100 |
| 0E78 | E100 | 0000 | 511 | | SVC | 0,0 | ZERO PSW 7 | MPC05110 |
| 0E7C | 41F0 | 1E2C | 512 | | BAL | R15,TSTBRK | CHECK FOR BREAK | MPC05120 |
| 0E80 | 9FEF | | 513 | | AIR | R14,R15 | ACKNOWLEDGE ANY PENDING INTERRUPTS | MPC05130 |
| 0E82 | C8F0 | 5858 | 514 | | LHI | R15,X'5858' | TEST NUMBER = XX | MPC05140 |
| 0E86 | 40F0 | 20A8 | 515 | | STH | R15,TESTNUM | STORE IT | MPC05150 |
| 0E8A | 2450 | | 516 | | LIS | BLK,0 | ZERO BLOCK REGISTER | MPC05160 |
| 0E8C | 4050 | 2268 | 517 | | STH | BLK,BLKPT | CLEAR WRITE-READ PROTECT STATUS | MPC05170 |
| 0E90 | 4050 | 226A | 518 | | STH | BLK,BLKPT | CLEAR EXECUTE PROTECT STATUS | MPC05180 |
| 0E94 | C8F0 | 6000 | 519 | | LHI | R15,X'6000' | ENABLE EXT & MAP INTER. | MPC05190 |
| 0E98 | 95EF | | 520 | | EPSR | R14,R15 | | MPC05200 |
| 0E9A | 4881 | 0EAO | 521 | | LH | R8,TST(R1) | GET TEST NUMBER | MPC05210 |
| 0E9E | 0308 | | 522 | | BR | R8 | GO TO TEST | MPC05220 |
| 0EA0 | 0F38 | | 523 | TST | DC | Z(TEST1) | TEST 1 | MPC05230 |
| 0EA2 | 0FFC | | 524 | | DC | Z(TEST2) | TEST 2 | MPC05240 |
| 0EA4 | 1104 | | 525 | | DC | Z(TEST3) | TEST 3 | MPC05250 |
| 0EA6 | 11E2 | | 526 | | DC | Z(TEST4) | TEST 4 | MPC05260 |
| 0EA8 | 1242 | | 527 | | DC | Z(TEST5) | TEST 5 | MPC05270 |
| 0EAA | 136A | | 528 | | DC | Z(TEST6) | TEST 6 | MPC05280 |
| 0EAC | 13FC | | 529 | | DC | Z(TEST7) | TEST 7 | MPC05290 |
| 0EAE | D3B0 | 0A1C | 530 | OPTCHK | LB | R11,ADDRESS | OPTION CHECK | MPC05300 |
| 0EB2 | 48C0 | 0A24 | 531 | | LH | R12,PASLFLG | PASLA | MPC05310 |
| 0EB6 | 2336 | | 532 | | BZS | CMD2 | NO | MPC05320 |
| 0EB8 | 26E1 | | 533 | | AIS | R11,1 | YES, INCREMENT DISPALY ADDRESS | MPC05330 |
| 0EBA | DEB0 | 0A22 | 534 | | OC | R11,WRTCMD | WRITE COMMAND TO CRT | MPC05340 |
| 0EBE | 27B1 | | 535 | | SIS | R11,1 | RESTORE DISPLAY ADDRESS | MPC05350 |
| 0ECO | 2303 | | 536 | | BS | MSGTST | CONTINUE | MPC05360 |
| 0EC2 | DEB0 | 0A22 | 537 | CMD2 | CC | R11,WRTCMD | WRITE COMMAND TO TTY | MPC05370 |
| 0EC6 | 4810 | 0C20 | 538 | MSGTST | LH | R1,NOMSG | IS NOMSG OPT SET ? | MPC05380 |
| 0ECA | 2136 | | 539 | | BNZS | DISTOT | YES, INCREMENT COUNTERS | MPC05390 |
| 0ECC | DEB0 | 0A30 | 540 | | CC | R11,RDCMD | READ | MPC05400 |
| 0EDO | 9DEC | | 541 | | SSR | R11,R12 | NO, IS TTY DU ? | MPC05410 |
| 0ED2 | 4310 | 0EF4 | 542 | | BNM | CONCHK | NO, CHECK CONTINUE OPT | MPC05420 |
| 0ED6 | 2411 | | 543 | DISTOT | LIS | R1,1 | | MPC05430 |
| 0ED8 | 6110 | 21D4 | 544 | | AHM | R1,TOTAL | YES, INCREMENT TOTAL COUNT | MPC05440 |
| 0EDC | 4870 | 21D4 | 545 | | LH | R7,TOTAL | | MPC05450 |
| 0EE0 | 41E0 | 1DE0 | 546 | | BAL | R14,WRITE | WRITE CURRENT COUNT ON DISPLAY | MPC05460 |

| | | | | | | | |
|------|------|------|-----|--------|-------------|----------------------------------|-------------------------|
| 0EE4 | DEB0 | 0A30 | 547 | OC | R11,RDCMD | READ | MPC05470 |
| 0EE8 | 9DBC | | 548 | SSR | R11,R12 | IS TTY DU ? | MPC05480 |
| 0EEA | 2315 | | 549 | BNMS | CONCHK | NO, CHECK CONTINUE OPT | MPC05490 |
| 0EEC | 40B0 | 0A28 | 550 | STH | R11,TTYFLG | YES, SET TTY OFF FLAG | MPC05500 |
| 0EFO | 4300 | 0E3C | 551 | B | SELTST | SELECT NEXT TEST | MPC05510 |
| 0EF4 | 4810 | 0C28 | 552 | CONCHK | LH | R1,CONTIN | IS THE CONTIN OPT SET ? |
| 0EF8 | 2335 | | 553 | BZS | TTYCHK | NO, CHECK TTY FLAG | MPC05530 |
| 0EFA | 41F0 | 1E2C | 554 | BAL | R15,TSTBRK | CHECK FOR BREAK | MPC05540 |
| 0EFE | 4300 | 0E3C | 555 | B | SELTST | | MPC05550 |
| 0F02 | 48C0 | 0A28 | 556 | TTYCHK | LH | R12,TTYFLG | WAS TTY TURNED OFF |
| 0F06 | 4330 | 0C7C | 557 | BZ | TTYIN | NO, CONTINUE | MPC05560 |
| 0F0A | 4810 | 21D4 | 558 | LH | R1,TOTAL | YES, DISPLAY TEST & ERROR COUNT | MPC05580 |
| 0F0E | 41E0 | 1DF2 | 559 | BAL | R14,CONVERT | WHEN TTY BECOMES AVAILABLE AGAIN | MPC05590 |
| 0F12 | 000C | | 560 | DC | X'C' | SHIFT INDEX | MPC05600 |
| 0F14 | 210C | | 561 | DC | Z(TOTALMSG) | STORE INDEX | MPC05610 |
| 0F16 | 41F0 | 1EA4 | 562 | BAL | R15,PRINT | PRINT TOTAL PASS COUNT | MPC05620 |
| 0F1A | 210A | | 563 | DC | Z(TOTMSG) | START ADRS OF MESSAGE | MPC05630 |
| 0F1C | 2117 | | 564 | DC | Z(TOTALEND) | END ADRS OF MESSAGE | MPC05640 |
| 0F1E | 4810 | 21D6 | 565 | LH | R1,TOTALERR | LOAD TOTAL ERROR COUNT | MPC05650 |
| 0F22 | 41E0 | 1DF2 | 566 | BAL | R14,CONVERT | CONVERT TO ASCII CHARS | MPC05660 |
| 0F26 | 000C | | 567 | DC | X'C' | SHIFT INDEX | MPC05670 |
| 0F28 | 210C | | 568 | DC | Z(TOTALMSG) | STORE INDEX | MPC05680 |
| 0F2A | 41F0 | 1EA4 | 569 | BAL | R15,PRINT | PRINT TOTAL ERROR COUNT | MPC05690 |
| 0F2E | 210C | | 570 | DC | Z(TOTALMSG) | START ADRS OF MESSAGE | MPC05700 |
| 0F30 | 211F | | 571 | DC | Z(ERROREND) | END ADRS OF MESSAGE | MPC05710 |
| 0F32 | 4300 | 0C7C | 572 | B | TTYIN | RETURN TO COMMAND MODE | MPC05720 |
| 0F36 | | | 573 | BADR8 | DS | 2 | MPC05730 |

TEST 1

```

575 *****
576 *
577 *
578 *
579 * PURPOSE
580 * TO INSURE THE MEMORY PROTECT CONTROLLER IS INACTIVE WHEN
581 * PSW 7 IS ZERO OR WHEN AN OFF COMMAND IS ISSUED TO IT. ALSO,
582 * TO INSURE THAT NO INTERRUPTS ARE GENERATED WHEN A DISARM
583 * COMMAND IS ISSUED.
584 *
585 *
586 * DESIGN SPECIFICATIONS
587 * A MEMORY PROTECT MAP IS GENERATED PROTECTING ALL BLOCKS
588 * BETWEEN THE REFERENCE BLOCK AND THE LAST BLOCK OF MEMORY
589 * FROM WRITE, READ AND EXECUTE OPERATIONS.
590 * THESE BLOCKS ARE TESTED FOR PROTECTION BY WRITING TO THE
591 * FIRST LOCATION OF EACH BLOCK, READING FROM THIS LOCATION AND
592 * EXECUTING THIS LOCATION WHICH CONTAINS A RETURN INSTRUCTION.
593 * EACH BLOCK IS TESTED UNDER THREE CONDITIONS:
594 * (1) PSW7 = 0 WITH THE CONTROLLER IN THE WRITE-READ PROTECT
595 * MODE.
596 * (2) PSW7 = 1 WITH THE CONTROLLER IN THE OFF MODE.
597 * (3) PSW7 = 1 WITH THE CONTROLLER IN THE WRITE-READ PROTECT
598 * MODE WITH INTERRUPTS DISARMED.
599 * UNDER CONDITIONS 1 & 2 THE CONTROLLER IS TO REMAIN INACTIVE.
600 * UNDER CONDITION 3 NO INTERRUPTS ARE TO BE GENERATED BUT STATUS
601 * IS SENSED FOR PROPER CONDITIONS.
602 *
603 *
604 *****
605 TEST1 BAL R15,TSTNUM TEST 1
606 LIS R15,1 PROTECT PATTERN = 1
607 STH R15,PROTPATT PROTECT EVERY BLOCK
608 BAL R15,PTNONE CLEAR MEMORY PROTECT MAP
609 BAL R15,PTBLOCKS GENERATE WRITE-READ PROTECT MAP
610 LIS R2,1 SET WRITE-READ COMMAND FLAG
611 LIS EXPT,X'F' EXECUTE PROTECT FLAG
612 BAL R15,PTBLOCKS GENERATE EXECUTE PROTECT MAP
613 BAL R15,LOADMAP LOAD MEMORY PROTECT MAP
614 AIR R10,R14 ACK NO PENDING INTERRUPTS
615 LHR R10,R10 DEVICE CODE 0
616 BNZ ERR17 ELSE ERROR
617 CLHI R14,X'04' STATUS X'04'
618 BNE ERR10 ELSE ERROR
619 LHI R15,TST1B EXIT ADDRESS
620 STH R15,TESTEND
621 SIS BLK,1 DECREMENT
622 TST1A SVC 0,0 ZERO PSW 7
623 BAL R15,BLKADR FIRST & LAST ADDRESS OF NEXT BLOCK
624 SVC 7,0 WRITE & READ PROTECT COMMAND
625 LHI R15,X'030F' "BR R15" INSTRUCTION
626 LIS INCQR,3
627 STH R15,0(R12) WRITE TO FIRST ADDRESS

```

```

OF38 41F0 1D96
OF3C 24F1
OF3E 40F0 2274
OF42 41F0 1678
OF46 41F0 18A0
OF4A 2421
OF4C 24AF
OF4E 41F0 18A0
OF52 41F0 1690
OF56 9FAE
OF58 08AA
OF5A 4230 1FE8
OF5E C5E0 0004
OF62 4230 1FA2
OF66 C8F0 OFF8
OF6A 40F0 2278
OF6E 2751
OF70 E100 0000
OF74 41F0 16F4
OF78 E170 0000
OF7C C8F0 030F
OF80 2493
OF82 40FC 0000

```

```

MPC05750
MPC05760
MPC05770
MPC05780
MPC05790
MPC05800
MPC05810
MPC05820
MPC05830
MPC05840
MPC05850
MPC05860
MPC05870
MPC05880
MPC05890
MPC05900
MPC05910
MPC05920
MPC05930
MPC05940
MPC05950
MPC05960
MPC05970
MPC05980
MPC05990
MPC06000
MPC06010
MPC06020
MPC06030
MPC06040
MPC06050
MPC06060
MPC06070
MPC06080
MPC06090
MPC06100
MPC06110
MPC06120
MPC06130
MPC06140
MPC06150
MPC06160
MPC06170
MPC06180
MPC06190
MPC06200
MPC06210
MPC06220
MPC06230
MPC06240
MPC06250
MPC06260
MPC06270

```

TEST 1

| | | | | | | | |
|------|------|------|-----|-------|-------------|------------------------------------|----------|
| OF86 | 48FC | 0000 | 628 | LH | R15,0(R12) | READ FROM FIRST ADDRESS | MPC06280 |
| OF8A | 4140 | 15D4 | 629 | BAL | NO,RORW | NO READ OR WRITE VIOLATIONS | MPC06290 |
| OF8E | 01FC | | 630 | BALR | R15,R12 | EXECUTE FIRST ADDRESS | MPC06300 |
| OF90 | 4140 | 15CC | 631 | BAL | NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC06310 |
| OF94 | E160 | 0000 | 632 | SVC | 6,0 | OFF PROTECT COMMAND | MPC06320 |
| OF98 | 48E0 | 0C30 | 633 | LH | R11,PRADR | GET MEMORY PROTECT ADDRESS | MPC06330 |
| OF9C | DEB0 | 0A38 | 634 | OC | R11,POFFWR | OFF PROTECT WITH W-R BIT SET | MPC06340 |
| 0FA0 | C8F0 | 030F | 635 | LHI | R15,X'030F' | "BR R15" INSTRUCTION | MPC06350 |
| 0FA4 | 40FC | 0000 | 636 | STH | R15,0(R12) | WRITE TO FIRST ADDRESS | MPC06360 |
| 0FA8 | 48FC | 0000 | 637 | LH | R15,0(R12) | READ FROM FIRST ADDRESS | MPC06370 |
| 0FAC | 4140 | 15D4 | 638 | BAL | NO,RORW | NO READ OR WRITE VIOLATIONS | MPC06380 |
| 0FB0 | 01FC | | 639 | BALR | R15,R12 | EXECUTE FIRST ADDRESS | MPC06390 |
| 0FB2 | 4140 | 15CC | 640 | BAL | NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC06400 |
| 0FB6 | E100 | 0000 | 641 | SVC | 0,0 | ZERO PSW 7 | MPC06410 |
| 0FBA | DEB0 | 0A39 | 642 | OC | R11,DISARM | DISARM COMMAND, WRITE-READ BIT SET | MPC06420 |
| 0FBE | E140 | 0000 | 643 | SVC | 4,0 | SET PSW 7 | MPC06430 |
| 0FC2 | C8F0 | 030F | 644 | LHI | R15,X'030F' | "BR R15" INSTRUCTION | MPC06440 |
| 0FC6 | 40FC | 0000 | 645 | STH | R15,0(R12) | WRITE TO FIRST ADDRESS | MPC06450 |
| 0FCA | 4140 | 15BC | 646 | BAL | NO,WRT | NO WRITE VIOLATIONS | MPC06460 |
| 0FCE | C8E0 | 0034 | 647 | LHI | R14,X'34' | WRITE VIOLATION STATUS | MPC06470 |
| 0FD2 | 4140 | 15E2 | 648 | BAL | NO,STATERR | TEST STATUS | MPC06480 |
| 0FD6 | 48FC | 0000 | 649 | LH | R15,0(R12) | READ FROM FIRST ADDRESS | MPC06490 |
| 0FDA | 4140 | 15C4 | 650 | BAL | NO,READ | NO READ VIOLATIONS | MPC06500 |
| 0FDE | C8E0 | 0064 | 651 | LHI | R14,X'64' | READ VIOLATION STATUS | MPC06510 |
| 0FE2 | 4140 | 15E2 | 652 | BAL | NO,STATERR | TEST STATUS | MPC06520 |
| 0FE6 | 01FC | | 653 | BALR | R15,R12 | EXECUTE FIRST ADDRESS | MPC06530 |
| 0FE8 | 4140 | 15CC | 654 | BAL | NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC06540 |
| 0FEC | C8E0 | 00E4 | 655 | LHI | R14,X'E4' | EX & READ VIOLATION STATUS | MPC06550 |
| 0FF0 | 4140 | 15E2 | 656 | BAL | NO,STATERR | TEST STATUS | MPC06560 |
| 0FF4 | 4300 | 0F70 | 657 | B | TST1A | NEXT BLOCK | MPC06570 |
| 0FF8 | 4300 | 1B02 | 658 | TST1B | B | TSTCHK | MPC06580 |

TEST 2

```

660 *****
661 *
662 *
663 *
664 * PURPOSE
665 * TO INSURE THAT MEMORY CAN BE WRITE AND WRITE-READ
666 * PROTECTED.
667 *
668 * DESIGN SPECIFICATION
669 * VARIOUS MEMORY PROTECT MAPS ARE GENERATED AND SENT TO
670 * THE MEMORY PROTECT CONTROLLER. FIRST A MAP IS GENERATED
671 * FOR EVERY BLOCK BETWEEN THE REFERENCE BLOCK AND THE LAST
672 * BLOCK OF MEMORY, PROTECTING ONLY THAT BLOCK AT A GIVEN
673 * TIME.
674 * NEXT MAPS ARE GENERATED TO PROTECT NO BLOCKS, EVERY
675 * SECOND BLOCK, EVERY THIRD BLOCK, ETC.
676 * AFTER EACH MAP IS GENERATED THE BLOCKS OF MEMORY ARE
677 * TESTED FOR PROTECTION BY WRITING TO THE FIRST AND LAST
678 * LOCATION OF THE BLOCK AND READING FROM THESE LOCATIONS.
679 * THE TEST IS PERFORMED WITH THE CONTROLLER IN THE WRITE
680 * PROTECT MODE AND IS REPEATED FOR THE WRITE-READ PROTECT MODE.
681 *
682 *
683 * NOTE: IN THE WRITE MODE, A PROTECT VIOLATION CAUSED BY
684 * A STORE BYTE (STB) INSTRUCTION IS CONSIDERED A WRITE
685 * VIOLATION, BUT IS CONSIDERED A READ VIOLATION IN THE WRITE-READ
686 * PROTECT MODE.
687 *
688 *
689 *****
690 OFFC 41F0 1D96 TEST2 BAL R15,TSTNUM TEST 2
691 1000 C8F0 10A6 LHI R15,TST2G GET EXIT ADDRESS
692 1004 40F0 2278 STH R15,TESTEND
693 1008 48F0 0C30 TST2A LH R15,PRTADR GET DEVICE ADR
694 100C DEF2 0A35 OC R15,WTONLY(R2) ISSUE WRITE(READ) PROTECT COMMAND
695 1010 2410 LIS R1,0 INITIALIZE MAP FLAG
696 1012 4050 2260 TST2B STH BLK,BLOCK2 PROTECT THIS BLOCK
697 1016 41F0 1616 BAL R15,PTONEBLK GENERATE WRITE-READ PROTECT MAP
698 101A 41F0 1690 TST2C BAL R15,LOADMAP LOAD MEMORY PROTECT MAP
699 101E 4850 219C LH BLK,BLOCKREF REFERENCE BLOCK FIRST
700 1022 27E1 SIS BLK,1 INITIALIZE BLOCK VALUE
701 1024 41F0 16F4 TST2D BAL R15,BLKADR FIRST & LAST ADDRESS OF NEXT BLOCK
702 1028 25F1 LCS R15,1 DATA X'FFFF'
703 102A 40FC 0000 STH R15,0(R12) WRITE TO FIRST ADDRESS
704 102E 40FE 0000 STH R15,0(R14) WRITE TO LAST ADDRESS
705 1032 E140 0000 SVC 4,0 SET PSW 7
706 1036 40CC 0000 STH R12,0(R12) WRITE TO FIRST ADDRESS
707 103A D2CC 0001 STB R12,1(R12) STORE BYTE
708 103E 40EE 0000 STH R14,0(R14) WRITE TO LAST ADDRESS
709 1042 D2EE 0001 STB R14,1(R14) STORE BYTE
710 1046 48BC 0000 LH R11,0(R12) READ FROM FIRST ADDRESS
711 104A 48DE 0000 LH R13,0(R14) READ FROM LAST ADDRESS
712 104E E100 0000 SVC 0,0 ZERO PSW 7
MPC06600
MPC06610
MPC06620
MPC06630
MPC06640
MPC06650
MPC06660
MPC06670
MPC06680
MPC06690
MPC06700
MPC06710
MPC06720
MPC06730
MPC06740
MPC06750
MPC06760
MPC06770
MPC06780
MPC06790
MPC06800
MPC06810
MPC06820
MPC06830
MPC06840
MPC06850
MPC06860
MPC06870
MPC06880
MPC06890
MPC06900
MPC06910
MPC06920
MPC06930
MPC06940
MPC06950
MPC06960
MPC06970
MPC06980
MPC06990
MPC07000
MPC07010
MPC07020
MPC07030
MPC07040
MPC07050
MPC07060
MPC07070
MPC07080
MPC07090
MPC07100
MPC07110
MPC07120

```

TEST 2

| | | | | | | |
|------|-----------|-----|--------|------------------|------------------------------|----------|
| 1052 | 0799 | 713 | XHR | INCQR,INCQR | INVALID ILLEGAL INS. | MPC07130 |
| 1054 | 4140 15CC | 714 | BAL | NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC07140 |
| 1058 | 48A0 2268 | 715 | LH | R10,BLKPT | BLOCK WRITE-READ PROTECTED? | MPC07150 |
| 105C | 4330 1092 | 716 | BZ | TST2F | NO | MPC07160 |
| 1060 | 0822 | 717 | LHR | R2,R2 | WRITE-READ COMMAND? | MPC07170 |
| 1062 | 233A | 718 | BZS | TST2E | NO, THEN NO READ VIOLATIONS | MPC07180 |
| 1064 | C960 0004 | 719 | CHI | INTCR,4 | FOUR READ VIOLATIONS | MPC07190 |
| 1068 | 4230 1F7A | 720 | BNE | ERR5 | ELSE ERROR | MPC07200 |
| 106C | C970 0002 | 721 | CHI | INTCW,2 | TWO WRITE VIOLATIONS | MPC07210 |
| 1070 | 4230 1F72 | 722 | BNE | ERR4 | ELSE ERROR | MPC07220 |
| 1074 | 2307 | 723 | BS | TST2EE | NEXT BLOCK | MPC07230 |
| 1076 | 4140 15C4 | 724 | TST2E | BAL NO,READ | NO READ VIOLATIONS | MPC07240 |
| 107A | C970 0004 | 725 | CHI | INTCW,4 | FOUR WRITE VIOLATIONS | MPC07250 |
| 107E | 4230 1F72 | 726 | BNE | ERR4 | ELSE ERROR | MPC07260 |
| 1082 | 05BF | 727 | TST2EE | CLHR R11,R15 | EXPECT ORG DATA | MPC07270 |
| 1084 | 4230 200C | 728 | BNE | ERR21 | ELSE ERROR | MPC07280 |
| 1088 | 05DF | 729 | CLHR | R13,R15 | EXPECT ORG DATA | MPC07290 |
| 108A | 4230 2014 | 730 | BNE | ERR22 | ELSE ERROR | MPC07300 |
| 108E | 4300 1024 | 731 | B | TST2D | NEXT BLOCK | MPC07310 |
| 1092 | 4140 15D4 | 732 | TST2F | BAL NO,RORW | NO READ VIOLATIONS | MPC07320 |
| 1096 | 05BC | 733 | CLHR | R11,R12 | NEW DATA | MPC07330 |
| 1098 | 4230 201E | 734 | BNE | ERR23 | ELSE ERROR | MPC07340 |
| 109C | 05DE | 735 | CLHR | R13,R14 | NEW DATA | MPC07350 |
| 109E | 4230 2026 | 736 | BNE | ERR24 | ELSE ERROR | MPC07360 |
| 10A2 | 4300 1024 | 737 | B | TST2D | NEXT BLOCK | MPC07370 |
| 10A6 | 0811 | 738 | TST2G | LHR R1,R1 | MAP FLAG | MPC07380 |
| 10A8 | 2135 | 739 | BNZS | TST2H | GENERATE NEW PROTECT MAP | MPC07390 |
| 10AA | 41F0 17CA | 740 | BAL | R15,INCBLK | ELSE PROTECT NEXT BLOCK | MPC07400 |
| 10AE | 4320 1012 | 741 | BFC | 2,TST2B | MORE BLOCKS? | MPC07410 |
| 10B2 | 2411 | 742 | TST2H | LIS R1,1 | NO, SET MAP FLAG TO 1 | MPC07420 |
| 10B4 | 41F0 1678 | 743 | BAL | R15,PTNONE | CLEAR MEMORY PROTECT MAP | MPC07430 |
| 10B8 | 48F0 2274 | 744 | LH | R15,PROTPATT | GET PROTECT PATTERN | MPC07440 |
| 10BC | 233A | 745 | BZS | TST2J | PROTECT NONE FIRST | MPC07450 |
| 10BE | 41F0 18A0 | 746 | BAL | R15,PTBLOCKS | ELSE PROTECT THESE BLOCKS | MPC07460 |
| 10C2 | 48F0 2274 | 747 | LH | R15,PROTPATT | GET PATTERN AGAIN | MPC07470 |
| 10C6 | 26F1 | 748 | AIS | R15,1 | ADD 1 | MPC07480 |
| 10C8 | 49F0 2198 | 749 | CH | R15,BLOCKMP | GREATER THAN MAXIMUM? | MPC07490 |
| 10CC | 212B | 750 | BTFS | 2,TST2Z | YES, CHECK WRITE-READ STATUS | MPC07500 |
| 10CE | 2302 | 751 | BS | TST2K | ELSE GENERATE NEW MAP | MPC07510 |
| 10D0 | 24F1 | 752 | TST2J | LIS R15,1 | ONE | MPC07520 |
| 10D2 | 40F0 2274 | 753 | TST2K | STH R15,PROTPATT | NEW PATTERN | MPC07530 |
| 10D6 | 4850 219C | 754 | LH | BLK,BLOCKREF | GET BLOCK REFERENCE VALUE | MPC07540 |
| 10DA | 4050 2260 | 755 | STH | BLK,BLOCK2 | SET BLOCK COUNTER | MPC07550 |
| 10DE | 4300 101A | 756 | B | TST2C | SET MEMORY PROTECT MAP | MPC07560 |
| 10E2 | C920 0001 | 757 | TST2Z | CHI R2,1 | END OF WRITE/READ? | MPC07570 |
| 10E6 | 4330 1B02 | 758 | BE | TSTCHK | YES, EXIT | MPC07580 |
| 10EA | 48F0 0C40 | 759 | LH | R15,MODEL70 | MODEL 70? | MPC07590 |
| 10EE | 4230 1B02 | 760 | BNZ | TSTCHK | YES, EXIT | MPC07600 |
| 10F2 | 4850 219C | 761 | LH | BLK,BLOCKREF | GET REFERENCE BLOCK | MPC07610 |
| 10F6 | 24F0 | 762 | LIS | R15,0 | | MPC07620 |
| 10F8 | 40F0 2274 | 763 | STH | R15,PROTPATT | ZERO PROTECT PATTERN | MPC07630 |
| 10FC | 2421 | 764 | LIS | R2,1 | WRITE-READ PROTECT COMMAND | MPC07640 |
| 10FE | 4300 1008 | 765 | B | TST2A | REPEAT TEST | MPC07650 |

8/16E MEMORY PROTECT TEST 06-223R01A13

PAGE 17 13:48:37 07/03/79

TEST 2

1102

766 BADR9 DS 2

MPC07660

TEST 3

| | | | | | | |
|------|-----------|-----|---|-----------------------------------|--|----------|
| | | 768 | ***** | | | MPC07680 |
| | | 769 | * | | | MPC07690 |
| | | 770 | * | T E S T 3 | | MPC07700 |
| | | 771 | * | | | MPC07710 |
| | | 772 | * PURPOSE | | | MPC07720 |
| | | 773 | * TO INSURE THAT PROGRAM MEMORY CAN BE WRITE AND WRITE-READ | | | MPC07730 |
| | | 774 | * PROTECTED. | | | MPC07740 |
| | | 775 | * | | | MPC07750 |
| | | 776 | * DESIGN SPECIFESATION | | | MPC07760 |
| | | 777 | * A MEMORY PROTECT MAP IS GENERATED FOR EVERY MEMORY BLOCK | | | MPC07770 |
| | | 778 | * BETWEEN BLOCK 1 AND THE LAST BLOCK OF THE PROGRAM PROTECTING | | | MPC07780 |
| | | 779 | * ONLY THAT BLOCK AT A GIVEN TIME. | | | MPC07790 |
| | | 780 | * AFTER EACH MAP IS GENERATED EVERY BLOCK OF PROGRAM MEMORY | | | MPC07800 |
| | | 781 | * IS TESTED FOR PROTECTION BY WRITING TO A SPECIFIED LOCATION | | | MPC07810 |
| | | 782 | * IN THE BLOCK AND READING FROM THIS LOCATION. | | | MPC07820 |
| | | 783 | * THE TEST IS PERFORMED WITH THE CONTROLLER IN THE WRITE | | | MPC07830 |
| | | 784 | * PROTECT MODE AND IS REPEATED FOR THE WRITE-READ PROTECT MODE. | | | MPC07840 |
| | | 785 | * | | | MPC07850 |
| | | 786 | * | | | MPC07860 |
| | | 787 | ***** | | | MPC07870 |
| 1104 | 41F0 1D96 | 788 | TEST3 BAL R15,TSTNUM | TEST 3 | | MPC07880 |
| 1108 | 2451 | 789 | TST3A LIS BLK,1 | BLOCK 1 FIRST | | MPC07890 |
| 110A | 48F0 0C30 | 790 | TST3B LH R15,PRTADR | GET PROTECT ADDRESS | | MPC07900 |
| 110E | DEF2 0A35 | 791 | OC R15,WTONLY(R2) | ISSUE WRITE(READ) PROTECT COMMAND | | MPC07910 |
| 1112 | 4050 2260 | 792 | TST3C STH BLK,BLOCK2 | PROTECT THIS BLOCK | | MPC07920 |
| 1116 | 41F0 1616 | 793 | BAL R15,PTONEBLK | GENERATE WRITE-READ PROTECT MAP | | MPC07930 |
| 111A | 41F0 1690 | 794 | BAL R15,LOADMAP | LOAD MEMORY PROTECT MAP | | MPC07940 |
| 111E | 24D0 | 795 | TST3D LIS R13,0 | ZERO INDEX | | MPC07950 |
| 1120 | 2451 | 796 | LIS BLK,1 | BLOCK 1 FIRST | | MPC07960 |
| 1122 | 2470 | 797 | TST3E LIS INTCW,0 | CLEAR INTERRUPT COUNTERS | | MPC07970 |
| 1124 | 2460 | 798 | LIS INTCR,0 | | | MPC07980 |
| 1126 | 2480 | 799 | LIS INTCE,0 | | | MPC07990 |
| 1128 | 48CD 21A0 | 800 | LH R12,BADR(R13) | GET ADDRESS IN THIS BLOCK | | MPC08000 |
| 112C | 41F0 1840 | 801 | BAL R15,QBLKPT | GET PROTECT STATUS | | MPC08010 |
| 1130 | 2493 | 802 | LIS INCQR,3 | | | MPC08020 |
| 1132 | 48F0 2260 | 803 | LH R15,BLOCK2 | GET PROTECT BLOCK NUMBER | | MPC08030 |
| 1136 | 45F0 219E | 804 | CLH R15,BLOCKFLG | COMPARE TO BLOCK FLAG | | MPC08040 |
| 113A | 4280 2044 | 805 | BL TEST3X | BRANCH TO ADDR > 1E00 | | MPC08050 |
| 113E | E140 0000 | 806 | SVC 4,0 | SET PSW 7 | | MPC08060 |
| 1142 | 48FC 0000 | 807 | LH R15,0(R12) | WRITE TO THIS ADDRESS | | MPC08070 |
| 1146 | 40FC 0000 | 808 | STH R15,0(R12) | READ FROM THIS ADDRESS | | MPC08080 |
| 114A | E100 0000 | 809 | SVC 0,0 | ZERO PSW 7 | | MPC08090 |
| 114E | 2490 | 810 | TST3F LIS INCQR,0 | | | MPC08100 |
| 1150 | 4140 15CC | 811 | BAL NO,EXECUTE | NO EXECUTE VIOLATIONS | | MPC08110 |
| 1154 | 48F0 2268 | 812 | LH R15,BLKPT | BLOCK WRITE-READ PROTECTED? | | MPC08120 |
| 1158 | 4330 117A | 813 | BZ TST3H | NO | | MPC08130 |
| 115C | C970 0001 | 814 | CHI INTCW,1 | ONE WRITE VIOLATION | | MPC08140 |
| 1160 | 4230 1F72 | 815 | BNE ERR4 | ELSE ERROR | | MPC08150 |
| 1164 | C920 0001 | 816 | CHI R2,1 | WRITE-READ PROTECT COMMAND? | | MPC08160 |
| 1168 | 2334 | 817 | BES TST3G | YES | | MPC08170 |
| 116A | 4140 15C4 | 818 | BAL NO,READ | NO READ VIOLATIONS | | MPC08180 |
| 116E | 2308 | 819 | BS TST3J | | | MPC08190 |
| 1170 | C960 0001 | 820 | TST3G CHI INTCR,1 | ONE READ VIOLATION | | MPC08200 |

TEST 3

| | | | | | | | |
|------|------|------|-----|-------|-------|---------------|----------|
| 1174 | 4230 | 1F7A | 821 | BNE | ERR5 | ELSE ERROR | MPC08210 |
| 1178 | 2303 | | 822 | BS | TST3J | NEXT BLOCK | MPC08220 |
| 117A | 4140 | 15D4 | 823 | TST3H | BAL | NO,RORW | MPC08230 |
| 117E | 4130 | 17DC | 824 | TST3J | BAL | RET1,NEXTBLK | MPC08240 |
| 1182 | 4280 | 1122 | 825 | | BL | TST3E | MPC08250 |
| 1186 | 4130 | 1782 | 826 | | BAL | RET1,NEXTPSW | MPC08260 |
| 118A | 2125 | | 827 | | BTFS | 2,TST3Z | MPC08270 |
| 118C | 41E0 | 179C | 828 | | BAL | RET2,DISPLAY | MPC08280 |
| 1190 | 4300 | 111E | 829 | | B | TST3D | MPC08290 |
| 1194 | 41E0 | 17EC | 830 | TST3Z | BAL | RET2,INCLOWBL | MPC08300 |
| 1198 | 4280 | 1112 | 831 | | BL | TST3C | MPC08310 |
| 119C | C920 | 0001 | 832 | | CHI | R2,1 | MPC08320 |
| 11A0 | 4330 | 1B02 | 833 | | BE | TSTCHK | MPC08330 |
| 11A4 | 48F0 | 0C40 | 834 | | LH | R15,MODEL70 | MPC08340 |
| 11A8 | 4230 | 1B02 | 835 | | BNZ | TSTCHK | MPC08350 |
| 11AC | 2421 | | 836 | | LIS | R2,1 | MPC08360 |
| 11AE | 4300 | 1108 | 837 | | B | TST3A | MPC08370 |

TEST 4

| | | | | | | | |
|------|-----------|-----|-------|--|------------------------------------|--|----------|
| | | 839 | ***** | | | | MPC08390 |
| | | 840 | * | | | | MPC08400 |
| | | 841 | * | | T E S T 4 | | MPC08410 |
| | | 842 | * | | | | MPC08420 |
| | | 843 | * | PURPOSE | | | MPC08430 |
| | | 844 | * | TO INSURE THAT MEMORY CAN BE EXECUTE PROTECTED. | | | MPC08440 |
| | | 845 | * | | | | MPC08450 |
| | | 846 | * | | | | MPC08460 |
| | | 847 | * | DESIGN SPECIFICATION | | | MPC08470 |
| | | 848 | * | A MEMORY PROTECT MAP IS GENERATED FOR EVERY BLOCK | | | MPC08480 |
| | | 849 | * | BETWEEN THE REFERENCE BLOCK AND THE LAST BLOCK OF MEMORY | | | MPC08490 |
| | | 850 | * | PROTECTING ONLY THAT BLOCK AT A GIVEN TIME. | | | MPC08500 |
| | | 851 | * | AFTER EACH MAP IS GENERATED THE BLOCKS ARE TESTED FOR | | | MPC08510 |
| | | 852 | * | EXECUTE PROTECTION BY EXECUTING THE FIRST AND LAST MEMORY | | | MPC08520 |
| | | 853 | * | LOCATIONS OF EACH BLOCK WHICH CONTAINS A RETURN INSTRUCTION. | | | MPC08530 |
| | | 854 | * | THIS TEST IS PERFORMED WITH THE CONTROLLER IN THE WRITE | | | MPC08540 |
| | | 855 | * | PROTECT MODE. | | | MPC08550 |
| | | 856 | * | | | | MPC08560 |
| | | 857 | * | | | | MPC08570 |
| | | 858 | ***** | | | | MPC08580 |
| 11B2 | 41F0 1D96 | 859 | TEST4 | BAL R15,TSTNUM | TEST NUMBER 4 | | MPC08590 |
| 11B6 | 48F0 0C40 | 860 | | LH R15,MODEL70 | IS IT MODEL 70 | | MPC08600 |
| 11BA | 4230 0C72 | 861 | | BNZ QUESTNZ | YES,DONOT EXECUTE THIS TEST | | MPC08610 |
| 11BE | C8F0 1236 | 862 | | LHI R15,TST4Z | EXIT ADDRESS | | MPC08620 |
| 11C2 | 40F0 2278 | 863 | | STH R15,TESTEND | | | MPC08630 |
| 11C6 | E160 0000 | 864 | | SVC 6,0 | OFF PROTECT COMMAND | | MPC08640 |
| 11CA | E100 0000 | 865 | TST4A | SVC 0,0 | ZERO PSW 7 | | MPC08650 |
| 11CE | 24AF | 866 | | LIS EXPT,X'F' | EXECUTE PROTECT FLAG | | MPC08660 |
| 11D0 | 41F0 1616 | 867 | | BAL R15,PTONEBLK | GENERATE EXECUTE PROTECT MAP | | MPC08670 |
| 11D4 | 41F0 1690 | 868 | | BAL R15,LOADMAP | LOAD MEMORY PROTECT MAP | | MPC08680 |
| 11D8 | 4850 219C | 869 | | LH BLK,BLOCKREF | GET BLOCK REFERENCE | | MPC08690 |
| 11DC | 2751 | 870 | | SIS BLK,1 | INITIALIZE BLOCK VALUE | | MPC08700 |
| 11DE | 41F0 16F4 | 871 | TST4B | BAL R15,BLKADR | FIRST & LAST ADDRESS OF NEXT BLOCK | | MPC08710 |
| 11E2 | E140 0000 | 872 | | SVC 4,0 | SET PSW 7 | | MPC08720 |
| 11E6 | E150 0000 | 873 | | SVC 5,0 | WRITE ONLY PROTECT COMMAND | | MPC08730 |
| 11EA | C8F0 030F | 874 | | LHI R15,X'030F' | "BR R15" INSTRUCTION | | MPC08740 |
| 11EE | 40FC 0000 | 875 | | STH R15,0(R12) | WRITE TO FIRST ADDRESS | | MPC08750 |
| 11F2 | 40FE 0000 | 876 | | STH R15,0(R14) | WRITE TO LAST ADDRESS | | MPC08760 |
| 11F6 | 48DC 0000 | 877 | | LH R13,0(R12) | READ FROM FIRST ADDRESS | | MPC08770 |
| 11FA | C5D0 030F | 878 | | CLHI R13,X'030F' | EXPECTED RESULTS | | MPC08780 |
| 11FE | 4230 1FD6 | 879 | | BNE ERR15 | ELSE ERROR | | MPC08790 |
| 1202 | 48DE 0000 | 880 | | LH R13,0(R14) | READ FROM LAST ADDRESS | | MPC08800 |
| 1206 | C5D0 030F | 881 | | CLHI R13,X'030F' | EXPECTED RESULTS | | MPC08810 |
| 120A | 4230 1FDE | 882 | | BNE ERR16 | ELSE ERROR | | MPC08820 |
| 120E | 2494 | 883 | | LIS INCQR,4 | | | MPC08830 |
| 1210 | 01FC | 884 | | BALR R15,R12 | EXECUTE FIRST ADDRESS | | MPC08840 |
| 1212 | 01FE | 885 | | BALR R15,R14 | EXECUTE LAST ADDRESS | | MPC08850 |
| 1214 | 2493 | 886 | | LIS INCQR,3 | | | MPC08860 |
| 1216 | E160 0000 | 887 | | SVC 6,0 | OFF PROTECT COMMAND | | MPC08870 |
| 121A | 48A0 226A | 888 | | LH R10,BLKPTE | BLOCK EXECUTE PROTECTED? | | MPC08880 |
| 121E | 2336 | 889 | | BZS TST4C | NO VIOLATIONS | | MPC08890 |
| 1220 | C580 0002 | 890 | | CLHI INTCE,2 | TWO VIOLATIONS | | MPC08900 |
| 1224 | 4230 1F82 | 891 | | BNE ERR6 | ELSE ERROR | | MPC08910 |

TEST #

| | | | | | | |
|------|-----------|-----|-------|----------------|-----------------------------|----------|
| 1228 | 2303 | 892 | BS | TST4D | NO OTHER VIOLATIONS | MPC08920 |
| 122A | 4140 15CC | 893 | TST4C | BAL NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC08930 |
| 122E | 4140 15D4 | 894 | TST4D | BAL NO,RORW | NO WRITE OR READ VIOLATIONS | MPC08940 |
| 1232 | 4300 11DE | 895 | B | TST4B | NEXT BLOCK | MPC08950 |
| 1236 | 41F0 17CA | 896 | TST4Z | BAL R15,INCBLK | PROTECT NEXT BLOCK | MPC08960 |
| 123A | 4320 11CA | 897 | BFC | 2,TST4A | CONTINUE IF MORE BLOCKS | MPC08970 |
| 123E | 4300 1B02 | 898 | B | TSTCHK | NEXT TEST | MPC08980 |

TEST 5

| | | | | | | |
|------|-----------|-----|---|------------------------------------|--|----------|
| | | 900 | ***** | | | MPC09000 |
| | | 901 | * | | | MPC09010 |
| | | 902 | * | T E S T 5 | | MPC09020 |
| | | 903 | * | | | MPC09030 |
| | | 904 | * PURPOSE | | | MPC09040 |
| | | 905 | * TO INSURE THAT MEMORY CAN BE WRITE, READ AND EXECUTE | | | MPC09050 |
| | | 906 | * PROTECTED. | | | MPC09060 |
| | | 907 | * | | | MPC09070 |
| | | 908 | * | | | MPC09080 |
| | | 909 | * DESIGN SPECIFICATION | | | MPC09090 |
| | | 910 | * FIRST, MEMORY PROTECT MAPS ARE GENERATED FOR EVERY BLOCK | | | MPC09100 |
| | | 911 | * BETWEEN THE REFERENCE BLOCK AND THE LAST BLOCK OF MEMORY | | | MPC09110 |
| | | 912 | * PROTECTING ONLY THAT BLOCK AT A GIVEN TIME. | | | MPC09120 |
| | | 913 | * NEXT, A MAP IS GENERATED PROTECTING EVERY OTHER BLOCK | | | MPC09130 |
| | | 914 | * FROM WRITE(READ) OPERATIONS AND THE REMAINING BLOCKS FROM | | | MPC09140 |
| | | 915 | * EXECUTE OPERATIONS. | | | MPC09150 |
| | | 916 | * AFTER EACH MAP IS GENERATED THE BLOCKS ARE TESTED FOR PROTECTION* | | | MPC09160 |
| | | 917 | * BY WRITING TO THE FIRST AND LAST LOCATION OF EACH BLOCK READING | | | MPC09170 |
| | | 918 | * FROM THESE LOCATIONS AND EXECUTING THESE LOACTIONS WHICH CONTAIN | | | MPC09180 |
| | | 919 | * A RETURN INSTRUCTION. | | | MPC09190 |
| | | 920 | * THE TEST IS PERFORMED WITH THE CONTROLLER IN THE WRITE | | | MPC09200 |
| | | 921 | * PROTECT MODE AND IS REPEATED FOR THE WRITE-READ PROTECT MODE. | | | MPC09210 |
| | | 922 | * | | | MPC09220 |
| | | 923 | * | | | MPC09230 |
| | | 924 | ***** | | | MPC09240 |
| 1242 | 41F0 1D96 | 925 | TEST5 BAL R15,TSTNUM | TEST 5 | | MPC09250 |
| 1246 | 48F0 0C40 | 926 | LH R15,MODEL70 | | | MPC09260 |
| 124A | 4230 0C72 | 927 | BNZ QUESTNZ | ??? | | MPC09270 |
| 124E | C8F0 131E | 928 | LHI R15,TST5Z | EXIT ADDRESS | | MPC09280 |
| 1252 | 40F0 2278 | 929 | STH R15,TESTEND | | | MPC09290 |
| 1256 | 2430 | 930 | TST5A LIS R3,0 | INITIALIZE MAP FLAG | | MPC09300 |
| 1258 | E160 0000 | 931 | SVC 6,0 | OFF PROTECT COMMAND | | MPC09310 |
| 125C | E100 0000 | 932 | SVC 0,0 | ZERO PSW 7 | | MPC09320 |
| 1260 | 4050 2260 | 933 | STH BLK,BLOCK2 | PROTECT THIS BLOCK | | MPC09330 |
| 1264 | 41F0 1616 | 934 | BAL R15,PTONEBLK | GENERATE WRITE-READ PROTECT MAP | | MPC09340 |
| 1268 | 24AF | 935 | LIS EXPT,X'F' | EXECUTE PROTECT FLAG | | MPC09350 |
| 126A | 41F0 1620 | 936 | BAL R15,PTALLBLK | GENERATE EXECUTE PROTECT MAP | | MPC09360 |
| 126E | 41F0 1690 | 937 | TST5B BAL R15,LOADMAP | LOAD MEMORY PROTECT MAP | | MPC09370 |
| 1272 | 4850 219C | 938 | LH BLK,BLOCKREF | GET REFERNCE BLOCK | | MPC09380 |
| 1276 | 2751 | 939 | SIS BLK,1 | INITIALIZE BLOCK VALUE | | MPC09390 |
| 1278 | 41F0 16F4 | 940 | TST5C BAL R15,BLKADR | FIRST & LAST ADDRESS OF NEXT BLOCK | | MPC09400 |
| 127C | C8F0 030F | 941 | LHI R15,X'030F' | "BR R15" INSTRUCTIONON | | MPC09410 |
| 1280 | 40FC 0000 | 942 | STH R15,0(R12) | WRITE TO FIRST ADDRESS | | MPC09420 |
| 1284 | 40FE 0000 | 943 | STH R15,0(R14) | WRITE TO LAST ADDRESS | | MPC09430 |
| 1288 | E140 0000 | 944 | SVC 4,0 | SET PSW 7 | | MPC09440 |
| 128C | 0822 | 945 | LHR R2,R2 | GET WRITE-READ PROTECT FLAG | | MPC09450 |
| 128E | 2134 | 946 | BNZS TST5D | | | MPC09460 |
| 1290 | E150 0000 | 947 | SVC 5,0 | WRITE PROTECT COMMAND | | MPC09470 |
| 1294 | 2303 | 948 | BS TST5E | | | MPC09480 |
| 1296 | E170 0000 | 949 | TST5D SVC 7,0 | WRITE & READ PROTECT COMMAND | | MPC09490 |
| 129A | 48DC 0000 | 950 | TST5E LH R13,0(R12) | READ FROM FIRST ADDRESS | | MPC09500 |
| 129E | C5D0 030F | 951 | CLHI R13,X'030F' | EXPECTED DATA | | MPC09510 |
| 12A2 | 4230 1FD6 | 952 | BNE ERR15 | ELSE ERROR | | MPC09520 |

TEST 5

| | | | | | | |
|------|-----------|------|------|--------------|------------------------------------|----------|
| 12A6 | 48DE 0000 | 953 | LH | R13,0(R14) | READ FROM LAST ADDRESS | MPC09530 |
| 12AA | C5D0 030F | 954 | CLHI | R13,X'030F' | EXPECTED DATA | MPC09540 |
| 12AE | 4230 1FDE | 955 | BNE | ERR16 | ELSE ERROR | MPC09550 |
| 12B2 | C8F0 030F | 956 | LHI | R15,X'030F' | "BR R15" INSTRUCTIONON | MPC09560 |
| 12B6 | 40FC 0000 | 957 | STH | R15,0(R12) | WRITE TO FIRST ADDRESS | MPC09570 |
| 12BA | 40FE 0000 | 958 | STH | R15,0(R14) | WRITE TO LAST ADDRESS | MPC09580 |
| 12BE | 2494 | 959 | LIS | INCQR,4 | | MPC09590 |
| 12C0 | 01FC | 960 | BALR | R15,R12 | EXECUTE FIRST ADDRESS | MPC09600 |
| 12C2 | 01FE | 961 | BALR | R15,R14 | EXECUTE LAST ADDRESS | MPC09610 |
| 12C4 | 2493 | 962 | LIS | INCQR,3 | | MPC09620 |
| 12C6 | E160 0000 | 963 | SVC | 6,0 | OFF PROTECT COMMAND | MPC09630 |
| 12CA | E100 0000 | 964 | SVC | 0,0 | ZERO PSW 7 | MPC09640 |
| 12CE | 48A0 2268 | 965 | LH | R10,BLKPT | BLOCK WRITE-READ PROTECTED? | MPC09650 |
| 12D2 | 4330 1302 | 966 | BZ | TST5H | NO | MPC09660 |
| 12D6 | C970 0002 | 967 | CHI | INTCW,2 | TWO WRITE VIOLATIONS | MPC09670 |
| 12DA | 4230 1F72 | 968 | BNE | ERR4 | ELSE ERROR | MPC09680 |
| 12DE | 0822 | 969 | LHR | R2,R2 | GET WRITE-READ COMMAND FLAG | MPC09690 |
| 12E0 | 233E | 970 | BZS | TST5G | WRITE ONLY | MPC09700 |
| 12E2 | C530 0001 | 971 | CLHI | R3,1 | MAP FLAG AT 1 | MPC09710 |
| 12E6 | 2336 | 972 | BES | TST5F | YES | MPC09720 |
| 12E8 | C960 0002 | 973 | CHI | INTCR,2 | TWO READ VIOLATIONS | MPC09730 |
| 12EC | 4230 1F7A | 974 | BNE | ERR5 | ELSE ERROR | MPC09740 |
| 12F0 | 230B | 975 | BS | TST5J | | MPC09750 |
| 12F2 | C960 0004 | 976 | CHI | INTCR,4 | FOUR READ VIOLATIONS | MPC09760 |
| 12F6 | 4230 1F7A | 977 | BNE | ERR5 | ELSE ERROR | MPC09770 |
| 12FA | 2306 | 978 | BS | TST5J | | MPC09780 |
| 12FC | 4140 15C4 | 979 | BAL | NO,READ | NO READ VIOLATIONS | MPC09790 |
| 1300 | 2303 | 980 | BS | TST5J | | MPC09800 |
| 1302 | 4140 15D4 | 981 | BAL | NO,RORW | NO READ OR WRITE VIOLATIONS | MPC09810 |
| 1306 | 48A0 226A | 982 | LH | R10,BLKPT | BLOCK EXECUTE PROTECTED? | MPC09820 |
| 130A | 2336 | 983 | BZS | TST5K | NO | MPC09830 |
| 130C | C980 0002 | 984 | CHI | INTCE,2 | TWO EXECUTE VIOLATIONS | MPC09840 |
| 1310 | 4230 1F82 | 985 | BNE | ERR6 | ELSE ERROR | MPC09850 |
| 1314 | 2303 | 986 | BS | TST5L | | MPC09860 |
| 1316 | 4140 15CC | 987 | BAL | NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC09870 |
| 131A | 4300 1278 | 988 | B | TST5C | NEXT BLOCK | MPC09880 |
| 131E | C930 0001 | 989 | CHI | R3,1 | MAP FLAG AT 1 | MPC09890 |
| 1322 | 2139 | 990 | BNES | TST5M | NO, THEN GENERATE NEW MAP | MPC09900 |
| 1324 | 0822 | 991 | LHR | R2,R2 | WRITE-READ PROTECT COMMAND ISSUED? | MPC09910 |
| 1326 | 4230 1B02 | 992 | BNZ | TSTCHK | NEXT TEST IF ISSUED | MPC09920 |
| 132A | 2421 | 993 | LIS | R2,1 | SET WRITE-READ PROTECT COMMAND | MPC09930 |
| 132C | 4850 219C | 994 | LH | BLK,BLOCKREF | GET BLOCK REF VALUE | MPC09940 |
| 1330 | 4300 1256 | 995 | B | TST5A | REPEAT TEST | MPC09950 |
| 1334 | 41F0 17CA | 996 | BAL | R15,INCBLK | PROTECT NEXT BLOCK | MPC09960 |
| 1338 | 4320 1256 | 997 | BFC | 2,TST5A | CONTINUE IF MORE BLOCKS | MPC09970 |
| 133C | 2431 | 998 | LIS | R3,1 | SET MAP FLAG TO 1 | MPC09980 |
| 133E | 41F0 1678 | 999 | BAL | R15,PTNONE | CLEAR MEMORY PROTECT MAP | MPC09990 |
| 1342 | 24F2 | 1000 | LIS | R15,2 | | MPC10000 |
| 1344 | 40F0 2274 | 1001 | STH | R15,PROTPATT | PROTECT EVERY SECOND BLOCK | MPC10010 |
| 1348 | 41F0 18A0 | 1002 | BAL | R15,PTBLOCKS | GENERATE WRITE-READ PROTECT MAP | MPC10020 |
| 134C | 24AF | 1003 | LIS | EXPT,X'F' | EXECUTE PROTECT FLAG | MPC10030 |
| 134E | 6130 219C | 1004 | AHM | R3,BLOCKREF | START WITH BLOCK REF + 1 | MPC10040 |
| 1352 | 41F0 18A0 | 1005 | BAL | R15,PTBLOCKS | EX. PROT EVERY SECOND BLOCK | MPC10050 |

TEST 5

| | | | | | | |
|------|-----------|------|--------|--------------|-------------------------------|----------|
| 1356 | 48F0 219C | 1006 | LH | R15,BLOCKREF | RESTORE BLOCK REFERENCE VALUE | MPC10060 |
| 135A | 27F1 | 1007 | SIS | R15,1 | | MPC10070 |
| 135C | 40F0 219C | 1008 | STH | R15,BLOCKREF | | MPC10080 |
| 1360 | 40F0 2260 | 1009 | STH | R15,BLOCK2 | ADJUST VALUE FOR DISPLAY | MPC10090 |
| 1364 | 4300 126E | 1010 | B | TST5B | SET MEMORY PROTECT MAP | MPC10100 |
| 1368 | | 1011 | BADR10 | DS | 2 | MPC10110 |

TEST 6

| | | | | | | |
|------|-----------|------|---|------------------------------|--|----------|
| | | 1013 | ***** | | | MPC10130 |
| | | 1014 | * | | | MPC10140 |
| | | 1015 | * | T E S T 6 | | MPC10150 |
| | | 1016 | * | | | MPC10160 |
| | | 1017 | * PURPOSE | | | MPC10170 |
| | | 1018 | * TO INSURE THAT THE PROGRAM MEMORY CAN BE EXECUTE PROTECTED. | | | MPC10180 |
| | | 1019 | * | | | MPC10190 |
| | | 1020 | * | | | MPC10200 |
| | | 1021 | * DESIGN SPECIFICATION | | | MPC10210 |
| | | 1022 | * MEMORY PROTECT MAPS ARE GENERATED FOR EVERY BLOCK BETWEEN | | | MPC10220 |
| | | 1023 | * BLOCK 1 AND THE LAST BLOCK OF PROGRAM MEMORY PROTECTING ONLY | | | MPC10230 |
| | | 1024 | * THAT BLOCK AT A GIVEN TIME. | | | MPC10240 |
| | | 1025 | * AFTER EACH MAP IS GENERATED THESE BLOCKS ARE TESTED FOR | | | MPC10250 |
| | | 1026 | * PROTECTION BY EXECUTING AN INSTRUCTION AT SPECIFIED LOCATIONS | | | MPC10260 |
| | | 1027 | * IN EACH BLOCK. THESE LOCATIONS CONTAIN A RETURN INSTRUCTION. | | | MPC10270 |
| | | 1028 | * THIS TEST IS PERFORMED WITH THE CONTROLLER IN THE WRITE-READ | | | MPC10280 |
| | | 1029 | * MODE. | | | MPC10290 |
| | | 1030 | * | | | MPC10300 |
| | | 1031 | * | | | MPC10310 |
| | | 1032 | ***** | | | MPC10320 |
| 136A | 41F0 1D96 | 1033 | TEST6 BAL R15,TSTNUM | TEST 6 | | MPC10330 |
| 136E | 48F0 0C40 | 1034 | LH R15,MODEL70 | | | MPC10340 |
| 1372 | 4230 0C72 | 1035 | BNZ QUESTNZ | | | MPC10350 |
| 1376 | 2451 | 1036 | LIS BLK,1 | START WITH BLOCK 1 | | MPC10360 |
| 1378 | 4050 2260 | 1037 | TST6A STH BLK,BLOCK2 | PROTECT THIS BLOCK | | MPC10370 |
| 137C | 24AF | 1038 | LIS EXPT,X'F' | EXECUTE PROTECT FLAG | | MPC10380 |
| 137E | 41F0 1616 | 1039 | BAL R15,PTONEBLK | GENERATE EXECUTE PROTECT MAP | | MPC10390 |
| 1382 | 41F0 1690 | 1040 | BAL R15,LOADMAP | LOAD MEMORY PROTECT MAP | | MPC10400 |
| 1386 | E170 0000 | 1041 | SVC 7,0 | WRITE & READ PROTECT COMMAND | | MPC10410 |
| 138A | 24D0 | 1042 | TST6B LIS R13,0 | ZERO INDEX | | MPC10420 |
| 138C | 2451 | 1043 | LIS BLK,1 | BLOCK 1 FIRST | | MPC10430 |
| 138E | 2470 | 1044 | TST6C LIS INTCW,0 | CLEAR INTERRUPT COUNTERS | | MPC10440 |
| 1390 | 2460 | 1045 | LIS INTCR,0 | | | MPC10450 |
| 1392 | 2480 | 1046 | LIS INTCE,0 | | | MPC10460 |
| 1394 | 48CD 21A0 | 1047 | LH R12,BADRX(R13) | GET ADDRESS IN THIS BLOCK | | MPC10470 |
| 1398 | C8F0 030F | 1048 | LHI R15,X'030F' | "BR R15" INSTRUCTION | | MPC10480 |
| 139C | 40FC 0000 | 1049 | STH R15,0(R12) | WRITE TO THIS ADDRESS | | MPC10490 |
| 13A0 | 41F0 1840 | 1050 | BAL R15,QBLKPT | GET PROTECT STATUS | | MPC10500 |
| 13A4 | 2493 | 1051 | LIS INCQR,3 | | | MPC10510 |
| 13A6 | 48F0 2260 | 1052 | LH R15,BLOCK2 | GET PROTECTED BLOCK | | MPC10520 |
| 13AA | 45F0 219E | 1053 | CLH R15,BLOCKFLG | COMPARE TO BLOCK FLAG | | MPC10530 |
| 13AE | 4280 2058 | 1054 | BL TEST6X | BRANCH TO ADDR > 1E00 | | MPC10540 |
| 13B2 | E140 0000 | 1055 | SVC 4,0 | SET PSW 7 | | MPC10550 |
| 13B6 | 2494 | 1056 | LIS INCQR,4 | | | MPC10560 |
| 13B8 | 01FC | 1057 | BALR R15,R12 | EXECUTE THIS ADDRESS | | MPC10570 |
| 13BA | 2493 | 1058 | LIS INCQR,3 | | | MPC10580 |
| 13BC | E100 0000 | 1059 | TST6D SVC 0,0 | ZERO PSW 7 | | MPC10590 |
| 13C0 | 4140 15D4 | 1060 | TST6E BAL NO,RORW | NO READ OR WRITE VIOLATIONS | | MPC10600 |
| 13C4 | 48F0 226A | 1061 | LH R15,BLKPTE | BLOCK EXECUTE PROTECTED? | | MPC10610 |
| 13C8 | 2336 | 1062 | BZS TST6F | NO | | MPC10620 |
| 13CA | C980 0001 | 1063 | CHI INTCE,1 | ONE EXECUTE VIOLATION | | MPC10630 |
| 13CE | 4230 1F82 | 1064 | BNE ERR6 | ELSE ERROR | | MPC10640 |
| 13D2 | 2303 | 1065 | BS TST6G | | | MPC10650 |

TEST 6

| | | | | | | | | |
|------|------|------|------|-------|------|--------------|-----------------------|----------|
| 13D4 | 4140 | 15CC | 1066 | TST6F | BAL | NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC10660 |
| 13D8 | 4130 | 17DC | 1067 | TST6G | BAL | RET1,NEXTBLK | NEXT BLOCK | MPC10670 |
| 13DC | 4280 | 138E | 1068 | | BL | TST6C | | MPC10680 |
| 13E0 | 4130 | 1782 | 1069 | | BAL | RET1,NEXTPSW | NEXT PSW 8-11 | MPC10690 |
| 13E4 | 2125 | | 1070 | | BTFS | 2,TST6Z | | MPC10700 |
| 13E6 | 41E0 | 179C | 1071 | | BAL | RET2,DISPLAY | DISPLAY NEW PSW | MPC10710 |
| 13EA | 4300 | 138A | 1072 | | B | TST6B | | MPC10720 |
| 13EE | 41E0 | 17EC | 1073 | TST6Z | BAL | RET2,INCLWBL | PROTECT NEXT BLOCK | MPC10730 |
| 13F2 | 4280 | 1378 | 1074 | | BL | TST6A | | MPC10740 |
| 13F6 | 2490 | | 1075 | | LIS | INCQR,0 | | MPC10750 |
| 13F8 | 4300 | 1B02 | 1076 | | B | TSTCHK | NEXT TEST | MPC10760 |

TEST 7

| | | | | | |
|------|-----------|------|---|--------------------------------------|----------|
| | | 1078 | ***** | | MPC10780 |
| | | 1079 | * | | MPC10790 |
| | | 1080 | * | T E S T 7 | MPC10800 |
| | | 1081 | * | | MPC10810 |
| | | 1082 | * PURPOSE | | MPC10820 |
| | | 1083 | * TO INSURE THAT MEMORY CAN BE WRITE, READ AND EXECUTE PROTECTED | | MPC10830 |
| | | 1084 | * WITH PSW 8 BIT SET. | | MPC10840 |
| | | 1085 | * | | MPC10850 |
| | | 1086 | * | | MPC10860 |
| | | 1087 | * DESIGN SPECIFICATION | | MPC10870 |
| | | 1088 | * THE PROGRAM IS MOVED IN MEMORY TO A STARTING LOCATION OF X'8000' | | MPC10880 |
| | | 1089 | * PSW 8 BIT IS THEN SET. | | MPC10890 |
| | | 1090 | * MEMORY PROTECT MAPS ARE AGAIN GENERATED FOR EVERY BLOCK | | MPC10900 |
| | | 1091 | * BETWEEN THE REFERENCE BLOCK AND THE LAST BLOCK OF MEMORY | | MPC10910 |
| | | 1092 | * PROTECTING ONLY THAT BLOCK AT A GIVEN TIME. | | MPC10920 |
| | | 1093 | * AFTER EACH MAP IS GENERATED THE BLOCKS ARE TESTED FOR PROTECTION* | | MPC10930 |
| | | 1094 | * BY WRITING TO THE FIRST AND LAST LOCATION OF EACH BLOCK, READING | | MPC10940 |
| | | 1095 | * FROM THESE LOCATIONS AND EXECUTING THESE LOCATIONS. | | MPC10950 |
| | | 1096 | * THIS TEST IS PERFORMED WITH THE CONTROLLER IN THE WRITE | | MPC10960 |
| | | 1097 | * PROTECT MODE AND IS REPEATED FOR THE WRITE-READ PROTECT MODE. | | MPC10970 |
| | | 1098 | * THIS TEST REQUIRES MINIMUM OF 48 KB OF MEMORY | | MPC10980 |
| | | 1099 | * | | MPC10990 |
| | | 1100 | ***** | | MPC11000 |
| 13FC | 41F0 1D96 | 1101 | TEST7 BAL R15,TSTNUM | TEST 7 | MPC11010 |
| 1400 | 48F0 0C40 | 1102 | LH R15,MODEL70 | | MPC11020 |
| 1404 | 4230 0C72 | 1103 | BNZ QUESTNZ | ?? | MPC11030 |
| 1408 | 41F0 1678 | 1104 | BAL R15,PTNONE | CLEAR MEMORY PROTECT MAP | MPC11040 |
| 140C | 41F0 1690 | 1105 | BAL R15,LOADMAP | LOAD MEMORY PROTECT MAP | MPC11050 |
| | | 1106 | * MOVE PROGRAM TO 8000 | | MPC11060 |
| 1410 | 0722 | 1107 | XHR R2,R2 | SET UP BXLE PARAMETERS | MPC11070 |
| 1412 | 2432 | 1108 | LIS R3,2 | | MPC11080 |
| 1414 | C840 2287 | 1109 | LHI R4,PROGEND | PROGRAM END | MPC11090 |
| 1418 | 4812 0000 | 1110 | MOVEPROG LH R1,0(R2) | GET A LINE OF PROGRAM | MPC11100 |
| 141C | 4012 8000 | 1111 | STH R1,X'8000'(R2) | STORE IT | MPC11110 |
| 1420 | C120 1418 | 1112 | BXLE R2,MOVEPROG | | MPC11120 |
| 1424 | E120 0000 | 1113 | SVC 2,0 | SET PSW 8 | MPC11130 |
| | | 1114 | * NOW AT 8000 + PROGRAM COUNTER | | MPC11140 |
| 1428 | E100 0000 | 1115 | SVC 0,0 | ZERO PSW 7 | MPC11150 |
| 142C | 2421 | 1116 | LIS R2,1 | WRITE-READ COMMAND | MPC11160 |
| 142E | 41F0 1616 | 1117 | TST7A BAL R15,PTONEBLK | GENERATE WRITE-READ PROTECT MAP | MPC11170 |
| 1432 | 24AF | 1118 | LIS EXPT,X'F' | EXECUTE PROTECT FLAG | MPC11180 |
| 1434 | 41F0 1620 | 1119 | BAL R15,PTALLBLK | GENERATE EXECUTE PROTECT FLAG | MPC11190 |
| 1438 | 41F0 1690 | 1120 | BAL R15,LOADMAP | LOAD MEMORY PROTECT MAP | MPC11200 |
| 143C | E100 0000 | 1121 | SVC 0,0 | ZERO PSW 7 | MPC11210 |
| 1440 | E170 0000 | 1122 | SVC 7,0 | WRITE & READ PROTECT COMMAND | MPC11220 |
| 1444 | 4850 219C | 1123 | IST7B LH BLK,BLOCKREF | REFERENCE BLOCK FIRST | MPC11230 |
| 1448 | C8F0 1486 | 1124 | LHI R15,TST7E | EXIT ADDRESS | MPC11240 |
| 144C | 40F0 2278 | 1125 | STH R15,TESTEND | | MPC11250 |
| 1450 | 2751 | 1126 | SIS BLK,1 | INITIALIZE BLOCK VALUE | MPC11260 |
| 1452 | 41F0 16F4 | 1127 | TST7C BAL R15,BLKADR | FIRST & LAST ADDRESS OF NEXT BLOCK | MPC11270 |
| 1456 | 41F0 17FE | 1128 | BAL R15,PSW8X | MODIFY BLOCK VALUE WITH PSW 8-11 = 8 | MPC11280 |
| 145A | C8F0 030F | 1129 | LHI R15,X'030F' | "BR R15" INSTRUCTION | MPC11290 |
| 145E | 40FC 0000 | 1130 | STH R15,0(R12) | WRITE TO FIRST ADDRESS | MPC11300 |

TEST 7

| | | | | | | | |
|------|------|------|------|------|------------|--------------------------------|----------|
| 1452 | 40FE | 0000 | 1131 | STH | R15,0(R14) | WRITE TO LAST ADDRESS | MPC11310 |
| 1466 | E140 | 0000 | 1132 | SVC | 4,0 | SET PSW 7 | MPC11320 |
| 146A | 40FC | 0000 | 1133 | STH | R15,0(R12) | WRITE TO FIRST ADDRESS | MPC11330 |
| 146E | 40FE | 0000 | 1134 | STH | R15,0(R14) | WRITE TO LAST ADDRESS | MPC11340 |
| 1472 | 48FC | 0000 | 1135 | LH | R15,0(R12) | READ FROM FIRST ADDRESS | MPC11350 |
| 1476 | 48FE | 0000 | 1136 | LH | R15,0(R14) | READ FROM LAST ADDRESS | MPC11360 |
| 147A | 2494 | | 1137 | LIS | INCQR,4 | | MPC11370 |
| 147C | 01FC | | 1138 | BALR | R15,R12 | EXECUTE FIRST ADDRESS | MPC11380 |
| 147E | 01FE | | 1139 | BALR | R15,R14 | EXECUTE LAST ADDRESS | MPC11390 |
| 1480 | 2493 | | 1140 | LIS | INCQR,3 | | MPC11400 |
| 1482 | E100 | 0000 | 1141 | SVC | 0,0 | ZERO PSW 7 | MPC11410 |
| 1486 | 2490 | | 1142 | LIS | INCQR,0 | | MPC11420 |
| 1488 | 48A0 | 2268 | 1143 | LH | R10,BLKPT | BLOCK WRITE OR READ PROTECTED? | MPC11430 |
| 148C | 2137 | | 1144 | BNZS | TST7D | YES | MPC11440 |
| 148E | 4140 | 15D4 | 1145 | BAL | NO,RORW | NO READ OR WRITE VIOLATIONS | MPC11450 |
| 1492 | 4140 | 15CC | 1146 | BAL | NO,EXECUTE | NO EXECUTE VIOLATIONS | MPC11460 |
| 1496 | 4300 | 1452 | 1147 | B | TST7C | NEXT BLOCK | MPC11470 |
| 149A | C970 | 0002 | 1148 | CHI | INTCW,2 | TWO WRITE VIOLATIONS | MPC11480 |
| 149E | 4230 | 1F72 | 1149 | BNE | ERR4 | ELSE ERROR | MPC11490 |
| 14A2 | C960 | 0002 | 1150 | CHI | INTCR,2 | TWO READ VIOLATIONS | MPC11500 |
| 14A6 | 4230 | 1F7A | 1151 | BNE | ERR5 | ELSE ERROR | MPC11510 |
| 14AA | C980 | 0002 | 1152 | CHI | INTCE,2 | TWO EXECUTE VIOLATIONS | MPC11520 |
| 14AE | 4230 | 1F82 | 1153 | BNE | ERR6 | ELSE ERROR | MPC11530 |
| 14B2 | 4300 | 1452 | 1154 | B | TST7C | NEXT BLOCK | MPC11540 |
| 14B6 | 41F0 | 17CA | 1155 | BAL | R15,INCBLK | PROTECT NEXT BLOCK | MPC11550 |
| 14BA | 4320 | 142E | 1156 | BFC | 2,TST7A | MORE BLOCKS? | MPC11560 |
| 14BE | E130 | 0000 | 1157 | SVC | 3,0 | NO, ZERO PSW 8 | MPC11570 |
| 14C2 | 4300 | 1B02 | 1158 | B | TSTCHK | AND EXIT TEST | MPC11580 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|---------|----------------|-------------------------------------|----------|
| | | 1160 | ***** | | | HPC11600 |
| | | 1161 | * | | | HPC11610 |
| | | 1162 | * | SVC TABLE | | HPC11620 |
| | | 1163 | * | | | HPC11630 |
| | | 1164 | * | | | HPC11640 |
| | | 1165 | * | SVC 0 | PROTECT MODE OFF PSW7 = 0 | HPC11650 |
| | | 1166 | * | | | HPC11660 |
| 14C6 | 4800 0096 | 1167 | NOPROT | LH RO,X'96' | GET OLD PSW 0-15 | HPC11670 |
| 14CA | C400 FEFF | 1168 | | NHI RO,X'FEFF' | CLEAR PSW 7 | HPC11680 |
| 14CE | 4000 0096 | 1169 | | STH RO,X'96' | STORE IT | HPC11690 |
| 14D2 | C200 0096 | 1170 | | LPSW X'96' | NEW PSW | HPC11700 |
| | | 1171 | * | | | HPC11710 |
| | | 1172 | * | SVC 1 | DISABLE CONTROLLER & PSW7 = 0 | HPC11720 |
| | | 1173 | * | | | HPC11730 |
| 14D6 | C800 2000 | 1174 | DISPROT | LHI RO,X'2000' | NEW PSW X'2000' | HPC11740 |
| 14DA | 4000 0096 | 1175 | | STH RO,X'96' | STORE IT | HPC11750 |
| 14DE | 4800 0C30 | 1176 | | LH RO,PRTADR | CONTROLLER ADDRESS | HPC11760 |
| 14E2 | DE00 0A37 | 1177 | | OC RO,POFF | PROTECT OFF CONHAND | HPC11770 |
| 14E6 | C200 0096 | 1178 | | LPSW X'96' | USE NEW PSW | HPC11780 |
| | | 1179 | * | | | HPC11790 |
| | | 1180 | * | SVC 2 | SET PSW 8 | HPC11800 |
| | | 1181 | * | | | HPC11810 |
| 14EA | 4800 0096 | 1182 | SETS | LH RO,X'96' | GET OLD PSW 0-15 | HPC11820 |
| 14EE | C600 0080 | 1183 | | OHI RO,X'0080' | SET BIT 8 | HPC11830 |
| 14F2 | 4000 0096 | 1184 | | STH RO,X'96' | RESTORE PSW | HPC11840 |
| 14F6 | C200 0096 | 1185 | | LPSW X'96' | USE MODIFIED PSW | HPC11850 |
| | | 1186 | * | | | HPC11860 |
| | | 1187 | * | SVC 3 | CLEAR PSW 8 | HPC11870 |
| | | 1188 | * | | | HPC11880 |
| 14FA | 4800 0096 | 1189 | CLEAR8 | LH RO,X'96' | GET OLD PSW 0-15 | HPC11890 |
| 14FE | C400 FF7F | 1190 | | NHI RO,X'FF7F' | CLEAR BIT 8 | HPC11900 |
| 1502 | 4000 0096 | 1191 | | STH RO,X'96' | RESTORE PSW | HPC11910 |
| 1506 | C200 0096 | 1192 | | LPSW X'96' | USE MODIFIED PSW | HPC11920 |
| | | 1193 | * | | | HPC11930 |
| | | 1194 | * | SVC 4 | PROTECT MODE ON SET PSW 7 | HPC11940 |
| | | 1195 | * | | | HPC11950 |
| 150A | 4800 0096 | 1196 | PROT | LH RO,X'96' | GET OLD PSW 0-15 | HPC11960 |
| 150E | C600 0100 | 1197 | | OHI RO,X'0100' | SET BIT 7 PROTECT | HPC11970 |
| 1512 | 4000 0096 | 1198 | | STH RO,X'96' | RESTORE PSW | HPC11980 |
| 1516 | C200 0096 | 1199 | | LPSW X'96' | | HPC11990 |
| | | 1200 | * | | | HPC12000 |
| | | 1201 | * | SVC 5 | WRITE PROTECT COMMAND TO CONTROLLER | HPC12010 |
| | | 1202 | * | | | HPC12020 |
| 151A | 4800 0C30 | 1203 | ONPROT | LH RO,PRTADR | GET CONTROL ADDRESS | HPC12030 |
| 151E | DE00 0A35 | 1204 | | OC RO,PON | COMMAND FOR ON PROTECTION | HPC12040 |
| 1522 | 9D0F | 1205 | | SSR RO,R15 | SENSE STATUS FOR ON PROTECTION | HPC12050 |
| 1524 | C3F0 0020 | 1206 | | THI R15,X'20' | PROTECTION ON? | HPC12060 |
| 1528 | 4330 1F92 | 1207 | | BZ ERR8 | NO, ERROR | HPC12070 |
| 152C | C200 0096 | 1208 | | LPSW X'96' | RETURN | HPC12080 |
| | | 1209 | * | | | HPC12090 |
| | | 1210 | * | SVC 6 | OFF PROTECT COMMAND TO CONTROLLER | HPC12100 |
| | | 1211 | * | | | HPC12110 |
| 1530 | 4800 0C30 | 1212 | OFFPROT | LH RO,PRTADR | GET CONTROL ADDRESS | HPC12120 |

SUBROUTINES

| | | | | | | | |
|------|------|------|------|--------|--|---------------------------------------|----------|
| 1534 | DE00 | 0A37 | 1213 | OC | RO,POFF | COMMAND FOR OFF PROTECTION | NPC12130 |
| 1538 | 9D0F | | 1214 | SSR | RO,R15 | SENSE STATUS FOR OFF PROTECT | NPC12140 |
| 153A | 08FF | | 1215 | LHR | R15,R15 | | NPC12150 |
| 153C | 4230 | 1F8A | 1216 | BNZ | ERR7 | STATUS NONE ZERO, ERROR | NPC12160 |
| 1540 | C200 | 0096 | 1217 | LPSW | X'96' | RETURN | NPC12170 |
| | | | 1218 | * | | | NPC12180 |
| | | | 1219 | * | | | NPC12190 |
| | | | 1220 | * | SVC 7 | WRITE & READ PROTECT COMMAND TO CONTR | NPC12200 |
| 1544 | 4800 | 0C30 | 1221 | WRPROT | LH RO,PRADR | GET PROTECT DEVICE ADDRESS | NPC12210 |
| 1548 | DE00 | 0A36 | 1222 | GC | RO,WTRD | WRITE & READ PROTECT | NPC12220 |
| 154C | 9D0F | | 1223 | SSR | RO,R15 | SENSE STATUS | NPC12230 |
| 154E | C3F0 | 0020 | 1224 | THI | R15,X'20' | EXPECT ON PROTECT | NPC12240 |
| 1552 | 4330 | 1F92 | 1225 | BZ | ERR8 | ELSE ERROR | NPC12250 |
| 1556 | C200 | 0096 | 1226 | LPSW | X'96' | RETURN | NPC12260 |
| | | | 1227 | * | | | NPC12270 |
| | | | 1228 | * | SVC 8 | SENSE STATUS WITH PSW7 SET | NPC12280 |
| | | | 1229 | * | | | NPC12290 |
| 155A | 4800 | 0C30 | 1230 | SENSEZ | LH RO,PRADR | | NPC12300 |
| 155E | 9D0F | | 1231 | SSR | RO,R15 | | NPC12310 |
| 1560 | C200 | 0096 | 1232 | LPSW | X'96' | | NPC12320 |
| 1564 | | | 1233 | BADR11 | DS 2 | | NPC12330 |
| | | | 1234 | * | ***** | | NPC12340 |
| | | | 1235 | * | | | NPC12350 |
| | | | 1236 | * | INTPRT INTERRUPT ROUTINE FOR THE MEMORY PROTECT CONTROLLER | | NPC12360 |
| | | | 1237 | * | | | NPC12370 |
| | | | 1238 | * | EXTERNAL I/O INTERRUPTS CREATED BY WRITE OR | | NPC12380 |
| | | | 1239 | * | READ MEMORY PROTECT VIOLATIONS ARE SERVICED IN THIS | | NPC12390 |
| | | | 1240 | * | ROUTINE | | NPC12400 |
| | | | 1241 | * | THE STATUS OF THE CONTROLLER IS SENSED, THE | | NPC12410 |
| | | | 1242 | * | INTERRUPT ACKNOWLEDGED AND THE WRITE OR READ INTERRUPT | | NPC12420 |
| | | | 1243 | * | COUNTER IS INCREMENTED. | | NPC12430 |
| | | | 1244 | * | | | NPC12440 |
| | | | 1245 | * | REGISTERS R10-R15 ARE SAVED | | NPC12450 |
| | | | 1246 | * | REGISTERS INTCR OR INTCW ARE UPDATED | | NPC12460 |
| | | | 1247 | * | | | NPC12470 |
| | | | 1248 | * | | | NPC12480 |
| 1566 | DOA0 | 223C | 1249 | INTPRT | STH R10,SSAVE | SAVE REGISTERS | NPC12490 |
| 156A | 4800 | 0C30 | 1250 | LH | RO,PRADR | GET PROTECT DEVICE ADR | NPC12500 |
| 156E | 9D0F | | 1251 | SSR | RO,R15 | SENSE STATUS | NPC12510 |
| 1570 | 9FAE | | 1252 | AIR | R10,R14 | ACKNOWLEDGE INTERRUPT | NPC12520 |
| 1572 | 050A | | 1253 | CLHR | RO,R10 | IS THIS THE MEMORY PROTECT? | NPC12530 |
| 1574 | 4230 | 1FBE | 1254 | BNE | ERR13 | NO, ERROR | NPC12540 |
| 1578 | C5F0 | 0064 | 1255 | CLHI | R15,X'64' | READ VIOLATION? | NPC12550 |
| 157C | 233B | | 1256 | BES | INT0 | YES | NPC12560 |
| 157E | C5F0 | 0034 | 1257 | CLHI | R15,X'34' | THEN WRITE VIOLATION | NPC12570 |
| 1582 | 4230 | 1F9A | 1258 | BNE | ERR9 | ELSE ERROR | NPC12580 |
| 1586 | 2671 | | 1259 | AIS | INTCW,1 | INCREMENT WRITE COUNTER | NPC12590 |
| 1588 | C570 | 0010 | 1260 | CLHI | INTCW,16 | < 16 INTERRUPTS ? | NPC12600 |
| 158C | 4380 | 2004 | 1261 | BWL | ERR20 | NO, ERROR | NPC12610 |
| 1590 | 2306 | | 1262 | BS | INTZ | CONTINUE | NPC12620 |
| 1592 | 2661 | | 1263 | INT0 | AIS INTCR,1 | INCREMENT READ COUNTER | NPC12630 |
| 1594 | C560 | 0010 | 1264 | CLHI | INTCR,16 | < 16 INTERRUPTS ? | NPC12640 |
| 1598 | 4380 | 2004 | 1265 | BWL | ERR20 | NO, ERROR | NPC12650 |

SUBROUTINES

| | | | | | | | | |
|------|------|------|------|----------|------|-----------------------|-------------------------------|----------|
| 159C | 48F0 | 0040 | 1266 | INTZ | LH | R15,X'40' | GET OLD PSW 0-15 | NPC12660 |
| 15A0 | C4F0 | 00F0 | 1267 | | NHI | R15,X'00F0' | KEEP PSW 8-11 | NPC12670 |
| 15A4 | 950F | | 1268 | | EPSR | R0,R15 | NEW PSW 0-15 | NPC12680 |
| 15A6 | 48F0 | 2268 | 1269 | | LH | R15,BLKPT | GET PROTECT STATUS OF BLOCK | NPC12690 |
| 15AA | 24A0 | | 1270 | | LIS | R10,0 | ZERO | NPC12700 |
| 15AC | 950A | | 1271 | | EPSR | R0,R10 | NEW PSW 0-15 | NPC12710 |
| 15AE | 08FF | | 1272 | | LHR | R15,R15 | CHECK PROTECT STATUS OF BLOCK | NPC12720 |
| 15B0 | 4330 | 1FF4 | 1273 | | BZ | ERR18 | ERROR IF NOT PROTECTED | NPC12730 |
| 15B4 | D1A0 | 223C | 1274 | INTZZ | LH | R10,SSAVE | RESTORE REGISTERS | NPC12740 |
| 15B8 | C200 | 0040 | 1275 | | LPSW | X'40' | RETURN | NPC12750 |
| | | | 1276 | | | | | NPC12760 |
| | | | 1277 | * | | | | NPC12770 |
| | | | 1278 | * | | | | NPC12780 |
| | | | 1279 | * | | | | NPC12790 |
| | | | 1280 | WRT | LHR | INTCW,INTCW | NO WRITE VIOLATIONS | NPC12800 |
| 15BC | 0877 | | 1281 | | BNZ | ERR1 | ELSE ERROR | NPC12810 |
| 15BE | 4230 | 1F5A | 1282 | | BR | NO | | NPC12820 |
| 15C2 | 0304 | | 1283 | * | | | | NPC12830 |
| 15C4 | 0866 | | 1284 | READ | LHR | INTCR,INTCR | NO READ VIOLATIONS | NPC12840 |
| 15C6 | 4230 | 1F62 | 1285 | | BNZ | ERR2 | ELSE ERROR | NPC12850 |
| 15CA | 0304 | | 1286 | | BR | NO | | NPC12860 |
| | | | 1287 | * | | | | NPC12870 |
| 15CC | 0888 | | 1288 | EXECUTE | LHR | INTCE,INTCE | NO EXECUTE VIOLATIONS | NPC12880 |
| 15CE | 4230 | 1F6A | 1289 | | BNZ | ERR3 | ELSE ERROR | NPC12890 |
| 15D2 | 0304 | | 1290 | | BR | NO | | NPC12900 |
| | | | 1291 | * | | | | NPC12910 |
| 15D4 | 0877 | | 1292 | RORW | LHR | INTCW,INTCW | NO WRITE VIOLATIONS | NPC12920 |
| 15D6 | 4230 | 1F5A | 1293 | | BNZ | ERR1 | ELSE ERROR | NPC12930 |
| 15DA | 0866 | | 1294 | | LHR | INTCR,INTCR | NO READ VIOLATIONS EITHER | NPC12940 |
| 15DC | 4230 | 1F62 | 1295 | | BNZ | ERR2 | ELSE ERROR | NPC12950 |
| 15E0 | 0304 | | 1296 | | BR | NO | | NPC12960 |
| | | | 1297 | * | | | | NPC12970 |
| | | | 1298 | * | | | | NPC12980 |
| | | | 1299 | *STATERR | | SENSE AND TEST STATUS | | NPC12990 |
| | | | 1300 | * | | | | NPC13000 |
| | | | 1301 | * | | | | NPC13010 |
| 15E2 | E180 | 0000 | 1302 | STATERR | SVC | 8,0 | SENSE STATUS | NPC13020 |
| 15E6 | 48A0 | 2268 | 1303 | | LH | R10,BLKPT | BLOCK PROTECTED? | NPC13030 |
| 15EA | 2136 | | 1304 | | BNZS | SE2 | YES CHECK FOR VIOL STATUS | NPC13040 |
| 15EC | C3F0 | 0020 | 1305 | SE1 | THI | R15,X'20' | PON BIT SET | NPC13050 |
| 15F0 | 4330 | 203C | 1306 | | BZ | ERR26 | ELSE ERROR | NPC13060 |
| 15F4 | 0304 | | 1307 | | BR | NO | | NPC13070 |
| 15F6 | 48A0 | 0C40 | 1308 | SE2 | LH | R10,MODEL70 | MODEL 70? | NPC13080 |
| 15FA | 2139 | | 1309 | | BNZS | SE4 | YES TEST FOR WRITE VIOL. | NPC13090 |
| 15FC | 05FE | | 1310 | SE3 | CLHR | R15,R14 | TEST FOR STATUS | NPC13100 |
| 15FE | 0334 | | 1311 | | BER | NO | | NPC13110 |
| 1600 | C3E0 | 0080 | 1312 | | THI | R14,X'80' | EX VIOL EXPECTED? | NPC13120 |
| 1604 | 4330 | 1F9A | 1313 | | BZ | ERR9 | NO, ERROR 9 | NPC13130 |
| 1608 | 4300 | 1FAE | 1314 | | B | ERR11 | ELSE ERROR 11 | NPC13140 |
| 160C | C3E0 | 0010 | 1315 | SE4 | THI | R14,X'10' | EXPECT WRITE VIOLATION ? | NPC13150 |
| 1610 | 203A | | 1316 | | BNZS | SE3 | YES | NPC13160 |
| 1612 | 4300 | 15EC | 1317 | | B | SE1 | NO | NPC13170 |
| | | | 1318 | * | | | | NPC13180 |

SUBROUTINES

| | | | | | | | | | |
|------|------|------|------|-----------|--|---------------------------|--|--|----------|
| | | | 1319 | * | | | | | NPC13190 |
| | | | 1320 | *PTONEBLK | GENERATES A MEMORY PROTECT MAP WITH ONLY THE BLOCK | | | | NPC13200 |
| | | | 1321 | * | SPECIFIED IN REGISTER "BLK" PROTECTED. | | | | NPC13210 |
| | | | 1322 | * | | | | | NPC13220 |
| | | | 1323 | *PTALLBLK | GENERATES A MEMORY PROTECT MAP PROTECTING PREVIOUSLY | | | | NPC13230 |
| | | | 1324 | * | PROTECTED BLOCKS IN ADDITION TO THE BLOCK IN REGISTER | | | | NPC13240 |
| | | | 1325 | * | "BLK". | | | | NPC13250 |
| | | | 1326 | *PTNONE | CLEAR THE MEMORY PROTECT MAP, NO BLOCKS PROTECTED. | | | | NPC13260 |
| | | | 1327 | * | | | | | NPC13270 |
| | | | 1328 | * | | | | | NPC13280 |
| | | | 1329 | * | REGISTER "EXPT" DETERMINES THE TYPE OF PROTECTON | | | | NPC13290 |
| | | | 1330 | * | THAT THE BLOCK IS TO HAVE: WRITE-READ PROTECTION,(EXPT | | | | NPC13300 |
| | | | 1331 | * | = 0) OR EXECUTE PROTECTION (EXPT = F). | | | | NPC13310 |
| | | | 1332 | * | | | | | NPC13320 |
| | | | 1333 | * | REGISTERS R11-15 ARE SAVED. | | | | NPC13330 |
| | | | 1334 | * | | | | | NPC13340 |
| | | | 1335 | * | | | | | NPC13350 |
| 1616 | DOBO | 220C | 1336 | PTONEBLK | STM R11,SAVEA | SAVE REGISTERS | | | NPC13360 |
| 161A | 41E0 | 1678 | 1337 | | BAL R15,PTNONE | CLEAR MEMORY MAP | | | NPC13370 |
| 161E | 2303 | | 1338 | | BS PBL2 | CONTINUE | | | NPC13380 |
| 1620 | DOBO | 220C | 1339 | PTALLBLK | STM R11,SAVEA | SAVE REGISTERS | | | NPC13390 |
| 1624 | C9A0 | 000F | 1340 | PBL2 | CHI EXPT,X'F' | EXECUTE PROTECT FLAG SET? | | | NPC13400 |
| 1628 | 2133 | | 1341 | | BNES PBL20 | NO | | | NPC13410 |
| 162A | CA50 | 0040 | 1342 | | AHI BLK,64 | YES, ADD 64 | | | NPC13420 |
| 162E | 08F5 | | 1343 | PBL20 | LHR R15,BLK | GET BLOCK NUMBER | | | NPC13430 |
| 1630 | 4330 | 1668 | 1344 | | BZ PBLZ | EXIT ON ZERO | | | NPC13440 |
| 1634 | 07DD | | 1345 | | MHR R13,R13 | ZERO | | | NPC13450 |
| 1636 | 24E9 | | 1346 | | LIS R14,9 | SET COUNT TO 9 | | | NPC13460 |
| 1638 | 09FE | | 1347 | PBL3 | CHR R15,R14 | BLOCK LESS THAN COUNT? | | | NPC13470 |
| 163A | 2184 | | 1348 | | BLS PBL4 | YES CONTINUE | | | NPC13480 |
| 163C | 26D2 | | 1349 | | AIS R13,2 | INC BY 2 | | | NPC13490 |
| 163E | 26E8 | | 1350 | | AIS R14,8 | INC BY 8 | | | NPC13500 |
| 1640 | 2204 | | 1351 | | BS PBL3 | CHECK NEXT LOCATION | | | NPC13510 |
| 1642 | C950 | 0041 | 1352 | PBL4 | CHI BLK,X'41' | COMPARE TO 65 | | | NPC13520 |
| 1646 | 2186 | | 1353 | | BLS PBL5 | LESS USE THESE VALUES | | | NPC13530 |
| 1648 | CB00 | 0010 | 1354 | | SHI R13,X'10' | ALTER VALUES | | | NPC13540 |
| 164C | C8C0 | 0100 | 1355 | | LHI R12,X'0100' | | | | NPC13550 |
| 1650 | 2302 | | 1356 | | BS PBL6 | | | | NPC13560 |
| 1652 | 24C1 | | 1357 | PBL5 | LIS R12,1 | SET BIT 15 | | | NPC13570 |
| 1654 | 0BEF | | 1358 | PBL6 | SHR R14,R15 | CALCULATE BIT POSTION | | | NPC13580 |
| 1656 | 27E1 | | 1359 | PBL7 | SIS R14,1 | DECRMENT | | | NPC13590 |
| 1658 | 2333 | | 1360 | | BZS PBL8 | CONTINUE | | | NPC13600 |
| 165A | 91C1 | | 1361 | | SLLS R12,1 | ELSE TRY NEXT BIT | | | NPC13610 |
| 165C | 2203 | | 1362 | | BS PBL7 | | | | NPC13620 |
| 165E | 48BD | 21DC | 1363 | PBL8 | LH R11,MPHF1(R13) | GET BLOCK PROTECT | | | NPC13630 |
| 1662 | 06BC | | 1364 | | OHR R11,R12 | SET THIS BIT | | | NPC13640 |
| 1664 | 40BD | 21DC | 1365 | | STH R11,MPHF1(R13) | RESTORE BLOCK PROTECT | | | NPC13650 |
| 1668 | C9A0 | 000F | 1366 | PBLZ | CHI EXPT,X'F' | EXECUTE PROTECT FLAG SET? | | | NPC13660 |
| 166C | 2133 | | 1367 | | BNES PBLZ0 | NO | | | NPC13670 |
| 166E | CB50 | 0040 | 1368 | | SHI BLK,64 | YES, RESTORE VALUE | | | NPC13680 |
| 1672 | D1B0 | 220C | 1369 | PBLZ0 | LH R11,SAVEA | RESTOR REGISTERS | | | NPC13690 |
| 1676 | 030F | | 1370 | | BR R15 | RETURN | | | NPC13700 |
| | | | 1371 | * | | | | | NPC13710 |

SUBROUTINES

| | | | | | | | | | |
|------|------|------|------|----------|------|---|----------------------------|--|----------|
| | | | 1372 | * | | | | | MPC13720 |
| | | | 1373 | * | | | | | MPC13730 |
| 1678 | D0E0 | 2216 | 1374 | PTNONE | STM | R14,SAVEC | SAVE REGISTERS | | MPC13740 |
| 167C | 24F0 | | 1375 | | LIS | R15,0 | ZERO | | MPC13750 |
| 167E | C8E0 | 0010 | 1376 | | LHI | R14,X'10' | SET COUNT AT 16 | | MPC13760 |
| 1682 | 27E2 | | 1377 | PTN1 | SIS | R14,2 | DECREMENT BY 2 | | MPC13770 |
| 1684 | 40FE | 21DC | 1378 | | STH | R15,MPHF1(R14) | GENERATE MEMORY PROTECT HW | | MPC13780 |
| 1688 | 2033 | | 1379 | | BNZS | PTN1 | 8 HW | | MPC13790 |
| 168A | D1E0 | 2216 | 1380 | | LM | R14,SAVEC | RESTORE REGISTERS | | MPC13800 |
| 168E | 030F | | 1381 | | BR | R15 | RETURN | | MPC13810 |
| | | | 1382 | ***** | | | | | MPC13820 |
| | | | 1383 | * | | | | | MPC13830 |
| | | | 1384 | *LOADMAP | | LOADS THE MEMORY PROTECT MAP TO THE MEMORY PROTECT | | | MPC13840 |
| | | | 1385 | * | | CONTROLLER. | | | MPC13850 |
| | | | 1386 | * | | | | | MPC13860 |
| | | | 1387 | * | | SENDS EIGHT "WRITE HALFWORDS" COMMANDS (THE CONTENTS | | | MPC13870 |
| | | | 1388 | * | | OF MEMORY LOCATIONS MPHF1-8) TO THE CONTROLLER OR EIGHT | | | MPC13880 |
| | | | 1389 | * | | "WRITE BYTES" FOR THE MODEL 70 OPTION. THE ROUTINE ALSO | | | MPC13890 |
| | | | 1390 | * | | DISPLAYS THE NUMBER OF THE FIRST PROTECTED BLOCK, PSW 0-15, | | | MPC13900 |
| | | | 1391 | * | | AND THE MODE OF THE CONTROLLER (WRITE PROTECT OR | | | MPC13910 |
| | | | 1392 | * | | WRITE-READ PROTECT). IT WILL ALSO TERMINATE TESTING IF THE | | | MPC13920 |
| | | | 1393 | * | | BREAK KEY IS DEPRESSED. | | | MPC13930 |
| | | | 1394 | * | | | | | MPC13940 |
| | | | 1395 | * | | REGISTERS R13-15 ARE SAVED. | | | MPC13950 |
| | | | 1396 | * | | | | | MPC13960 |
| | | | 1397 | * | | | | | MPC13970 |
| 1690 | D0D0 | 221C | 1398 | LOADMAP | STM | R13,SAVEE | SAVE REGISTERS | | MPC13980 |
| 1694 | 24A0 | | 1399 | | LIS | EXPT,0 | NO EXECUTE PROTECT | | MPC13990 |
| 1696 | D3F0 | 0A1C | 1400 | | LB | R15,ADDRESS | CONSOLE ADDRESS | | MPC14000 |
| 169A | DEF0 | 0A30 | 1401 | | OC | R15,RDCMD | READ MODE | | MPC14010 |
| 169E | 41F0 | 1E2C | 1402 | | BAL | R15,TSTBRK | CHECK FOR BREAK | | MPC14020 |
| 16A2 | 95DD | | 1403 | | EPSR | R13,R13 | ELSE GET PSW 0-15 | | MPC14030 |
| 16A4 | C4D0 | FFFO | 1404 | | NHI | R13,X'FFFO' | ONLY PSW 0-11 | | MPC14040 |
| 16A8 | 24F1 | | 1405 | | LIS | R15,1 | DISPLAY PANEL | | MPC14050 |
| 16AA | DEF0 | 0A34 | 1406 | | OC | R15,INCRMT | INC MODE | | MPC14060 |
| 16AE | 48E0 | 2260 | 1407 | | LH | R14,BLOCK2 | PROTECTED BLOCK NUMBER | | MPC14070 |
| 16B2 | 0822 | | 1408 | | LHR | R2,R2 | WRITE PROTECT MODE? | | MPC14080 |
| 16B4 | 2333 | | 1409 | | BZS | LM1 | YES | | MPC14090 |
| 16B6 | C6E0 | F000 | 1410 | | OHI | R14,X'F000' | ELSE SET WRITE-READ FLAG | | MPC14100 |
| 16BA | 40E0 | 2266 | 1411 | LM1 | STH | R14,BLKPROT | STORE RESULTS | | MPC14110 |
| 16BE | 94FE | | 1412 | | EXBR | R14,R14 | SET BYTES FOR DISPLAY | | MPC14120 |
| 16C0 | 98FE | | 1413 | | WHR | R15,R14 | DISPLAY | | MPC14130 |
| 16C2 | 94DD | | 1414 | | EXBR | R13,R13 | SET BYTES FOR DISPLAY | | MPC14140 |
| 16C4 | 98FD | | 1415 | | WHR | R15,R13 | DISPLAY | | MPC14150 |
| 16C6 | 24D0 | | 1416 | | LIS | R13,0 | ZERO | | MPC14160 |
| 16C8 | 07DD | | 1417 | | XHR | R13,R13 | ZERO COUNTER | | MPC14170 |
| 16CA | 48FC | 0C30 | 1418 | | LH | R15,PRTADR | GET MEMOERY PROTECT ADR | | MPC14180 |
| 16CE | 48E0 | 0C40 | 1419 | | LH | R14,MODEL70 | MODEL 70? | | MPC14190 |
| 16D2 | 213A | | 1420 | | BNZS | LM3 | YES USE BYTE | | MPC14200 |
| 16D4 | D8FD | 21DC | 1421 | LM2 | WH | R15,MPHF1(R13) | WRITE THIS HW TO MEM PROT | | MPC14210 |
| 16D8 | 26D2 | | 1422 | | AIS | R13,2 | INC BY 2 | | MPC14220 |
| 16DA | C5D0 | 0010 | 1423 | | CLHI | R13,X'10' | 16? | | MPC14230 |
| 16DE | 2085 | | 1424 | | BLS | LM2 | IF LESS WRITE ANOTHER HW | | MPC14240 |

SUBROUTINES

| | | | | | | | | |
|------|------|------|------|---------|-------|--|----------------------------|----------|
| 16E0 | D1D0 | 221C | 1425 | LMZ | LM | R13,SAVEE | RESTORE REGISTERS | MPC14250 |
| 16E4 | 030F | | 1426 | | BR | R15 | RETURN | MPC14260 |
| 16E6 | DAFD | 21DD | 1427 | LM3 | WD | R15,MPHF1+1(R13) | WRITE BYTE TO MEM PROT | MPC14270 |
| 16EA | 26E2 | | 1428 | | AIS | R13,2 | INC BY 2 | MPC14280 |
| 16EC | C5D0 | 0010 | 1429 | | CLHI | R13,X'10' | 16? | MPC14290 |
| 16F0 | 2085 | | 1430 | | BLS | LM3 | IF LESS WRITE ANOTHER BYTE | MPC14300 |
| 16F2 | 2209 | | 1431 | | BS | LMZ | EXIT | MPC14310 |
| | | | 1432 | | ***** | | | MPC14320 |
| | | | 1433 | * | | | | MPC14330 |
| | | | 1434 | *BALADR | | CALCULATES THE FIRST AND LAST ADDRESS OF THE NEXT | | MPC14340 |
| | | | 1435 | * | | BLOCK | | MPC14350 |
| | | | 1436 | * | | | | MPC14360 |
| | | | 1437 | * | | THE VALUE OF REGISTER "BLK" IS INCREMENTED AND THE | | MPC14370 |
| | | | 1438 | * | | FIRST AND LAST ADDRESS OF THIS BLOCK ARE CALCULATED AND | | MPC14380 |
| | | | 1439 | * | | STORED IN R12 AND R14 RESPECTIVELY. AFTER THE LAST BLOCK | | MPC14390 |
| | | | 1440 | * | | OF MEMORY IS REACHED, PSW 8-11 IS INCREMENTED, AND WHEN | | MPC14400 |
| | | | 1441 | * | | THE TOP OF MEMORY IS REACHED THE ROUTINE BRANCHES TO THE | | MPC14410 |
| | | | 1442 | * | | ADDRESS CONTAINED IN "TESTEND". THIS ROUTINE ALSO BRANCHES | | MPC14420 |
| | | | 1443 | * | | TO SUBROUTINES "QBLKPT" AND "DISPLAYS" | | MPC14430 |
| | | | 1444 | * | | | | MPC14440 |
| | | | 1445 | * | | REGISTER R15 IS SAVED | | MPC14450 |
| | | | 1446 | * | | REGISTERS R6-8 ARE CLEARED INTERRUPT COUNTERS | | MPC14460 |
| | | | 1447 | * | | REGISTERS R12 & R14 CONTAIN THE BLOCK ADDRESS ON EXIT. | | MPC14470 |
| | | | 1448 | * | | REGISTER R13 PREVIOUS CONTENTS ARE DESTROYED. | | MPC14480 |
| | | | 1449 | * | | | | MPC14490 |
| | | | 1450 | * | | | | MPC14500 |
| 16F4 | 40F0 | 2222 | 1451 | BLKADR | STH | R15,SAVEG | SAVE REGISTERS | MPC14510 |
| 16F8 | 2460 | | 1452 | DBZ | LIS | INTCR,0 | CLEAR INTERRUPT COUNTERS | MPC14520 |
| 16FA | 2470 | | 1453 | | LIS | INTCW,0 | | MPC14530 |
| 16FC | 2480 | | 1454 | | LIS | INTCE,0 | | MPC14540 |
| 16FE | 24C0 | | 1455 | | LIS | R12,0 | CLEAR FLAGS | MPC14550 |
| 1700 | 24D0 | | 1456 | | LIS | R13,0 | | MPC14560 |
| 1702 | 2651 | | 1457 | | AIS | BLK,1 | INCREMENT BLOCK VALUE | MPC14570 |
| 1704 | 49E0 | 2264 | 1458 | | CH | BLK,BLOCKMAX | AT MAXIMUM? | MPC14580 |
| 1708 | 4220 | 173A | 1459 | | BTC | 2,DB6 | YES | MPC14590 |
| 170C | 08E5 | | 1460 | DB0 | LHR | R15,BLK | ELSE USE BLOCK NUMBER | MPC14600 |
| 170E | 27E1 | | 1461 | DB1 | SIS | R15,1 | DECREMENT | MPC14610 |
| 1710 | 2334 | | 1462 | | BZS | DB2 | LAST BLOCK | MPC14620 |
| 1712 | 4AC0 | 0C38 | 1463 | | AH | R12,BLOCKSZ | ADD BLOCK SIZE | MPC14630 |
| 1716 | 2204 | | 1464 | | BS | DB1 | NEXT BLOCK | MPC14640 |
| 1718 | 08DD | | 1465 | DB2 | LHR | R13,R13 | FLAG SET? | MPC14650 |
| 171A | 2134 | | 1466 | | BWZS | DB3 | YES | MPC14660 |
| 171C | 45C0 | 2282 | 1467 | | CLH | R12,ACTTOCLS | NO, EXAMINE ADDRESS | MPC14670 |
| 1720 | 238B | | 1468 | | BNLS | DB5 | EXIT IF > TOP OF MEMORY | MPC14680 |
| 1722 | 08EC | | 1469 | DB3 | LHR | R14,R12 | GET START ADR OF BLOCK | MPC14690 |
| 1724 | 4AE0 | 0C38 | 1470 | | AH | R14,BLOCKSZ | ADD BLOCK SIZE FOR END ADR | MPC14700 |
| 1728 | 27E2 | | 1471 | | SIS | R14,2 | | MPC14710 |
| 172A | 41F0 | 1840 | 1472 | | BAL | R15,QBLKPT | IS THIS BLOCK PROTECTED? | MPC14720 |
| 172E | 48E0 | 2222 | 1473 | DB4 | LH | R15,SAVEG | RESTORE REGISTERS | MPC14730 |
| 1732 | 2493 | | 1474 | | LIS | INCQR,3 | EXPECT ILLEGAL INSTRUCTION | MPC14740 |
| 1734 | 030F | | 1475 | | BR | R15 | RETURN | MPC14750 |
| 1736 | 24E1 | | 1476 | DB5 | LIS | R13,1 | SET THIS FLAG | MPC14760 |
| 1738 | 2303 | | 1477 | | BS | DB7 | CONTINUE | MPC14770 |

SUBROUTINES

| | | | | | | | | |
|------|------|------|------|-----|---------|---|----------------------------|----------|
| 173A | 4250 | 219C | 1478 | DB6 | LH | BLK,BLOCKREF | GET REF BLOCK NUMBER | MPC14780 |
| 173E | 4100 | 0000 | 1479 | DB7 | SVC | 0,0 | ZERO PSW 7 | MPC14790 |
| 1742 | 4130 | 220C | 1480 | | STH | R3,SAVEA | SAVE REGISTER | MPC14800 |
| 1746 | 4130 | 1782 | 1481 | | BAL | RET1,NEXTPSW | NEXT PSW 8-11 | MPC14810 |
| 174A | 4220 | 176A | 1482 | | BTC | 2,DB9 | EXIT IF > MAXIMUM | MPC14820 |
| 174E | 4230 | 220C | 1483 | | LH | R3,SAVEA | RESTORE REGISTER | MPC14830 |
| 1752 | 020D | | 1484 | | LHR | R13,R13 | FLAG SET? | MPC14840 |
| 1754 | 2237 | | 1485 | | BZS | DB8 | NO | MPC14850 |
| 1756 | 4250 | 2282 | 1486 | | LH | R13,ACTTOCLS | GET TOP OF MEMORY | MPC14860 |
| 175A | C5D0 | FF00 | 1487 | | CLHI | R13,X'FF00' | NEAR MAXIMUM VALUE? | MPC14870 |
| 175E | 4280 | 177C | 1488 | | BL | DBA | NO | MPC14880 |
| 1762 | 41E0 | 179C | 1489 | DB8 | BAL | RET2,DISPLAY | YES, DISPLAY DATA | MPC14890 |
| 1766 | 4200 | 170C | 1490 | | B | DB0 | CONTINUE | MPC14900 |
| 176A | 4230 | 220C | 1491 | DB9 | LH | R3,SAVEA | RESTORE REGISTER | MPC14910 |
| 176E | 2490 | | 1492 | | LIS | INCOR,0 | DO NOT EXPECT ILLEGAL INTR | MPC14920 |
| 1770 | C4F0 | FE8F | 1493 | | NHI | R15,X'FE8F' | RESET PSW 7,9-11 | MPC14930 |
| 1774 | 95EF | | 1494 | | EPSR | R14,R15 | NEW PSW | MPC14940 |
| 1776 | 42F0 | 2278 | 1495 | | LH | R15,TESTEND | GET EXIT ADDRESS | MPC14950 |
| 177A | 030F | | 1496 | | BR | R15 | EXIT TO THERE | MPC14960 |
| 177C | 24C0 | | 1497 | DBA | LIS | R12,0 | RESET ADDR FIRST | MPC14970 |
| 177E | 4300 | 170C | 1498 | | B | DB0 | | MPC14980 |
| | | | 1499 | | | | | MPC14990 |
| | | | 1500 | * | | | | MPC15000 |
| | | | 1501 | * | NEXTPSW | INCREMENT PSW 8-11 UNLESS MODEL 70 | | MPC15010 |
| | | | 1502 | * | | | | MPC15020 |
| 1782 | 95FF | | 1503 | | NEXTPSW | EPSR R15,R15 | SAVE PSW | MPC15030 |
| 1784 | 42E0 | 0C40 | 1504 | | LH | R14,MODEL70 | MODEL 70 ? | MPC15040 |
| 1788 | 4230 | 176A | 1505 | | BNZ | DB9 | YES,BRANCH | MPC15050 |
| 178C | 08EF | | 1506 | | LHR | R14,R15 | LOAD IT | MPC15060 |
| 178E | C4E0 | 0070 | 1507 | | NHI | R14,X'0070' | PSW 8-11 ONLY | MPC15070 |
| 1792 | CAE0 | 0010 | 1508 | | AHI | R14,X'10' | INCREMENT IT | MPC15080 |
| 1796 | 45E0 | 2276 | 1509 | | CLH | R14,PSW811 | COMPARE TO MAX AVAILABLE | MPC15090 |
| 179A | 0303 | | 1510 | | BR | RET1 | RETURN | MPC15100 |
| | | | 1511 | | | | | MPC15110 |
| | | | 1512 | * | | | | MPC15120 |
| | | | 1513 | * | DISPLAY | DISPLAYS PSW, PROTECT MODE AND FIRST PROTECTED BLOCK. | | MPC15130 |
| | | | 1514 | * | | ALSO SETS NEW PSW | | MPC15140 |
| | | | 1515 | * | | | | MPC15150 |
| | | | 1516 | * | | | | MPC15160 |
| 179C | 40E0 | 221C | 1517 | | DISPLAY | STH R13,SAVEE | SAVE REGISTER | MPC15170 |
| 17A0 | CAE0 | 0010 | 1518 | | AHI | R15,16 | ADD 16 | MPC15180 |
| 17A4 | 24C1 | | 1519 | | LIS | R12,1 | DISPLAY ADDRESS | MPC15190 |
| 17A6 | D1C0 | 0A34 | 1520 | | OC | R12,INCRMT | INC MODE FOR DISPLAY | MPC15200 |
| 17AA | 48D0 | 2266 | 1521 | | LH | R13,BLKPROT | GET DATA | MPC15210 |
| 17AE | 94ED | | 1522 | | EXBR | R13,R13 | SET BYTES | MPC15220 |
| 17B0 | 98CD | | 1523 | | WHR | R12,R13 | DISPLAY | MPC15230 |
| 17B2 | C4E0 | FFFF | 1524 | | NHI | R15,X'FFFF' | RESET PSW 12-15 | MPC15240 |
| 17B6 | 94FF | | 1525 | | EXBR | R15,R15 | SET BYTES | MPC15250 |
| 17B8 | 98CF | | 1526 | | WHR | R12,R15 | DISPLAY | MPC15260 |
| 17BA | 94FF | | 1527 | | EXBR | R15,R15 | RESTORE HW | MPC15270 |
| 17BC | DECO | 0A33 | 1528 | | OC | R12,NORM | RESTORE NORM MODE | MPC15280 |
| 17C0 | 24C0 | | 1529 | | LIS | R12,0 | ZERO | MPC15290 |
| 17C2 | 950F | | 1530 | | EPSR | R0,R15 | NEW PSW 0-15 | MPC15300 |

SUBROUTINES

| | | | | | | | |
|------|------|------|------|-----------|---|--------------------------------|----------|
| 17C4 | 48D0 | 221C | 1531 | LH | R13,SAVEE | RESTORE REGISTER | MPC15310 |
| 17C8 | 030E | | 1532 | BR | RET2 | RETURN | MPC15320 |
| | | | 1533 | ***** | | | MPC15330 |
| | | | 1534 | * | | | MPC15340 |
| | | | 1535 | *INCBLK | INCREMENTS PROTECTED BLOCK & COMPARE IT TO MAXIMUM | | MPC15350 |
| | | | 1536 | * | | | MPC15360 |
| | | | 1537 | * | | | MPC15370 |
| 17CA | 4850 | 2260 | 1538 | INCBLK | LH BLK,BLOCK2 | GET PROTECTED BLOCK | MPC15380 |
| 17CE | 2651 | | 1539 | AIS | BLK,1 | INCREMENT IT | MPC15390 |
| 17D0 | 4050 | 2260 | 1540 | STH | BLK,BLOCK2 | NEW PROTECTED BLOCK | MPC15400 |
| 17D4 | 4950 | 2198 | 1541 | CH | BLK,BLOCKMP | COMPARE TO MAXIMUM | MPC15410 |
| 17D8 | 030F | | 1542 | BR | R15 | RETURN | MPC15420 |
| 17DA | | | 1543 | BADR12 | DS 2 | | MPC15430 |
| | | | 1544 | ***** | | | MPC15440 |
| | | | 1545 | * | | | MPC15450 |
| | | | 1546 | *NEXTBLK | INCREMENTS THE NEXT BLOCK ADDRESS INDEX TO OBTAIN | | MPC15460 |
| | | | 1547 | * | THE ADDRESS FROM "BADRX" LIST. USED TO CALCULATE | | MPC15470 |
| | | | 1548 | * | ADDRESSES IN BLOCKS 0 TO BLOCKREF. | | MPC15480 |
| | | | 1549 | * | | | MPC15490 |
| | | | 1550 | * | | | MPC15500 |
| 17DC | 2651 | | 1551 | NEXTBLK | AIS BLK,1 | INCREMENT BLOCJK COUNT | MPC15510 |
| 17DE | 48F0 | 0C38 | 1552 | LH | R15,BLOCKSZ | GET BLOCK SIZE | MPC15520 |
| 17E2 | 90F8 | | 1553 | SRLS | R15,8 | ONLY MOST SIG 2 BITS | MPC15530 |
| 17E4 | 0ADF | | 1554 | AHR | R13,R15 | INCREMENT INDEX | MPC15540 |
| 17E5 | C5D0 | 0024 | 1555 | CLHI | R13,36 | COMPARE TO LAST ENTRY OF BADRX | MPC15550 |
| 17EA | 0303 | | 1556 | BR | RET1 | RETURN | MPC15560 |
| | | | 1557 | ***** | | | MPC15570 |
| | | | 1558 | * | | | MPC15580 |
| | | | 1559 | *INCLWBLK | PROTECT NEXT BLOCK OF PROGRAM MEMORY | | MPC15590 |
| | | | 1560 | * | | | MPC15600 |
| | | | 1561 | * | | | MPC15610 |
| 17EC | C8F0 | 6000 | 1562 | INCLWBL | LHI R15,X'6000' | ENABLE EXTERNAL INTERRUPTS | MPC15620 |
| 17F0 | 950F | | 1563 | EPSR | R0,R15 | NEW PSW 0-15 | MPC15630 |
| 17F2 | 4850 | 2260 | 1564 | LH | BLK,BLOCK2 | GET CURRENT PROTECTED BLOCK | MPC15640 |
| 17F6 | 2651 | | 1565 | AIS | BLK,1 | NEXT BLOCK | MPC15650 |
| 17F8 | 4550 | 219C | 1566 | CLH | BLK,BLOCKREF | COMPARE TO MAX (REF) | MPC15660 |
| 17FC | 030E | | 1567 | BR | RET2 | RETURN | MPC15670 |
| | | | 1568 | ***** | | | MPC15680 |
| | | | 1569 | * | | | MPC15690 |
| | | | 1570 | *PSW8X | ADJUST BLOCK COUNT, ADDRESSES & PROTECT STATUS WHEN | | MPC15700 |
| | | | 1571 | * | PSW 8-11 = 8 . | | MPC15710 |
| | | | 1572 | * | | | MPC15720 |
| | | | 1573 | * | | | MPC15730 |
| 17FE | 40F0 | 222C | 1574 | PSW8X | STH R15,SAVEJ | SAVE REGISTER | MPC15740 |
| 1802 | 95FF | | 1575 | EPSR | R15,R15 | GET PSW | MPC15750 |
| 1804 | C4F0 | 00F0 | 1576 | NHI | R15,X'F0' | ONLY PSW 8-11 | MPC15760 |
| 1808 | C5F0 | 0080 | 1577 | CLHI | R15,X'80' | ONLY PSW 8 SET? | MPC15770 |
| 180C | 4230 | 183A | 1578 | BNE | PSW8X2 | NO, CONTINUE | MPC15780 |
| 1810 | 4840 | 2264 | 1579 | LH | R4,BLOCKMAX | YES, GET MAXIMUM BLOCK | MPC15790 |
| 1814 | 9041 | | 1580 | SRLS | R4,1 | GET ONE HALF | MPC15800 |
| 1816 | 2641 | | 1581 | AIS | R4,1 | PLUS 1 | MPC15810 |
| 1818 | 0554 | | 1582 | CLHR | BLK,R4 | COMPARE BLOCK TO THIS NO. | MPC15820 |
| 181A | 4280 | 183A | 1583 | BL | PSW8X2 | < CONTINUE | MPC15830 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|---------|---|-----------------------------------|----------|
| 181E | 2136 | 1584 | BNES | PSW8X1 | > GET NEW PROTECT STATUS | MPC15840 |
| 1820 | 4A50 219C | 1585 | AH | BLK,BLOCKREF | = ADD BLOCKREF VALUE | MPC15850 |
| 1824 | 2752 | 1586 | SIS | BLK,2 | ADJUST FOR BLOCK ADDR CALCULATION | MPC15860 |
| 1826 | 41F0 16F4 | 1587 | BAL | R15,BLKADR | GET ADDRESSES OF NEW BLOCK | MPC15870 |
| 182A | 4050 2262 | 1588 | PSW8X1 | STH BLK,BLOCK3 | SAVE REGISTER | MPC15880 |
| 182E | 0B54 | 1589 | SHR | BLK,R4 | RESTORE BLOCK VALUE | MPC15890 |
| 1830 | 2651 | 1590 | AIS | BLK,1 | | MPC15900 |
| 1832 | 41F0 1840 | 1591 | BAL | R15,QBLKPT | GET PROTECT STATUS OF BLOCK | MPC15910 |
| 1836 | 4850 2262 | 1592 | LH | BLK,BLOCK3 | RESTORE REGISTER | MPC15920 |
| 183A | 48F0 222C | 1593 | PSW8X2 | LH R15,SAVEJ | RESTORE REGISTERS | MPC15930 |
| 183E | 030F | 1594 | BR | R15 | RETURN | MPC15940 |
| | | 1595 | ***** | | | MPC15950 |
| | | 1596 | * | | MPC15960 | |
| | | 1597 | *QBLKPT | DETERMINES THE PROTECTION STATUS OF THE BLOCK (CONTENTS | MPC15970 | |
| | | 1598 | * | OF REGISTER "BLK"). | MPC15980 | |
| | | 1599 | * | | MPC15990 | |
| | | 1600 | * | IF THE BLOCK IS WRITE-READ PROTECTED, FLAG "BLKPT" | MPC16000 | |
| | | 1601 | * | IS SET; IF IT IS EXECUTE PROTECTED FLAG "BLKPT" IS | MPC16010 | |
| | | 1602 | * | SET; OTHERWISE THESE FLAGS ARE CLEARED. | MPC16020 | |
| | | 1603 | * | | MPC16030 | |
| | | 1604 | * | REGISTERS R13-15 ARE SAVED. | MPC16040 | |
| | | 1605 | * | | MPC16050 | |
| | | 1606 | * | | MPC16060 | |
| 1840 | D0D0 221C | 1607 | QBLKPT | STM R13,SAVEE | STORE REGISTERS | MPC16070 |
| 1844 | 07FF | 1608 | XHR | R15,R15 | ZERO | MPC16080 |
| 1846 | 24E9 | 1609 | LIS | R14,X'9' | 9 | MPC16090 |
| 1848 | 4050 225E | 1610 | STH | BLK,BLOCK | SAVE BLOCK NO. | MPC16100 |
| 184C | C9E0 0041 | 1611 | CHI | BLK,65 | COMPARE TO 65 | MPC16110 |
| 1850 | 2183 | 1612 | BLS | QBP1 | < W-R PROTECT CANDIDATE | MPC16120 |
| 1852 | C850 0040 | 1613 | SHI | BLK,64 | ELSE EXECUTE PROTECT CAND. | MPC16130 |
| 1855 | 095E | 1614 | QBP1 | CHR BLK,R14 | BLOCK RANGE | MPC16140 |
| 1858 | 2184 | 1615 | BLS | QBP3 | FOUND RANGE | MPC16150 |
| 185A | 26F2 | 1616 | AIS | R15,2 | INC BY 2 | MPC16160 |
| 185C | 26E8 | 1617 | AIS | R14,8 | INC BY 8 | MPC16170 |
| 185E | 22C4 | 1618 | BS | QBP1 | USE NEW VALUES | MPC16180 |
| 1860 | 48DF 21DC | 1619 | QBP3 | LH R13,MPHF1(R15) | GET PROT STATE | MPC16190 |
| 1864 | 0BE5 | 1620 | SHR | R14,BLK | GET BIT POSITION | MPC16200 |
| 1865 | 27E1 | 1621 | QBP4 | SIS R14,1 | FIND BIT POSITION | MPC16210 |
| 1868 | 2333 | 1622 | BZS | QBP5 | FOUND IT | MPC16220 |
| 186A | 90D1 | 1623 | SRLS | R13,1 | NEXT BIT POSITION | MPC16230 |
| 186C | 2203 | 1624 | BS | QBP4 | NEXT BIT | MPC16240 |
| 186E | C3D0 0001 | 1625 | QBP5 | THI R13,X'01' | TEST BIT | MPC16250 |
| 1872 | 4230 189A | 1626 | BMZ | QBP6 | PROTECTED | MPC16260 |
| 1876 | 07FF | 1627 | XHR | R15,R15 | NOT PROTECTED | MPC16270 |
| 1878 | C9E0 0041 | 1628 | QBP7 | CHI BLK,65 | EX. PROTECT STATUS DETERMINED? | MPC16280 |
| 187C | 2388 | 1629 | BNLS | QBP8 | YES, EXIT | MPC16290 |
| 187E | 40F0 2268 | 1630 | STH | R15,BLKPT | NO, SAVE W-R PROTECT STATUS | MPC16300 |
| 1882 | CAC0 0040 | 1631 | AHI | BLK,64 | ADJUST PARAMETERS | MPC16310 |
| 1886 | 26E9 | 1632 | AIS | R14,9 | TO DETERMINE EX PROTECT STATUS | MPC16320 |
| 1888 | 4300 1866 | 1633 | B | QBP4 | OF THIS BLOCK | MPC16330 |
| 188C | 40F0 226A | 1634 | QBP8 | STH R15,BLKPT | SAVE EX PROTECT STATUS | MPC16340 |
| 1890 | 4850 225E | 1635 | LH | BLK,BLOCK | RESTORE BLOCK VALUE | MPC16350 |
| 1894 | D1D0 221C | 1636 | LN | R13,SAVEE | AND REGISTERS | MPC16360 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|-----------|---|----------------------------------|----------|
| 1898 | 030F | 1637 | BR | R15 | RETURN | MPC16370 |
| 189A | 24F1 | 1638 | QBP6 | LIS R15,1 | SET FLAG | MPC16380 |
| 189C | 4300 1878 | 1639 | B | QBP7 | EXIT | MPC16390 |
| | | 1640 | ***** | | | MPC16400 |
| | | 1641 | * | | | MPC16410 |
| | | 1642 | *PLBLOCKS | GENERATES A MEMORY PROTECT MAP ACCORDING TO THE | | MPC16420 |
| | | 1643 | * | VALUE OF "PROTPATT". IF VALUE = 1 EVERY BLOCK IS | | MPC16430 |
| | | 1644 | * | PROTECTED, IF = 2 EVERY SECOND, IF = 3 EVERY | | MPC16440 |
| | | 1645 | * | THIRD, ETC. | | MPC16450 |
| | | 1646 | * | | | MPC16460 |
| | | 1647 | * | | | MPC16470 |
| 18A0 | 40F0 2216 | 1648 | PTBLOCKS | STH R15,SAVEC | SAVE REGISTERS | MPC16480 |
| 18A4 | 40E0 2218 | 1649 | | STH BLK,SAVEC+2 | SAVE BLOCK NO. | MPC16490 |
| 18A8 | 4850 219C | 1650 | | LH BLK,BLOCKREF | GET REFERENCE BLOCK | MPC16500 |
| 18AC | 2751 | 1651 | | SIS BLK,1 | ADJUST VALUE | MPC16510 |
| 18AE | 4A50 2274 | 1652 | PTB1 | AH BLK,PROTPATT | SET BLOCK COUNTER VALUE | MPC16520 |
| 18B2 | 4950 2198 | 1653 | | CH BLK,BLOCKMP | AT MAXIMUM BLOCK VALUE? | MPC16530 |
| 18B6 | 2124 | 1654 | | BTFS 2,PTB2 | YES EXIT | MPC16540 |
| 18B8 | 41F0 1620 | 1655 | | BAL R15,PTALLBLK | NO, PROTECT ANOTHER BLOCK | MPC16550 |
| 18BC | 2207 | 1656 | | BS PTB1 | | MPC16560 |
| 18BE | 48F0 2216 | 1657 | PTB2 | LH R15,SAVEC | RESTORE REGISTERS | MPC16570 |
| 18C2 | 4850 2218 | 1658 | | LH BLK,SAVEC+2 | AND BLOCK NO. | MPC16580 |
| 18C6 | 030F | 1659 | | BR R15 | RETURN | MPC16590 |
| | | 1660 | ***** | | | MPC16600 |
| | | 1661 | * | | | MPC16610 |
| | | 1662 | *ADRTRAN | TRANSLATES ACTUAL ADR TO PROGRAM ADR & SET PSW 8-11 | | MPC16620 |
| | | 1663 | * | | | MPC16630 |
| | | 1664 | * | ON ENTRY "ACTADUP" CONTAINS THE TWO MOST SIG BITS OF THE ACTUAL | | MPC16640 |
| | | 1665 | * | ADDRESS & R11 CONTAINS LEAST SIG BITS (16). | | MPC16650 |
| | | 1666 | * | | | MPC16660 |
| | | 1667 | * | ON EXIT R12 CONTAINS THE PROGRAM ADDRESS AND THE PSW BITS 8-11 | | MPC16670 |
| | | 1668 | * | ARE MODIFIED. | | MPC16680 |
| | | 1669 | * | | | MPC16690 |
| | | 1670 | * | | | MPC16700 |
| 18C8 | D0D0 2216 | 1671 | ADRTRAN | STM R13,SAVEC | SAVE REGISTERS | MPC16710 |
| 18CC | 08CB | 1672 | | LHR R12,R11 | PUT ACT ADR IN R12 | MPC16720 |
| 18CE | 95DD | 1673 | | EPSR R13,R13 | GET CURRENT PSW | MPC16730 |
| 18D0 | C4D0 FFOF | 1674 | | NHI R13,X'FFOF' | ZERO BITS 8-11 | MPC16740 |
| 18D4 | 48E0 2284 | 1675 | | LH R14,ACTADUP | GET MS BITS OF ACTUAL ADR | MPC16750 |
| 18D8 | C4E0 0003 | 1676 | | NHI R14,3 | USE ONLY FIRST TWO BITS | MPC16760 |
| 18DC | 233B | 1677 | | BZS ADRTRANQ | | MPC16770 |
| 18DE | D3FE 18FA | 1678 | | LB R15,XADRTRAN(R14) | FIND POTENTIAL VALUE OF PSW 8-11 | MPC16780 |
| 18E2 | CAC0 8000 | 1679 | | AHI R12,X'8000' | ADD 8000 TO ACTUAL ADR | MPC16790 |
| 18E6 | 4EFO 2190 | 1680 | | ACH R15,ZERO | INC | MPC16800 |
| 18EA | C6C0 8000 | 1681 | | OHI R12,X'8000' | CONVERT TO PROGRAM ADR | MPC16810 |
| 18EE | 91F4 | 1682 | | SLLS R15,4 | PLA E IN BITS 8-11 | MPC16820 |
| 18F0 | 06DF | 1683 | | OHR R13,R15 | ADD TO CURRENT PSW HW | MPC16830 |
| 18F2 | 95FD | 1684 | ADRTRANQ | EPSR R15,R13 | NEW PSW | MPC16840 |
| 18F4 | D1D0 2216 | 1685 | | LM R13,SAVEC | RESTORE REGISTERS | MPC16850 |
| 18F8 | 030F | 1686 | | BR R15 | RETURN | MPC16860 |
| | | 1687 | * | | | MPC16870 |
| 18FA | 0001 0305 | 1688 | XADRTRAN | DB 0,1,3,5 | | MPC16880 |
| | | 1689 | ***** | | | MPC16890 |

SUBROUTINES

| | | | | | | | | | | |
|------|-----------|------|----------|--|-------------|--|--|--|----------|----------|
| | | 1690 | * | | | | | | NPC16900 | |
| | | 1691 | *BLKNUM | SETS BLOCK NUMBER FOR DISPLAY | | | | | NPC16910 | |
| | | 1692 | * | | | | | | NPC16920 | |
| | | 1693 | * | | | | | | NPC16930 | |
| 18FE | 0815 | 1694 | BLKNUM | LHR | R1,BLK | | | | NPC16940 | |
| 1900 | 41E0 1DF2 | 1695 | | BAL | R14,CONVERT | | | | NPC16950 | |
| 1904 | 0004 | 1696 | | DC | X'4' | | | | NPC16960 | |
| 1906 | 20B4 | 1697 | | DC | Z(BLOCKN) | | | | NPC16970 | |
| 1908 | 030F | 1698 | | BR | R15 | | | | NPC16980 | |
| 190A | | 1699 | BADR13 | DS | 2 | | | | NPC16990 | |
| | | 1700 | ***** | | | | | | | NPC17000 |
| | | 1701 | * | | | | | | NPC17010 | |
| | | 1702 | *REGSN | SETS WRITE, READ & EXECUTE VIOLATIONS COUNTERS | | | | | NPC17020 | |
| | | 1703 | * | FOR DISPLAYS | | | | | NPC17030 | |
| | | 1704 | * | | | | | | NPC17040 | |
| | | 1705 | * | | | | | | NPC17050 | |
| 190C | 0816 | 1706 | REGSN | LHR | R1,R6 | | | | NPC17060 | |
| 190E | 41E0 1DF2 | 1707 | | BAL | R14,CONVERT | | | | NPC17070 | |
| 1912 | 0004 | 1708 | | DC | X'4' | | | | NPC17080 | |
| 1914 | 20CC | 1709 | | DC | Z(WN) | | | | NPC17090 | |
| 1916 | 0817 | 1710 | | LHR | R1,R7 | | | | NPC17100 | |
| 1918 | 41E0 1DF2 | 1711 | | BAL | R14,CONVERT | | | | NPC17110 | |
| 191C | 0004 | 1712 | | DC | X'4' | | | | NPC17120 | |
| 191E | 20D0 | 1713 | | DC | Z(RN) | | | | NPC17130 | |
| 1920 | 0818 | 1714 | | LHR | R1,R8 | | | | NPC1714Q | |
| 1922 | 41E0 1DF2 | 1715 | | BAL | R14,CONVERT | | | | NPC17150 | |
| 1926 | 0004 | 1716 | | DC | X'4' | | | | NPC17160 | |
| 1928 | 20D4 | 1717 | | DC | Z(EN) | | | | NPC17170 | |
| 192A | 030F | 1718 | | BR | R15 | | | | NPC17180 | |
| | | 1719 | ***** | | | | | | | NPC17190 |
| | | 1720 | * | | | | | | NPC17200 | |
| | | 1721 | *PSPACE | PRINTS SPACE ON DISPLAY CONSOLE | | | | | NPC17210 | |
| | | 1722 | * | | | | | | NPC17220 | |
| | | 1723 | * | | | | | | NPC17230 | |
| 192C | 41F0 1EA4 | 1724 | PSPACE | BAL | R15,PRINT | | | | NPC17240 | |
| 1930 | 20C4 | 1725 | | DC | Z(SPACE2) | | | | NPC17250 | |
| 1932 | 20C5 | 1726 | | DC | Z(SPACE2+1) | | | | NPC17260 | |
| 1934 | 0301 | 1727 | | BR | R1 | | | | NPC17270 | |
| | | 1728 | ***** | | | | | | | NPC17280 |
| | | 1729 | * | | | | | | NPC17290 | |
| | | 1730 | *PCRLF | PRNTS CARRIAGE RETURN & LINE FEED ON DISPLAY CONSOLE | | | | | NPC17300 | |
| | | 1731 | * | | | | | | NPC17310 | |
| | | 1732 | * | | | | | | NPC17320 | |
| 1936 | 41F0 1EA4 | 1733 | PCRLF | BAL | R15,PRINT | | | | NPC17330 | |
| 193A | 20AC | 1734 | | DC | Z(CRLF) | | | | NPC17340 | |
| 193C | 20AD | 1735 | | DC | Z(CRLF+1) | | | | NPC17350 | |
| 193E | 0301 | 1736 | | BR | R1 | | | | NPC17360 | |
| | | 1737 | ***** | | | | | | | NPC17370 |
| | | 1738 | * | | | | | | NPC17380 | |
| | | 1739 | *MESSOUT | DISPLAYS THE OPTION LABELS AND VALUE ON CONSOLE | | | | | NPC17390 | |
| | | 1740 | * | | | | | | NPC17400 | |
| | | 1741 | * | | | | | | NPC17410 | |
| 0000 | 1940 | 1742 | MESSOUT | EQU | * | | | | NPC17420 | |

SUBROUTINES

| | | | | | | | |
|------|------|------|------|----------|--|----------------------------|----------|
| 1940 | 4110 | 1936 | 1743 | BAL | R1,PCRLF | CARRIAGE RETURN ,LINE FEED | MPC17430 |
| 1944 | 41F0 | 1EA4 | 1744 | BAL | R15,PRINT | PRINT 'TEST' | MPC17440 |
| 1948 | 0C1A | | 1745 | DC | Z(TEST+2) | | MPC17450 |
| 194A | 0C1F | | 1746 | DC | Z(TEST+7) | | MPC17460 |
| 194C | 4110 | 192C | 1747 | BAL | R1,PSPACE | PRINT TWO SPACES | MPC17470 |
| 1950 | 4820 | 0C18 | 1748 | LH | R2,TEST | GET TEST NUMBERS | MPC17480 |
| 1954 | 24G0 | | 1749 | LIS | R0,0 | | MPC17490 |
| 1956 | 2441 | | 1750 | LIS | R4,1 | | MPC17500 |
| 1958 | 9121 | | 1751 | OPTCMD3 | SLHLS R2,1 | | MPC17510 |
| 195A | 4380 | 1982 | 1752 | BNC | OPTCMD7 | | MPC17520 |
| 195E | 0800 | | 1753 | LHR | R0,R0 | | MPC17530 |
| 1960 | 4230 | 1968 | 1754 | BNZ | OPTCMD5 | | MPC17540 |
| 1964 | 2501 | | 1755 | LCS | R0,1 | | MPC17550 |
| 1966 | 2305 | | 1756 | BS | OPTCMD6 | | MPC17560 |
| 1968 | 41F0 | 1EA4 | 1757 | OPTCMD5 | BAL R15,PRINT | PRINT A COMMA | MPC17570 |
| 196C | 0A20 | | 1758 | DC | Z(COMMA) | | MPC17580 |
| 196E | 0A20 | | 1759 | DC | Z(COMMA) | | MPC17590 |
| 1970 | 0814 | | 1760 | OPTCMD6 | LHR R1,R4 | | MPC17600 |
| 1972 | 41E0 | 1DF2 | 1761 | BAL | R14,CONVERT | | MPC17610 |
| 1976 | 000C | | 1762 | DC | X'C' | | MPC17620 |
| 1978 | 226E | | 1763 | DC | Z(MESSAV) | | MPC17630 |
| 197A | 41F0 | 1EA4 | 1764 | BAL | R15,PRINT | PRINT TEST NUMBERS | MPC17640 |
| 197E | 2271 | | 1765 | DC | Z(MESSAV+3) | | MPC17650 |
| 1980 | 2271 | | 1766 | DC | Z(MESSAV+3) | | MPC17660 |
| 1982 | 2641 | | 1767 | OPTCMD7 | AIS R4,1 | | MPC17670 |
| 1984 | C540 | 0011 | 1768 | CLHI | R4,17 | | MPC17680 |
| 1988 | 4280 | 1958 | 1769 | BL | OPTCMD3 | | MPC17690 |
| 198C | 0766 | | 1770 | XHR | R6,R6 | OUTPUT PARAMETER VALUES | MPC17700 |
| 198E | C816 | 0C22 | 1771 | MESSOUT1 | LHI R1,NOMSG+2(R6) | | MPC17710 |
| 1992 | C510 | 0C50 | 1772 | CLHI | R1,OPTEND | ALL OPTIONS PRINTED? | MPC17720 |
| 1996 | 4380 | 0C7C | 1773 | BNL | TTYIN | YES,EXIT | MPC17730 |
| 199A | C826 | 0C27 | 1774 | LHI | R2,NOMSG+7(R6) | | MPC17740 |
| 199E | 4010 | 19AE | 1775 | STH | R1,PRTX | | MPC17750 |
| 19A2 | 4020 | 19B0 | 1776 | STH | R2,PRTZ | | MPC17760 |
| 19A6 | 4110 | 1936 | 1777 | MESSX | BAL R1,PCRLF | | MPC17770 |
| 19AA | 41F0 | 1EA4 | 1778 | MESSZ | BAL R15,PRINT | | MPC17780 |
| 19AE | 0C22 | | 1779 | PRTX | DC Z(NOMSG+2) | | MPC17790 |
| 19B0 | 0C27 | | 1780 | PRTZ | DC Z(NOMSG+7) | | MPC17800 |
| 19B2 | 4110 | 192C | 1781 | BAL | R1,PSPACE | | MPC17810 |
| 19B6 | 4816 | 0C20 | 1782 | LH | R1,NOMSG(R6) | | MPC17820 |
| 19BA | 41E0 | 1DF2 | 1783 | BAL | R14,CONVERT | | MPC17830 |
| 19BE | 000C | | 1784 | DC | X'C' | | MPC17840 |
| 19C0 | 226E | | 1785 | DC | Z(MESSAV) | | MPC17850 |
| 19C2 | 41F0 | 1EA4 | 1786 | BAL | R15,PRINT | | MPC17860 |
| 19C6 | 226E | | 1787 | DC | Z(MESSAV) | | MPC17870 |
| 19C8 | 2271 | | 1788 | DC | Z(MESSAV+3) | | MPC17880 |
| 19CA | 2668 | | 1789 | MIS | R6,8 | | MPC17890 |
| 19CC | 4300 | 198E | 1790 | B | MESSOUT1 | | MPC17900 |
| | | | 1791 | | ***** | | MPC17910 |
| | | | 1792 | | * | | MPC17920 |
| | | | 1793 | | *MAPOUT DISPLAYS THE MEMORY PROTECT MAP ON THE CONSOLE | | MPC17930 |
| | | | 1794 | | * | | MPC17940 |
| | | | 1795 | | * | | MPC17950 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|--|------|--------------|----------|
| 19D0 | 0766 | 1796 | MAPOUT | XHR | R6,R6 | MPC17960 |
| 19D2 | 4810 20FA | 1797 | | LH | R1,VALUE | MPC17970 |
| 19D6 | C510 3037 | 1798 | | CLHI | R1,X'3037' | MPC17980 |
| 19DA | 2133 | 1799 | | BNES | MAPO00 | MPC17990 |
| 19DC | C860 8000 | 1800 | | LHI | R6,X'8000' | MPC18000 |
| 19E0 | 4110 1936 | 1801 | MAPO00 | BAL | R1,PCRLF | MPC18010 |
| 19E4 | 4846 21DC | 1802 | MAPO0 | LH | R4,MPHF1(R6) | MPC18020 |
| 19E8 | 0777 | 1803 | | XHR | R7,R7 | MPC18030 |
| 19EA | C340 8000 | 1804 | MAPO | THI | R4,X'8000' | MPC18040 |
| 19EE | 2334 | 1805 | | BZS | MAP1 | MPC18050 |
| 19FO | C880 0031 | 1806 | | LHI | R8,X'31' | MPC18060 |
| 19F4 | 2303 | 1807 | | BS | MAP2 | MPC18070 |
| 19F6 | C880 0030 | 1808 | MAP1 | LHI | R8,X'30' | MPC18080 |
| 19FA | 4080 226C | 1809 | MAP2 | STH | R8,MAPSAVE | MPC18090 |
| 19FE | 41F0 1EA4 | 1810 | | BAL | R15,PRINT | MPC18100 |
| 1A02 | 226D | 1811 | | DC | Z(MAPSAVE+1) | MPC18110 |
| 1A04 | 226D | 1812 | | DC | Z(MAPSAVE+1) | MPC18120 |
| 1A06 | 2671 | 1813 | | AIS | R7,1 | MPC18130 |
| 1A08 | 9141 | 1814 | | SLLS | R4,1 | MPC18140 |
| 1A0A | C370 0003 | 1815 | | THI | R7,X'3' | MPC18150 |
| 1A0E | 4230 19EA | 1816 | | BWZ | MAPO | MPC18160 |
| 1A12 | 4110 192C | 1817 | | BAL | R1,PSPACE | MPC18170 |
| 1A15 | C970 0010 | 1818 | | CHI | R7,16 | MPC18180 |
| 1A1A | 4280 19EA | 1819 | | BL | MAPO | MPC18190 |
| 1A1E | 4110 1936 | 1820 | | BAL | R1,PCRLF | MPC18200 |
| 1A22 | 2662 | 1821 | | AIS | R6,2 | MPC18210 |
| 1A24 | C360 0010 | 1822 | | THI | R6,16 | MPC18220 |
| 1A28 | 4330 19E4 | 1823 | | BZ | MAPO0 | MPC18230 |
| 1A2C | 030E | 1824 | | BR | R14 | MPC18240 |
| | | 1825 | ***** | | | MPC18250 |
| | | 1826 | * | | | MPC18260 |
| | | 1827 | *PRTSTAT DISPLAYS TEST, ERROR AND BLOCK NUMBERS AND DEVICE | | | MPC18270 |
| | | 1828 | * ADDRESS AND STATUS ON CONSOLE. | | | MPC18280 |
| | | 1829 | * | | | MPC18290 |
| | | 1830 | * | | | MPC18300 |
| 1A2E | 08D0 | 1831 | PRTSTAT | LHR | R13,R0 | MPC18310 |
| 1A30 | E110 0000 | 1832 | | SVC | 1,0 | MPC18320 |
| 1A34 | 080D | 1833 | | LHR | R0,R13 | MPC18330 |
| 1A36 | 2490 | 1834 | | LIS | INCQR,0 | MPC18340 |
| 1A38 | 4010 20AA | 1835 | | STH | R1,ERRNUM | MPC18350 |
| 1A3C | 08AF | 1836 | | LHR | R10,R15 | MPC18360 |
| 1A3E | 41F0 18FE | 1837 | | BAL | R15,BLKNUM | MPC18370 |
| 1A42 | 931A | 1838 | | LBR | R1,R10 | MPC18380 |
| 1A44 | 41E0 1DF2 | 1839 | | BAL | R14,CONVERT | MPC18390 |
| 1A48 | 0004 | 1840 | | DC | X'4' | MPC18400 |
| 1A4A | 20E2 | 1841 | | DC | Z(STATUS) | MPC18410 |
| 1A4C | 0810 | 1842 | | LHR | R1,R0 | MPC18420 |
| 1A4E | 41E0 1DF2 | 1843 | | BAL | R14,CONVERT | MPC18430 |
| 1A52 | 0004 | 1844 | | DC | X'4' | MPC18440 |
| 1A54 | 20EE | 1845 | | DC | Z(DEVADD) | MPC18450 |
| 1A56 | 41F0 1EA4 | 1846 | | BAL | R15,PRINT | MPC18460 |
| 1A5A | 20A0 | 1847 | | DC | Z(ERRMSG) | MPC18470 |
| 1A5C | 20B7 | 1848 | | DC | Z(ENDZ) | MPC18480 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|----------|---|---------------|----------|
| 1A5E | 41F0 1EA4 | 1849 | STATZ | BAL | R15,PRINT | NPC18490 |
| 1A62 | 20DA | 1850 | | DC | Z(STATMSG) | NPC18500 |
| 1A64 | 20F1 | 1851 | | DC | Z(STATEND) | NPC18510 |
| 1A66 | 2401 | 1852 | INCERR | LIS | RO,1 | NPC18520 |
| 1A68 | 6100 21D6 | 1853 | | AHM | RO,TOTALERR | NPC18530 |
| 1A6C | 4300 0E44 | 1854 | | B | TSTSEL | NPC18540 |
| | | 1855 | ***** | | | NPC18550 |
| | | 1856 | * | | | NPC18560 |
| | | 1857 | *PRTREGS | DISPLAYS TEST, ERROR AND BLOCK NUMBERS AND THE NUMBER | | NPC18570 |
| | | 1858 | * | OF WRITE, READ AND EXECUTE PROTECT VIOLATIONS ON THE | | NPC18580 |
| | | 1859 | * | CONSOLE. | | NPC18590 |
| | | 1860 | * | | | NPC18600 |
| | | 1861 | * | | | NPC18610 |
| 1A70 | 08D0 | 1862 | PRTREGS | LHR | R13,RO | NPC18620 |
| 1A72 | E110 0000 | 1863 | | SVC | 1,0 | NPC18630 |
| 1A76 | 080D | 1864 | | LHR | RO,R13 | NPC18640 |
| 1A78 | 2490 | 1865 | | LIS | INCQR,0 | NPC18650 |
| 1A7A | 4010 20AA | 1866 | | STH | R1,ERRNUM | NPC18660 |
| 1A7E | 41F0 18FE | 1867 | | BAL | R15,BLKNUM | NPC18670 |
| 1A82 | 41F0 190C | 1868 | | BAL | R15,REGSM | NPC18680 |
| 1A86 | 41F0 1EA4 | 1869 | | BAL | R15,PRINT | NPC18690 |
| 1A8A | 20A0 | 1870 | | DC | Z(ERRMSG) | NPC18700 |
| 1A8C | 20B7 | 1871 | | DC | Z(ENDZ) | NPC18710 |
| 1A8E | 41F0 1EA4 | 1872 | | BAL | R15,PRINT | NPC18720 |
| 1A92 | 20CC | 1873 | | DC | Z(WN) | NPC18730 |
| 1A94 | 20D9 | 1874 | | DC | Z(ENDA) | NPC18740 |
| 1A96 | 4300 1A66 | 1875 | | B | INCERR | NPC18750 |
| | | 1876 | ***** | | | NPC18760 |
| | | 1877 | * | | | NPC18770 |
| | | 1878 | *PRTDATA | DISPLAYS THE TEST, ERROR AND BLOCK NUMBER AND THE | | NPC18780 |
| | | 1879 | * | ADDRESS (18 BITS) OF THE MEMORY ERROR ON THE | | NPC18790 |
| | | 1880 | * | CONSOLE. | | NPC18800 |
| | | 1881 | * | | | NPC18810 |
| | | 1882 | * | | | NPC18820 |
| 1A9A | 08D0 | 1883 | PRTDATA | LHR | R13,RO | NPC18830 |
| 1A9C | E100 0000 | 1884 | | SVC | 0,0 | NPC18840 |
| 1AA0 | E160 0000 | 1885 | | SVC | 6,0 | NPC18850 |
| 1AA4 | 080D | 1886 | | LHR | RO,R13 | NPC18860 |
| 1AA6 | 2490 | 1887 | | LIS | INCQR,0 | NPC18870 |
| 1AA8 | 4010 20AA | 1888 | | STH | R1,ERRNUM | NPC18880 |
| 1AAC | 41F0 18FE | 1889 | | BAL | R15,BLKNUM | NPC18890 |
| 1AB0 | 95DD | 1890 | | EPSR | R13,R13 | NPC18900 |
| 1AB2 | C5C0 8000 | 1891 | | CLHI | R12,X'8000' | NPC18910 |
| 1AB6 | 218A | 1892 | | BLS | PRTD1 | NPC18920 |
| 1AB8 | C4C0 0070 | 1893 | | MHI | R13,X'0070' | NPC18930 |
| 1ABC | 90D4 | 1894 | | SRLS | R13,4 | NPC18940 |
| 1ABE | D35D 1AFA | 1895 | | LB | R5,PRTDZ(R13) | NPC18950 |
| 1AC2 | C3E0 0001 | 1896 | | THI | R13,1 | NPC18960 |
| 1AC6 | 2338 | 1897 | | BZS | PRTD3 | NPC18970 |
| 1AC8 | 2305 | 1898 | | BS | PRTD2 | NPC18980 |
| 1ACA | 2450 | 1899 | PRTD1 | LIS | R5,0 | NPC18990 |
| 1ACC | C3D0 0080 | 1900 | | THI | R13,X'0080' | NPC19000 |
| 1ADD | 2333 | 1901 | | BZS | PRTD3 | NPC19010 |

SUBROUTINES

| | | | | | | | | |
|------|------|------|------|---------|--------------|-------------------------|----------------------|----------|
| 1B58 | D200 | 0A1C | 1955 | STB | RO,ADDRESS | CONSOLE ADDR | MPC19550 | |
| 1B5C | D300 | 0A32 | 1956 | LB | RO,TTYRD | TTY READ | MPC19560 | |
| 1B60 | D200 | 0A30 | 1957 | STB | RO,RDCMD | COMMAND | MPC19570 | |
| 1B64 | 2400 | | 1958 | LIS | RO,0 | ZERO | MPC19580 | |
| 1B66 | 4000 | 0A24 | 1959 | STH | RO,PASFLG | PASLA FLAG | MPC19590 | |
| 1B6A | 4300 | 1BB2 | 1960 | B | DEVCHK1 | LIST DEVICE NEXT | MPC19600 | |
| 1B6E | D300 | 0A2B | 1961 | DEVK1 | LB | RO,PASLWRT | PASLA WRITE | MPC19610 |
| 1B72 | D200 | 0A22 | 1962 | STB | RO,WRTCMD | WRITE COMMAND | MPC19620 | |
| 1B76 | D300 | 0A14 | 1963 | LB | RO,PASLADR | PASLA ADDRESS | MPC19630 | |
| 1B7A | D200 | 0A1C | 1964 | STB | RO,ADDRESS | CONSOLE ADDRESS | MPC19640 | |
| 1B7E | D310 | 0A2C | 1965 | LB | R1,PASLRD | PASLA READ | MPC19650 | |
| 1B82 | D210 | 0A30 | 1966 | STB | R1,RDCMD | COMMAND | MPC19660 | |
| 1B86 | DE00 | 0A1E | 1967 | OC | RO,PASLCMD | | MPC19670 | |
| 1B8A | 2401 | | 1968 | LIS | RO,1 | SET PASLA FLAG | MPC19680 | |
| 1B8C | 4000 | 0A24 | 1969 | STH | RO,PASFLG | | MPC19690 | |
| 1B90 | 4300 | 1BB2 | 1970 | B | DEVCHK1 | | MPC19700 | |
| | | | 1971 | * | | | MPC19710 | |
| 1B94 | D300 | 0A2E | 1972 | DEVK5 | LB | RO,MICWRT | MPC19720 | |
| 1B98 | D200 | 0A22 | 1973 | STB | RO,WRTCMD | | MPC19730 | |
| 1B9C | D300 | 0A1A | 1974 | LB | RO,MICADR | | MPC19740 | |
| 1BA0 | D200 | 0A1C | 1975 | STB | RO,ADDRESS | | MPC19750 | |
| 1BA4 | D300 | 0A2D | 1976 | LB | RO,MICRD | | MPC19760 | |
| 1BA8 | D200 | 0A30 | 1977 | STB | RO,RDCMD | | MPC19770 | |
| 1BAC | 2411 | | 1978 | LIS | R1,1 | | MPC19780 | |
| 1BAE | 4010 | 0A26 | 1979 | STH | R1,MICFLAG | | MPC19790 | |
| 1BB2 | D300 | 0A11 | 1980 | DEVCHK1 | LB | RO,IO+1 | GET LIST DEVICE TYPE | MPC19800 |
| 1BB6 | 0800 | | 1981 | LHR | RO,RO | | MPC19810 | |
| 1BB8 | 4330 | 0A3E | 1982 | BZ | EXEC | NO SEPARATE LIST DEVICE | MPC19820 | |
| 1BBC | C500 | 0001 | 1983 | CLHI | RO,1 | PASLA TYPE ? | MPC19830 | |
| 1BC0 | 4330 | 1BE8 | 1984 | BE | DEVK2 | YES | MPC19840 | |
| 1BC4 | C500 | 0003 | 1985 | CLHI | RO,3 | LINE PRINTER TYPE | MPC19850 | |
| 1BC8 | 4330 | 1C00 | 1986 | BE | DEVK3 | YES | MPC19860 | |
| 1BCC | C500 | 0005 | 1987 | CLHI | RO,5 | MICRO I-0 BUS | MPC19870 | |
| 1BD0 | 4330 | 1C14 | 1988 | BE | DEVK5.1 | | MPC19880 | |
| 1BD4 | D300 | 0A31 | 1989 | LB | RO,TTYWRT | ELSE TTY TYPE | MPC19890 | |
| 1BD8 | D200 | 0A23 | 1990 | STB | RO,WRTCMD+1 | WRITE COMMAND | MPC19900 | |
| 1BDC | D300 | 0A13 | 1991 | LB | RO,TTYADR+1 | 2 TTY ADDR | MPC19910 | |
| 1BE0 | D200 | 0A1D | 1992 | STB | RO,ADDRESS+1 | LIST ADDRESS | MPC19920 | |
| 1BE4 | 4300 | 0A3E | 1993 | B | EXEC | BEGIN EXECUTION | MPC19930 | |
| 1BE8 | D300 | 0A2B | 1994 | DEVK2 | LB | RO,PASLWRT | PASLA WRITE | MPC19940 |
| 1BEC | D200 | 0A23 | 1995 | STB | RO,WRTCMD+1 | LIST COMMAND | MPC19950 | |
| 1BF0 | D300 | 0A16 | 1996 | LB | RO,PASLADR2 | 2 PASLA ADDRESS | MPC19960 | |
| 1BF4 | D200 | 0A1D | 1997 | STB | RO,ADDRESS+1 | LIST ADDRESS | MPC19970 | |
| 1BF8 | DE00 | 0A1F | 1998 | OC | RO,PASLCND+1 | ISSUE PASLA COMMAND | MPC19980 | |
| 1BFC | 4300 | 0A3E | 1999 | B | EXEC | BEGIN EXECUTION | MPC19990 | |
| 1C00 | D300 | 0A2A | 2000 | DEVK3 | LB | RO,LNPWRT | LINE PRINTER | MPC20000 |
| 1C04 | D200 | 0A23 | 2001 | STB | RO,WRTCMD+1 | WRITE COMMAND | MPC20010 | |
| 1C08 | D300 | 0A18 | 2002 | LB | RO,LNPADR | LP ADDRESS | MPC20020 | |
| 1C0C | D200 | 0A1D | 2003 | STB | RO,ADDRESS+1 | LIST ADDRESS | MPC20030 | |
| 1C10 | 4300 | 0A3E | 2004 | B | EXEC | BEGIN EXECUTION | MPC20040 | |
| | | | 2005 | * | | | MPC20050 | |
| 1C14 | D300 | 0A2E | 2006 | DEVK5.1 | LB | RO,MICWRT | MPC20060 | |
| 1C18 | D200 | 0A23 | 2007 | STB | RO,WRTCMD+1 | | MPC20070 | |

SUBROUTINES

| | | | | | |
|------|-----------|------|---------|-----------------------------------|----------|
| 1C1C | D300 0A1B | 2008 | LB | RO,MICADR+1 | MPC20080 |
| 1C20 | D200 0A1D | 2009 | STB | RO,ADDRESS+1 | MPC20090 |
| 1C24 | 4300 0A3E | 2010 | B | EXEC | MPC20100 |
| | | 2011 | ***** | | |
| | | 2012 | * | | MPC20110 |
| | | 2013 | * | SERVICE POINTER ERROR SUBROUTINES | MPC20120 |
| | | 2014 | * | | MPC20130 |
| | | 2015 | * | | MPC20140 |
| | | 2016 | * | | MPC20150 |
| 1C28 | D000 223C | 2017 | SVCERR | STM RO,SSAVE | MPC20160 |
| 1C2C | C800 0096 | 2018 | LHI | RO,X'96' | MPC20170 |
| 1C30 | 4000 1C84 | 2019 | STH | RO,PSW+2 | MPC20180 |
| 1C34 | C800 4630 | 2020 | LHI | RO,X'4630' | MPC20190 |
| 1C38 | 4300 1C72 | 2021 | B | COMRTN | MPC20200 |
| | | 2022 | * | | MPC20210 |
| 1C3C | D000 223C | 2023 | FIXPT | STM RO,SSAVE | MPC20220 |
| 1C40 | C800 0048 | 2024 | LHI | RO,X'48' | MPC20230 |
| 1C44 | 4000 1C84 | 2025 | STH | RO,PSW+2 | MPC20240 |
| 1C48 | C800 4631 | 2026 | LHI | RO,X'4631' | MPC20250 |
| 1C4C | 4300 1C72 | 2027 | B | COMRTN | MPC20260 |
| | | 2028 | * | | MPC20270 |
| 1C50 | D000 223C | 2029 | FLPT | STM RO,SSAVE | MPC20280 |
| 1C54 | C800 0028 | 2030 | LHI | RO,X'28' | MPC20290 |
| 1C58 | 4000 1C84 | 2031 | STH | RO,PSW+2 | MPC20300 |
| 1C5C | C800 4633 | 2032 | LHI | RO,X'4633' | MPC20310 |
| 1C60 | 2309 | 2033 | BS | COMRTN | MPC20320 |
| | | 2034 | * | | MPC20330 |
| 1C62 | D000 223C | 2035 | SYSQ | STM RO,SSAVE | MPC20340 |
| 1C66 | C800 008C | 2036 | LHI | RO,X'8C' | MPC20350 |
| 1C6A | 4000 1C84 | 2037 | STH | RO,PSW+2 | MPC20360 |
| 1C6E | C800 4632 | 2038 | LHI | RO,X'4632' | MPC20370 |
| | | 2039 | * | | MPC20380 |
| 1C72 | 4000 20AA | 2040 | COMRTN | STH RO,ERRNUM | MPC20390 |
| 1C76 | 41F0 1EA4 | 2041 | BAL | R15,PRINT | MPC20400 |
| 1C7A | 20A0 | 2042 | DC | Z(ERRMSG) | MPC20410 |
| 1C7C | 20AD | 2043 | DC | Z(END) | MPC20420 |
| 1C7E | D100 223C | 2044 | LM | RO,SSAVE | MPC20430 |
| 1C82 | C200 0096 | 2045 | PSW | LPSW X'96' | MPC20440 |
| | | 2046 | * | | MPC20450 |
| 1C86 | | 2047 | BADR15 | DS 2 | MPC20460 |
| | | 2048 | * | | MPC20470 |
| | | 2049 | * | | MPC20480 |
| 1C88 | 0000 | 2050 | EXTINT1 | DC 0 | MPC20490 |
| 1C8A | 0000 | 2051 | DC | 0 | MPC20500 |
| 1C8C | 20F0 | 2052 | DC | X'20F0' | MPC20510 |
| 1C8E | 4810 20A8 | 2053 | LH | R1,TESTNUM | MPC20520 |
| 1C92 | 4010 2170 | 2054 | STH | R1,AUTOMSG1 | MPC20530 |
| 1C96 | 41F0 1EA4 | 2055 | BAL | R15,PRINT | MPC20540 |
| 1C9A | 216E | 2056 | DC | Z(AUTOMSG) | MPC20550 |
| 1C9C | 2173 | 2057 | DC | Z(AUTOEND) | MPC20560 |
| 1C9E | C200 1C88 | 2058 | LPSW | EXTINT1 | MPC20570 |
| | | 2059 | ***** | | |
| | | 2060 | * | | MPC20580 |
| | | | | | MPC20590 |
| | | | | | MPC20600 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|---------|--|------------------------------------|----------|
| | | 2061 | *ILGINT | ILLEGAL INSTRUCTION INTERRUPT ROUTINE | | NPC20610 |
| | | 2062 | * | | | NPC20620 |
| | | 2063 | * | THERE ARE TWO TYPES OF ILLEGAL INSTRUCTION INTERRUPTS: | | NPC20630 |
| | | 2064 | * | (1) A BONAFIDE INTERRUPT CAUSED BY AN ACTUAL ILLEGAL | | NPC20640 |
| | | 2065 | * | INSTRUCTION. (FLAG "INCQR" NOT= 3OR4).(2) OR AN | | NPC20650 |
| | | 2066 | * | INTERRUPT CREATED BY A EXECUTE VIOLATION. | | NPC20660 |
| | | 2067 | * | | | NPC20670 |
| | | 2068 | * | IN THE CASE OF THE EXECUTE VIOLATION, THE STATUS OF | | NPC20680 |
| | | 2069 | * | THE CONTROLLER IS SENSED, THE RESULTING I/O INTERRUPT IS | | NPC20690 |
| | | 2070 | * | ACKNOWLEDGED, AND THE EXECUTE INTERRUPT COUNTER IS | | NPC20700 |
| | | 2071 | * | INCREMENTED. | | NPC20710 |
| | | 2072 | * | NOTE: IN THE CASE OF THE SAME BLOCK BEING WRITE-READ & | | NPC20720 |
| | | 2073 | * | EXECUTE PROTECT AND THE CONTROLLER IN THE WRITE-READ | | NPC20730 |
| | | 2074 | * | PROTECT MODE AN "E4" STATUS IS EXPECTED, OTHERWISE "A4" | | NPC20740 |
| | | 2075 | * | IS EXPECTED. | | NPC20750 |
| | | 2076 | * | | | NPC20760 |
| | | 2077 | * | THERE ARE TWO TYPES OF EXECUTE VIOLATIONS: | | NPC20770 |
| | | 2078 | * | (1) INCQR = 3 A NON-EXPECTED VIOLATION. THE RETURN | | NPC20780 |
| | | 2079 | * | IS MADE TO THE NEXT LINE IN THE PROGRAM | | NPC20790 |
| | | 2080 | * | | | NPC20800 |
| | | 2081 | * | (2) INCQR = 4 AN EXPECTED VIOLATION. THE RETURN IS | | NPC20810 |
| | | 2082 | * | MADE TO R15 | | NPC20820 |
| | | 2083 | * | | | NPC20830 |
| | | 2084 | * | REGISTERS R10-15 ARE SAVED. | | NPC20840 |
| | | 2085 | * | | | NPC20850 |
| | | 2086 | * | | | NPC20860 |
| 1CA2 | C590 0003 | 2087 | ILGINT | CLHI INCQR,3 | EXPECTED ILLEGAL INTERRUPT | NPC20870 |
| 1CA6 | 2185 | 2088 | | BLS CONTO | NO, ACTUAL ILLEGAL INS. | NPC20880 |
| 1CA8 | C590 0005 | 2089 | | CLHI INCQR,5 | TEST SECOND PARAMETER | NPC20890 |
| 1CAC | 4280 1CE0 | 2090 | | BL ILGZ | TEST FOR EXECUTE PROTECT VIOLATION | NPC20900 |
| 1CB0 | 4810 0030 | 2091 | CONTO | LH R1,X'30' | LOAD DATA TO BE CONVERTED | NPC20910 |
| 1CB4 | 41E0 1DF2 | 2092 | | BAL R14,CONVERT | CONVERT TO ASCII CHARACTERS | NPC20920 |
| 1CB8 | 000C | 2093 | | DC X'C' | | NPC20930 |
| 1CBA | 2142 | 2094 | | DC Z(ADRS00) | | NPC20940 |
| 1CBC | 4810 0032 | 2095 | | LH R1,X'32' | LOAD DATA TO BE CONVERTED | NPC20950 |
| 1CC0 | 41E0 1DF2 | 2096 | | BAL R14,CONVERT | CONVERT TO ASCII CHARACTERS | NPC20960 |
| 1CC4 | 000C | 2097 | | DC X'C' | | NPC20970 |
| 1CC6 | 2148 | 2098 | | DC Z(ADRS0) | | NPC20980 |
| 1CC8 | 41F0 1EA4 | 2099 | | BAL R15,PRINT | PRINT ILLEGAL INSTRUCTION MESSAGE | NPC20990 |
| 1CCC | 212A | 2100 | | DC Z(ILGHSG) | | NPC21000 |
| 1CCE | 214D | 2101 | | DC Z(ILGEND) | | NPC21010 |
| 1CD0 | 9DBA | 2102 | | SSR R11,R10 | IS TTY OFF ? | NPC21020 |
| 1CD2 | 2315 | 2103 | | BNMS CONT14 | NO, LOAD NEW PSW | NPC21030 |
| 1CD4 | C870 5555 | 2104 | | LHI R7,X'5555' | YES, WRITE TO DISPLAY PANEL | NPC21040 |
| 1CD8 | 41E0 1DE0 | 2105 | | BAL R14,WRITE | | NPC21050 |
| 1CDC | C200 206A | 2106 | CONT14 | LPSW HALT | LOADS NEW PSW AND HALT | NPC21060 |
| 1CE0 | DOAO 222C | 2107 | ILGZ | STM R10,SAVEJ | SAVE REGISTERS | NPC21070 |
| 1CE4 | 4800 0C30 | 2108 | | LH R0,PRTADR | GET PROTECT DEVICE ADDRESS | NPC21080 |
| 1CE8 | 9DOF | 2109 | | SSR R0,R15 | SENSE STATUS | NPC21090 |
| 1CEA | 9FAE | 2110 | | AIR R10,R14 | ACKNOWLEDGE I-O INTERRUPT | NPC21100 |
| 1CEC | 050A | 2111 | | CLHR R0,R10 | MEMORY PROTECT CONTROLLER | NPC21110 |
| 1CEE | 4230 1FCA | 2112 | | BNE ERR14 | ELSE ERROR | NPC21120 |
| 1CF2 | 2681 | 2113 | | AIS INTCE,1 | INCREMENT EXECUTE INT COUNT | NPC21130 |

SUBROUTINES

| | | | | | | | |
|------|------|------|------|---------|-----------------------------|-------------------------------------|----------|
| 1CF4 | 48C0 | 0030 | 2114 | LH | R12,X'30' | GET OLD PSW 0-15 | MPC21140 |
| 1CF8 | C4C0 | 00F0 | 2115 | NHI | R12,X'00F0' | GET ONLY EXT MEMORY BITS 8-11 | MPC21150 |
| 1CFC | 95BC | | 2116 | EPSR | R11,R12 | USE THESE BITS IN NEW PSW | MPC21160 |
| 1CFE | 48C0 | 226A | 2117 | LH | R12,BLKPT | GET EX PROTECT STATUS | MPC21170 |
| 1D02 | 4800 | 2268 | 2118 | LH | R0,BLKPT | AND W-R PROTECT STATUS OF BLOCK | MPC21180 |
| 1D06 | 24D0 | | 2119 | LIS | R13,0 | ZERO | MPC21190 |
| 1D08 | 95BD | | 2120 | EPSR | R11,R13 | NEW PSW 0-15 | MPC21200 |
| 1D0A | 08CC | | 2121 | LHR | R12,R12 | BLOCK EXECUTE PROTECTED | MPC21210 |
| 1D0C | 4330 | 1FFC | 2122 | BZ | ERR19 | ELSE ERROR | MPC21220 |
| 1D10 | 0800 | | 2123 | LHR | R0,R0 | BLOCK ALSO W-R PROTECTED? | MPC21230 |
| 1D12 | 2338 | | 2124 | BZS | ILGZ0 | NO | MPC21240 |
| 1D14 | 0822 | | 2125 | LHR | R2,R2 | YES, THEN TEST FOR W-R PROTECT MODE | MPC21250 |
| 1D16 | 2336 | | 2126 | BZS | ILGZ0 | WRITE PROTECT MODE ONLY | MPC21260 |
| 1D18 | C5F0 | 00E4 | 2127 | CLHI | R15,X'E4' | THEN EXECUTE & READ VIOL | MPC21270 |
| 1D1C | 4230 | 1FAE | 2128 | BNE | ERR11 | ELSE ERROR | MPC21280 |
| 1D20 | 2305 | | 2129 | BS | ILGZ1 | CONTINUE | MPC21290 |
| 1D22 | C5F0 | 00A4 | 2130 | ILGZ0 | CLHI R15,X'A4' | THEN EXECUTE VIOLATION ONLY | MPC21300 |
| 1D26 | 4230 | 1FB6 | 2131 | BNE | ERR12 | ELSE ERROR | MPC21310 |
| 1D2A | 05FE | | 2132 | ILGZ1 | CLHR R15,R14 | STATUS IS THE SAME | MPC21320 |
| 1D2C | 4230 | 2030 | 2133 | BNE | ERR25 | ELSE ERROR | MPC21330 |
| 1D30 | C580 | 0010 | 2134 | CLHI | INTCE,16 | NUMBER OF EXECUTE VIOL. < 16 | MPC21340 |
| 1D34 | 4380 | 2004 | 2135 | BNL | ERR20 | ELSE ERROR | MPC21350 |
| 1D38 | C990 | 0004 | 2136 | CHI | INCQR,4 | EXPECTED EXECUTE PROTECT VIOL | MPC21360 |
| 1D3C | 2335 | | 2137 | BES | ILGZ2 | YES | MPC21370 |
| 1D3E | D1A0 | 222C | 2138 | LM | R10,SAVEJ | NO, RESTORE REGISTERS | MPC21380 |
| 1D42 | C200 | 0030 | 2139 | LPSW | X'30' | RETURN TO NEXT LINE OF TEST | MPC21390 |
| 1D46 | D1A0 | 222C | 2140 | ILGZ2 | LM R10,SAVEJ | RESTORE REGISTERS | MPC21400 |
| 1D4A | 40F0 | 0032 | 2141 | STH | R15,X'32' | R15 RETURN | MPC21410 |
| 1D4E | C200 | 0030 | 2142 | LPSW | X'30' | RETURN | MPC21420 |
| | | | 2143 | ***** | | | MPC21430 |
| | | | 2144 | * | | | MPC21440 |
| | | | 2145 | *MALFTN | MACHINE MALFUNCTION ROUTINE | | MPC21450 |
| | | | 2146 | * | | | MPC21460 |
| | | | 2147 | * | | | MPC21470 |
| 1D52 | 9511 | | 2148 | MALFTN | EPSR R1,R1 | | MPC21480 |
| 1D54 | 24C1 | | 2149 | | LIS R12,1 | | MPC21490 |
| 1D56 | 04C1 | | 2150 | | NHR R12,R1 | | MPC21500 |
| 1D58 | 23F5 | | 2151 | | BFFS X'F',5 | | MPC21510 |
| 1D5A | 4890 | 0024 | 2152 | | LH R9,X'24' | | MPC21520 |
| 1D5E | 4300 | 1D92 | 2153 | | B CONT16 | | MPC21530 |
| 1D62 | 0811 | | 2154 | CONT13 | LHR R1,R1 | | MPC21540 |
| 1D64 | 2133 | | 2155 | | BNZS CONT15 | | MPC21550 |
| 1D66 | 4090 | 0024 | 2156 | | STH R9,X'24' | | MPC21560 |
| 1D6A | 41E0 | 1DF2 | 2157 | CONT15 | BAL R14,CONVERT | | MPC21570 |
| 1D6E | 0000 | | 2158 | | DC X'0' | | MPC21580 |
| 1D70 | 2166 | | 2159 | | DC Z(CCADRS) | | MPC21590 |
| 1D72 | 4810 | 0024 | 2160 | | LH R1,X'24' | | MPC21600 |
| 1D76 | 41E0 | 1DF2 | 2161 | | BAL R14,CONVERT | | MPC21610 |
| 1D7A | 000C | | 2162 | | DC X'C' | | MPC21620 |
| 1D7C | 2168 | | 2163 | | DC Z(MMADRS) | | MPC21630 |
| 1D7E | 41F0 | 1EA4 | 2164 | | BAL R15,PRINT | | MPC21640 |
| 1D82 | 214E | | 2165 | | DC Z(MACHMAL) | | MPC21650 |
| 1D84 | 216D | | 2166 | | DC Z(MMEND) | | MPC21660 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|-------------------|---------------------------------------|----------------------------------|----------|
| 1D86 | 9DBA | 2167 | SSR | R11,R10 | | NPC21670 |
| 1D88 | 2315 | 2168 | BNMS | CONT16 | | NPC21680 |
| 1D8A | C870 AAAA | 2169 | LHI | R7,X'AAAA' | | NPC21690 |
| 1D8E | 41E0 1DE0 | 2170 | BAL | R14,WRITE | | NPC21700 |
| 1D92 | C200 206A | 2171 | CONT16 | LPSW HALT | | NPC21710 |
| | | 2172 | ***** | | | NPC21720 |
| | | 2173 | * | | | NPC21730 |
| | | 2174 | *TSTNUM | INITIALIZATION ROUTINE FOR ALL TEST. | | NPC21740 |
| | | 2175 | * | | | NPC21750 |
| | | 2176 | * | | | NPC21760 |
| 1D96 | D310 0A2F | 2177 | TSTNUM | LB R1,SUBTST | GET TEST NUMBER | NPC21770 |
| 1D9A | 2611 | 2178 | AIS | R1,1 | ADJUST VALUE | NPC21780 |
| 1D9C | E100 0000 | 2179 | TSTN1 | SVC 0,0 | CLEAR PSW 7 | NPC21790 |
| 1DA0 | 41E0 1DF2 | 2180 | BAL | R14,CONVERT | CONVERT TEST NUMBER FOR DISPLAY | NPC21800 |
| 1DA4 | 0004 | 2181 | DC | X'4' | | NPC21810 |
| 1DA6 | 20A8 | 2182 | DC | Z(TESTNUM) | | NPC21820 |
| 1DA8 | 4810 20A8 | 2183 | LH | R1,TESTNUM | GET CONVERTED VALUE | NPC21830 |
| 1DAC | 4010 20FA | 2184 | STH | R1,VALUE | STORE IT | NPC21840 |
| 1DB0 | 2711 | 2185 | SIS | R1,1 | ADJUST VALUE | NPC21850 |
| 1DB2 | 08EF | 2186 | LHR | R14,R15 | SAVE RETURN ADDRESS | NPC21860 |
| 1DB4 | 48F0 0C20 | 2187 | LH | R15,NOMSG | NO MESSAGES? | NPC21870 |
| 1DB8 | 2135 | 2188 | BNZS | TSTN2 | YES | NPC21880 |
| 1DBA | 41F0 1EA4 | 2189 | BAL | R15,PRINT | NO, THEN DISPLAY TEST NO. | NPC21890 |
| 1DBE | 20F2 | 2190 | DC | Z(TESTMSG) | | NPC21900 |
| 1DC0 | 20FD | 2191 | DC | Z(TESTED) | | NPC21910 |
| 1DC2 | 9F0F | 2192 | TSTN2 | AIR R0,R15 | ACKNOWLEDGE ANY INTERRUPTS | NPC21920 |
| 1DC4 | 48F0 219A | 2193 | LH | R15,BLOCKMZ | SET BLOCK PARAMETERS | NPC21930 |
| 1DC8 | 40F0 2264 | 2194 | STH | R15,BLOCKMAX | | NPC21940 |
| 1DCC | 4850 219C | 2195 | LH | BLK,BLOCKREF | | NPC21950 |
| 1DD0 | 4050 2260 | 2196 | STH | BLK,BLOCK2 | | NPC21960 |
| 1DD4 | 2490 | 2197 | LIS | INCQR,0 | ILLEGAL INSTRUCTION | NPC21970 |
| 1DD6 | 2420 | 2198 | LIS | R2,0 | WRITE PROTECT MODE | NPC21980 |
| 1DD8 | 24A0 | 2199 | LIS | EXPT,0 | NO EXECUTE PROTECTION | NPC21990 |
| 1DDA | 40A0 2274 | 2200 | STH | EXPT,PROTPATT | ZERO PROTECT PATTERN | NPC22000 |
| 1DDE | 030E | 2201 | BR | R14 | RETURN | NPC22010 |
| | | 2202 | ***** | | | NPC22020 |
| | | 2203 | * | | | NPC22030 |
| | | 2204 | *WRITE | DISPLAY CHARACTER ON CONSOLE | | NPC22040 |
| | | 2205 | * | | | NPC22050 |
| | | 2206 | * | | | NPC22060 |
| 1DE0 | 24D1 | 2207 | WRITE | LIS R13,1 | PUT DISPLAY IN INCREMENTAL MODE | NPC22070 |
| 1DE2 | DED0 0A34 | 2208 | OC | R13,INCRMT | LOAD CONTENTS OF R7 INTO R12 AND | NPC22080 |
| 1DE6 | 08C7 | 2209 | LHR | R12,R7 | WRITE VALUE ON DISPLAY PANEL | NPC22090 |
| 1DE8 | 94CC | 2210 | EXBR | R12,R12 | | NPC22100 |
| 1DEA | 98DC | 2211 | WHR | R13,R12 | | NPC22110 |
| 1DEC | DED0 0A33 | 2212 | OC | R13,NORM | | NPC22120 |
| 1DF0 | 030E | 2213 | BR | R14 | | NPC22130 |
| | | 2214 | ***** | | | NPC22140 |
| | | 2215 | * | | | NPC22150 |
| | | 2216 | * | | | NPC22160 |
| | | 2217 | * CONVERT ROUTINE | R1 = DATA TO BE CONVERTED TO ASCII | | NPC22170 |
| | | 2218 | * | R10 = ADRS WHERE DATA IS TO BE STORED | | NPC22180 |
| | | 2219 | * | R12 = SHIFT VALUE | | NPC22190 |

SUBROUTINES

| | | | | | | | |
|------|------|------|----------|------|------------|------------------------------------|----------|
| | | 2220 | * | | | | MPC22200 |
| 1DF2 | DOAO | 2222 | CONVERT | STM | R10,SAVEG | | MPC22210 |
| 1DF6 | 48CE | 0000 | | LH | R12,0(R14) | | MPC22220 |
| 1DFA | 48AE | 0002 | | LH | R10,2(R14) | | MPC22230 |
| 1DFE | 08E1 | | CONVERT1 | LHR | R11,R1 | LOAD DATA TO BE CONVERTED | MPC22240 |
| 1E00 | CCBC | 0000 | | SRHL | R11,0(R12) | SHIFT HEX DIGIT TO BE CONVERTED | MPC22250 |
| 1E04 | C4B0 | 000F | | NHI | R11,X'F' | ISOLATE HEX DIGIT | MPC22260 |
| 1E08 | C6B0 | 0030 | | OHI | R11,X'30' | CONVERT TO ASCII NUMBER | MPC22270 |
| 1E0C | C5B0 | 003A | | CLHI | R11,X'3A' | IS IT A VALID NUMBER ? | MPC22280 |
| 1E10 | 2182 | | | BLS | CONT | YES, CONTINUE | MPC22290 |
| 1E12 | 26E7 | | | AIS | R11,7 | NO, CONVERT TO ASCII LETTER | MPC22300 |
| 1E14 | D2BA | 0000 | CONT | STB | R11,0(R10) | STORE ASCII BYTE IN MESSAGE | MPC22310 |
| 1E18 | 08CC | | | LHR | R12,R12 | HAS ENTIRE NUMBER BEEN CONVERTED ? | MPC22320 |
| 1E1A | 2335 | | | BZS | CONT1 | | MPC22330 |
| 1E1C | 27C4 | | | SIS | R12,4 | NO, DECREMENT SHIFT INDEX | MPC22340 |
| 1E1E | 26A1 | | | AIS | R10,1 | INCREMENT STORAGE INDEX | MPC22350 |
| 1E20 | 4300 | 1DFE | | B | CONVERT1 | REPEAT FOR NEXT HEX DIGIT | MPC22360 |
| 1E24 | D1A0 | 2222 | CONT1 | LM | R10,SAVEG | | MPC22370 |
| 1E28 | 430E | 0004 | | B | 4(R14) | | MPC22380 |
| | | 2239 | * | | | | MPC22390 |
| | | 2240 | ***** | | | | MPC22400 |
| | | 2241 | * | | | | MPC22410 |
| | 0000 | 1E2C | TSTBRK | EQU | * | | MPC22420 |
| 1E2C | D000 | 223C | | STM | R0,SSAVE | | MPC22430 |
| 1E30 | D310 | 0A10 | | LB | R1,I0 | GET IO DEVICE NUMBER | MPC22440 |
| 1E34 | C510 | 0001 | | CLHI | R1,1 | IS IT ON PASLA I/O | MPC22450 |
| 1E38 | 4230 | 1E56 | | BNE | TSTBRK2 | YES, BRANCH | MPC22460 |
| 1E3C | D310 | 0A1C | | LB | R1,ADDRESS | | MPC22470 |
| 1E40 | 9D12 | | | SSR | R1,R2 | | MPC22480 |
| 1E42 | 4280 | 1E50 | | BTC | 8,TSTBRK1 | NO KEY DEPRESSED=NO BREAK | MPC22490 |
| 1E46 | 9B12 | | | RDR | R1,R2 | DUMMY READ | MPC22500 |
| 1E48 | 9B12 | | | RDR | R1,R2 | READ KEY DEPRESSED | MPC22510 |
| 1E4A | 0822 | | | LHR | R2,R2 | SET CC | MPC22520 |
| 1E4C | 4330 | 0C7C | | BZ | TTYIN | ZERO=BREAK | MPC22530 |
| 1E50 | D100 | 223C | TSTBRK1 | LM | R0,SSAVE | NO BREAK | MPC22540 |
| 1E54 | 030F | | | BR | R15 | RETURN | MPC22550 |
| 1E56 | D310 | 0A1C | TSTBRK2 | LB | R1,ADDRESS | | MPC22560 |
| 1E5A | 9D12 | | | SSR | R1,R2 | | MPC22570 |
| 1E5C | C320 | 0020 | | THI | R2,X'20' | BREAK STATUS | MPC22580 |
| 1E60 | 4330 | 1E50 | | BZ | TSTBRK1 | NO BREAK | MPC22590 |
| 1E64 | 9B12 | | TSTBRK22 | RDR | R1,R2 | | MPC22600 |
| 1E66 | C830 | 7FFF | | LHI | R3,X'7FFF' | | MPC22610 |
| 1E6A | 2731 | | TSTBRK3 | SIS | R3,1 | | MPC22620 |
| 1E6C | 2730 | | | SIS | R3,0 | DEL#AY | MPC22630 |
| 1E6E | 2032 | | | BNZS | TSTBRK3 | | MPC22640 |
| 1E70 | 9D12 | | | SSR | R1,R2 | | MPC22650 |
| 1E72 | C320 | 0020 | | THI | R2,X'20' | BREAK STATUS? | MPC22660 |
| 1E76 | 4330 | 0C7C | | BZ | TTYIN | NO GO TO COMMAND MODE | MPC22670 |
| 1E7A | 4300 | 1E64 | | B | TSTBRK22 | | MPC22680 |
| | | 2269 | * | | | | MPC22690 |
| 1E7E | | 2270 | BADR16 | DS | 16 | | MPC22700 |
| | | 2271 | * | | | | MPC22710 |
| | | 2272 | ***** | | | | MPC22720 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|---------|------------------------------------|----------------------------|----------|
| | | 2273 | * | | | MPC22730 |
| | | 2274 | *GETCHR | GET CHARACTER FROM CONSOLE | | MPC22740 |
| | | 2275 | * | | | MPC22750 |
| | | 2276 | * | | | MPC22760 |
| 1E8E | 9DB0 | 2277 | GETCHR | SSR R11,R0 | * READ CHAR ROUTINE | MPC22770 |
| 1E90 | 021F | 2278 | | BMR R15 | EXIT IF TTY DU | MPC22780 |
| 1E92 | 2082 | 2279 | | BCS GETCHR | IF BUSY SENSE AGAIN | MPC22790 |
| 1E94 | 9BB0 | 2280 | | RDR R11,R0 | READ A CHARACTER | MPC22800 |
| 1E96 | 4820 0A26 | 2281 | | LH R2,MICFLAG | IS IT MICRO I/O BUS | MPC22810 |
| 1E9A | 2332 | 2282 | | BZS GETCHR1 | NO BRANCH | MPC22820 |
| 1E9C | 9AB0 | 2283 | | WDR R11,R0 | | MPC22830 |
| 1E9E | C400 007F | 2284 | GETCHR1 | NHI R0,X'7F' | MASK OF PARITY BIT | MPC22840 |
| 1EA2 | 030F | 2285 | | BR R15 | RETURN | MPC22850 |
| | | 2286 | ***** | | | MPC22860 |
| | | 2287 | * | | | MPC22870 |
| | | 2288 | *PRINT | DISPLAYS MESSAGE ON CONSOLE DEVICE | | MPC22880 |
| | | 2289 | * | | | MPC22890 |
| | | 2290 | * | | | MPC22900 |
| 1EA4 | D3B0 0A1D | 2291 | PRINT | LB R11,ADDRESS+1 | GET LIST DEVICE ADDR | MPC22910 |
| 1EA8 | D3A0 0A11 | 2292 | | LB R10,IO+1 | GET LIST DEVICE TYPE | MPC22920 |
| 1EAC | 08AA | 2293 | | LHR R10,R10 | | MPC22930 |
| 1EAE | 4330 1EFA | 2294 | | BZ PRINT1 | NO SEPARATE LIST DEVICE | MPC22940 |
| 1EB2 | D3C0 0A1C | 2295 | | LB R12,ADDRESS | GET CONSOLE ADDR | MPC22950 |
| 1EB6 | 05BC | 2296 | | CLHR R11,R12 | | MPC22960 |
| 1EB8 | 2136 | 2297 | | BNES PRT0 | SHOULD NOT = LIST ADDR | MPC22970 |
| 1EBA | 24C0 | 2298 | | LIS R12,0 | ZERO | MPC22980 |
| 1EBC | D2C0 0A11 | 2299 | | STB R12,IO+1 | LIST DEVICE | MPC22990 |
| 1EC0 | 4300 1EFA | 2300 | | B PRINT1 | CONTINUE | MPC23000 |
| 1EC4 | C5A0 0001 | 2301 | PRT0 | CLHI R10,1 | PASLA TYPE | MPC23010 |
| 1EC8 | 2132 | 2302 | | BNES PRT1 | NO, TTY TYPE | MPC23020 |
| 1ECA | 26E1 | 2303 | | AIS R11,1 | TRANS ADDR | MPC23030 |
| 1ECC | DEB0 0A23 | 2304 | PRT1 | OC R11,WRTCMD+1 | WRITE COMMAND | MPC23040 |
| 1ED0 | 9DBA | 2305 | PRT2 | SSR R11,R10 | SENSE STATUS | MPC23050 |
| 1ED2 | 2313 | 2306 | | BNMS PRT3 | EXISTS ? | MPC23060 |
| 1ED4 | 4300 1EFA | 2307 | | B PRINT1 | CONSOLE DEVICE NEXT | MPC23070 |
| 1ED8 | 2084 | 2308 | PRT3 | BCS PRT2 | WAIT UNTIL NOT BUSY | MPC23080 |
| 1EDA | 48CF 0000 | 2309 | | LH R12,0(R15) | GET START OF MESS | MPC23090 |
| 1EDE | 48CF 0002 | 2310 | | LH R13,2(R15) | GET END OF MESS | MPC23100 |
| 1EE2 | 96BC | 2311 | | WBR R11,R12 | WRITE MESS | MPC23110 |
| 1EE4 | 9DBA | 2312 | | SSR R11,R10 | SENSE STATUS | MPC23120 |
| 1EE6 | 2081 | 2313 | | BTBS 8,1 | | MPC23130 |
| 1EE8 | D3A0 0A11 | 2314 | | LB R10,IO+1 | GET LIST DEVICE TYPE AGAIN | MPC23140 |
| 1EEC | C5A0 0001 | 2315 | | CLHI R10,1 | PASLA TYPE | MPC23150 |
| 1EFO | 2135 | 2316 | | BNES PRINT1 | ELSE CONSOLE DEVICE NEXT | MPC23160 |
| 1EF2 | 24A0 | 2317 | | LIS R10,0 | | MPC23170 |
| 1EF4 | 9ABA | 2318 | | WDR R11,R10 | WRITE ZERO | MPC23180 |
| 1EF6 | 9DBA | 2319 | | SSR R11,R10 | | MPC23190 |
| 1EF8 | 2081 | 2320 | | BTBS 8,1 | | MPC23200 |
| | | 2321 | * | | | MPC23210 |
| 1EFA | D3B0 0A1C | 2322 | PRINT1 | LB R11,ADDRESS | | MPC23220 |
| 1EFE | D3A0 0A10 | 2323 | | LB R10,IO | GET CONSOLE DEVICE TYPE | MPC23230 |
| 1F02 | C5A0 0001 | 2324 | | CLHI R10,1 | | MPC23240 |
| 1F06 | 2132 | 2325 | | BNES PRT4 | | MPC23250 |

SUBROUTINES

| | | | | | | | |
|------|-----------|------|--------|------------|-------------|----------|----------|
| 1F08 | 26B1 | 2326 | AIS | R11,1 | | MPC23260 | |
| 1F0A | DEBO OA22 | 2327 | PRT4 | OC | R11,WRTCHD | MPC23270 | |
| 1F0E | 9DBA | 2328 | PRT5 | SSR | R11,R10 | MPC23280 | |
| 1F10 | 2315 | 2329 | | BNMS | PRT6 | MPC23290 | |
| 1F12 | D2B0 OA28 | 2330 | | STB | R11,TTYFLG | MPC23300 | |
| 1F16 | 430F 0004 | 2331 | | B | 4(R15) | MPC23310 | |
| 1F1A | 2086 | 2332 | PRT6 | BCS | PRT5 | MPC23320 | |
| 1F1C | 48CF 0000 | 2333 | | LH | R12,0(R15) | MPC23330 | |
| 1F20 | 48DF 0002 | 2334 | | LH | R13,2(R15) | MPC23340 | |
| 1F24 | 96BC | 2335 | | WBR | R11,R12 | MPC23350 | |
| 1F26 | 9DBA | 2336 | | SSR | R11,R10 | MPC23360 | |
| 1F28 | 2081 | 2337 | | BTBS | 8,1 | MPC23370 | |
| 1F2A | 40F0 2286 | 2338 | | STH | R15,LINKSAV | MPC23380 | |
| 1F2E | D3A0 OA10 | 2339 | | LB | R10,IO | MPC23390 | |
| 1F32 | C5A0 0001 | 2340 | | CLHI | R10,1 | MPC23400 | |
| 1F36 | 4330 1F4A | 2341 | | BE | PRT7 | MPC23410 | |
| 1F3A | DEBO OA30 | 2342 | RETURN | OC | R11,RDCMD | MPC23420 | |
| 1F3E | 41F0 1E2C | 2343 | | BAL | R15,TSTBRK | MPC23430 | |
| 1F42 | 48F0 2286 | 2344 | | LH | R15,LINKSAV | MPC23440 | |
| 1F46 | 430F 0004 | 2345 | | B | 4(R15) | MPC23450 | |
| 1F4A | C8A0 00FF | 2346 | PRT7 | LHI | R10,X'FF' | MPC23460 | |
| 1F4E | 9ABA | 2347 | | WDR | R11,R10 | MPC23470 | |
| 1F50 | 9DBA | 2348 | | SSR | R11,R10 | MPC23480 | |
| 1F52 | 2081 | 2349 | | BTBS | 8,1 | MPC23490 | |
| 1F54 | 27B1 | 2350 | | SIS | R11,1 | MPC23500 | |
| 1F56 | 4300 1F3A | 2351 | | B | RETURN | MPC23510 | |
| | | 2352 | ***** | | | | MPC23520 |
| | | 2353 | * | | | MPC23530 | |
| | | 2354 | * | ERROR LIST | | MPC23540 | |
| | | 2355 | * | | | MPC23550 | |
| | | 2356 | * | | | MPC23560 | |
| 1F5A | C810 3031 | 2357 | ERR1 | LHI | R1,X'3031' | MPC23570 | |
| 1F5E | 4300 1A70 | 2358 | | B | PRTREGS | MPC23580 | |
| 1F62 | C810 3032 | 2359 | ERR2 | LHI | R1,X'3032' | MPC23590 | |
| 1F66 | 4300 1A70 | 2360 | | B | PRTREGS | MPC23600 | |
| 1F6A | C810 3033 | 2361 | ERR3 | LHI | R1,X'3033' | MPC23610 | |
| 1F6E | 4300 1A70 | 2362 | | B | PRTREGS | MPC23620 | |
| 1F72 | C810 3034 | 2363 | ERR4 | LHI | R1,X'3034' | MPC23630 | |
| 1F76 | 4300 1A70 | 2364 | | B | PRTREGS | MPC23640 | |
| 1F7A | C810 3035 | 2365 | ERR5 | LHI | R1,X'3035' | MPC23650 | |
| 1F7E | 4300 1A70 | 2366 | | B | PRTREGS | MPC23660 | |
| 1F82 | C810 3036 | 2367 | ERR6 | LHI | R1,X'3036' | MPC23670 | |
| 1F86 | 4300 1A70 | 2368 | | B | PRTREGS | MPC23680 | |
| 1F8A | C810 3037 | 2369 | ERR7 | LHI | R1,X'3037' | MPC23690 | |
| 1F8E | 4300 1A2E | 2370 | | B | PRTSTAT | MPC23700 | |
| 1F92 | C810 3038 | 2371 | ERR8 | LHI | R1,X'3038' | MPC23710 | |
| 1F96 | 4300 1A2E | 2372 | | B | PRTSTAT | MPC23720 | |
| 1F9A | C810 3039 | 2373 | ERR9 | LHI | R1,X'3039' | MPC23730 | |
| 1F9E | 4300 1A2E | 2374 | | B | PRTSTAT | MPC23740 | |
| 1FA2 | C810 3130 | 2375 | ERR10 | LHI | R1,X'3130' | MPC23750 | |
| 1FA6 | 080A | 2376 | | LHR | R0,R10 | MPC23760 | |
| 1FA8 | 08FE | 2377 | | LHR | R15,R14 | MPC23770 | |
| 1FAA | 4300 1A2E | 2378 | | B | PRTSTAT | MPC23780 | |

SUBROUTINES

| | | | | | | |
|------|-----------|------|---------|--|------------------------|----------|
| 1FAE | C810 3131 | 2379 | ERR11 | LHI | R1,X'3131' | NPC23790 |
| 1FB2 | 4300 1A2E | 2380 | | B | PRTSTAT | NPC23800 |
| 1FB6 | C810 3132 | 2381 | ERR12 | LHI | R1,X'3132' | NPC23810 |
| 1FBA | 4300 1A2E | 2382 | | B | PRTSTAT | NPC23820 |
| 1FBE | C810 3133 | 2383 | ERR13 | LHI | R1,X'3133' | NPC23830 |
| 1FC2 | 080A | 2384 | | LHR | RO,R10 | NPC23840 |
| 1FC4 | 08FE | 2385 | | LHR | R15,R14 | NPC23850 |
| 1FC6 | 4300 1A2E | 2386 | | B | PRTSTAT | NPC23860 |
| 1FCA | C810 3134 | 2387 | ERR14 | LHI | R1,X'3134' | NPC23870 |
| 1FCE | 080A | 2388 | | LHR | RO,R10 | NPC23880 |
| 1FDO | 08FE | 2389 | | LHR | R15,R14 | NPC23890 |
| 1FD2 | 4300 1A2E | 2390 | | B | PRTSTAT | NPC23900 |
| 1FD6 | C810 3135 | 2391 | ERR15 | LHI | R1,X'3135' | NPC23910 |
| 1FDA | 4300 1A9A | 2392 | | B | PRTDATA | NPC23920 |
| 1FDE | C810 3136 | 2393 | ERR16 | LHI | R1,X'3136' | NPC23930 |
| 1FE2 | 08CE | 2394 | | LHR | R12,R14 | NPC23940 |
| 1FE4 | 4300 1A9A | 2395 | | B | PRTDATA | NPC23950 |
| 1FE8 | C810 3137 | 2396 | ERR17 | LHI | R1,X'3137' | NPC23960 |
| 1FEC | 080A | 2397 | | LHR | RO,R10 | NPC23970 |
| 1FEE | 08FE | 2398 | | LHR | R15,R14 | NPC23980 |
| 1FF0 | 4300 1A2E | 2399 | | B | PRTSTAT | NPC23990 |
| 1FF4 | C810 3138 | 2400 | ERR18 | LHI | R1,X'3138' | NPC24000 |
| 1FF8 | 4300 1A70 | 2401 | | B | PRTREGS | NPC24010 |
| 1FFC | C810 3139 | 2402 | ERR19 | LHI | R1,X'3139' | NPC24020 |
| 2000 | 4300 1A70 | 2403 | | B | PRTREGS | NPC24030 |
| 2004 | C810 3230 | 2404 | ERR20 | LHI | R1,X'3230' | NPC24040 |
| 2008 | 4300 1A70 | 2405 | | B | PRTREGS | NPC24050 |
| 200C | C810 3231 | 2406 | ERR21 | LHI | R1,X'3231' | NPC24060 |
| 2010 | 4300 1A9A | 2407 | | B | PRTDATA | NPC24070 |
| 2014 | C810 3232 | 2408 | ERR22 | LHI | R1,X'3232' | NPC24080 |
| 2018 | 08CE | 2409 | | LHR | R12,R14 | NPC24090 |
| 201A | 4300 1A9A | 2410 | | B | PRTDATA | NPC24100 |
| 201E | C810 3233 | 2411 | ERR23 | LHI | R1,X'3233' | NPC24110 |
| 2022 | 4300 1A9A | 2412 | | B | PRTDATA | NPC24120 |
| 2026 | C810 3234 | 2413 | ERR24 | LHI | R1,X'3234' | NPC24130 |
| 202A | 08CE | 2414 | | LHR | R12,R14 | NPC24140 |
| 202C | 4300 1A9A | 2415 | | B | PRTDATA | NPC24150 |
| 2030 | C810 3235 | 2416 | ERR25 | LHI | R1,X'3235' | NPC24160 |
| 2034 | 08FE | 2417 | | LHR | R15,R14 | NPC24170 |
| 2036 | 080A | 2418 | | LHR | RO,R10 | NPC24180 |
| 2038 | 4300 1A2E | 2419 | | B | PRTSTAT | NPC24190 |
| 203C | C810 3236 | 2420 | ERR26 | LHI | R1,X'3236' | NPC24200 |
| 2040 | 4300 1A2E | 2421 | | B | PRTSTAT | NPC24210 |
| | | 2422 | * | | | NPC24220 |
| | | 2423 | * | | | NPC24230 |
| | | 2424 | ***** | | | NPC24240 |
| | | 2425 | * | | | NPC24250 |
| | | 2426 | *TEST3X | MOVE TESTING TO ANOTHER PROGRAM MEMORY BLOCK | | NPC24260 |
| | | 2427 | * | | | NPC24270 |
| | | 2428 | * | | | NPC24280 |
| 2044 | E140 0000 | 2429 | TEST3X | SVC 4,0 | SET PSW 7 | NPC24290 |
| 2048 | 48FC 0000 | 2430 | | LH R15,0(R12) | READ FROM THIS ADDRESS | NPC24300 |
| 204C | 40FC 0000 | 2431 | | STH R15,0(R12) | WRITE TO THIS LOCATION | NPC24310 |

SUBROUTINES

| | | | | | | |
|------|-----------|------|---------|---|--------------------------|----------|
| 2050 | E100 0000 | 2432 | SVC | 0,0 | CLEAR PSW 7 | MPC24320 |
| 2054 | 4300 114E | 2433 | B | TST3F | RETURN | MPC24330 |
| | | 2434 | ***** | | | MPC24340 |
| | | 2435 | * | | | MPC24350 |
| | | 2436 | *TEST6X | MOVE TEST TO ANOTHER PROGRAM MEMORY BLOCK | | MPC24360 |
| | | 2437 | * | | | MPC24370 |
| 2058 | E140 0000 | 2438 | TEST6X | SVC 4,0 | SET PSW 7 | MPC24380 |
| 205C | 2494 | 2439 | LIS | INCQR,4 | EXPECT EXECUTE VIOLATION | MPC24390 |
| 205E | 01FC | 2440 | BALR | R15,R12 | EXECUTE THIS ADDRESS | MPC24400 |
| 2060 | 2493 | 2441 | LIS | INCQR,3 | DO NOT EXPECT EX VIOL | MPC24410 |
| 2062 | E100 0000 | 2442 | SVC | 0,0 | CLEAR PSW 7 | MPC24420 |
| 2066 | 4300 13C0 | 2443 | B | TST6E | RETURN | MPC24430 |
| | | 2444 | ***** | | | MPC24440 |
| | | 2445 | * | | | MPC24450 |
| | | 2446 | * | PSW 0-32 LOAD | | MPC24460 |
| | | 2447 | * | | | MPC24470 |
| | | 2448 | * | | | MPC24480 |
| 206A | A000 | 2449 | HALT | DC X'A000',ORG | | MPC24490 |
| 206C | 0C14 | | | | | |
| 206E | 0000 | 2450 | SET1 | DC X'0000',DEVCHK | | MPC24500 |
| 2070 | 1B2E | | | | | |
| 2072 | 2000 | 2451 | ENABLE | DC X'2000',INCERR | | MPC24510 |
| 7 | A66 | | | | | |
| 2076 | | 2452 | BADR17 | DS 2 | | MPC24520 |

MESSAGES

| | | | | | | |
|------|---------------------|------|--|-----|--|----------|
| 2078 | OD0A | 2454 | TITLE | DC | X'0D0A',C'8-16E MEMORY PROTECT TEST 06-223R01',X'0D0A' | MPC24540 |
| 207A | 382D 3136 4520 4D45 | | | | | |
| 2082 | 4D4F 5259 2050 524F | | | | | |
| 208A | 5445 4354 2054 4553 | | | | | |
| 2092 | 5420 3036 2D32 3233 | | | | | |
| 209A | 5230 3120 | | | | | |
| 209E | OD0A | | | | | |
| | 0000 209F | 2455 | ENDOF | EQU | *-1 | MPC24550 |
| | | 2456 | * | | | MPC24560 |
| | | 2457 | * | | | MPC24570 |
| | | 2458 | * ERROR MESSAGE = ERROR TTEE | | | MPC24580 |
| | | 2459 | * EX XXXXXX | | | MPC24590 |
| | | 2460 | * RD YYYYYY | | | MPC24600 |
| | | 2461 | * | | | MPC24610 |
| | | 2462 | * TT = TEST NUMBER EE = ERROR NUMBER | | | MPC24620 |
| | | 2463 | * XXXXXX = EXP ECTED DATA YYYYYY = DATA READ | | | MPC24630 |
| | | 2464 | * | | | MPC24640 |
| 20A0 | OD0A | 2465 | ERRMSG | DC | X'0D0A' | MPC24650 |
| 20A2 | 4552 524F 5220 | 2466 | | DC | C'ERROR' | MPC24660 |
| 20A8 | 5858 | 2467 | TESTNUM | DC | X'5858' | MPC24670 |
| 20AA | 0000 | 2468 | ERRNUM | DC | X'0' | MPC24680 |
| 20AC | OD0A | 2469 | CRLF | DC | X'0D0A' | MPC24690 |
| | 0000 20AD | 2470 | END | EQU | *-1 | MPC24700 |
| 20AE | 424C 4F43 4B20 | 2471 | | DC | C'BLOCK' | MPC24710 |
| 20B4 | 0000 | 2472 | BLOCKN | DC | X'0000' | MPC24720 |
| 20B6 | OD0A | 2473 | | DC | X'0D0A' | MPC24730 |
| | 0000 20B7 | 2474 | ENDZ | EQU | *-1 | MPC24740 |
| 20B8 | 4D45 4D4F 5259 2020 | 2475 | | DC | C'MEMORY' | MPC24750 |
| 20C0 | 2020 | | | | | |
| 20C2 | 0000 | 2476 | MEMYMS | DC | 0 | MPC24760 |
| 20C4 | 2020 | 2477 | SPACE2 | DC | X'2020' | MPC24770 |
| 20C6 | 0000 | 2478 | MEMYLS | DC | 0 | MPC24780 |
| 20C8 | 0000 | 2479 | | DC | 0 | MPC24790 |
| 20CA | OD0A | 2480 | | DC | X'0D0A' | MPC24800 |
| | 0000 20CB | 2481 | END1 | EQU | *-1 | MPC24810 |
| | | 2482 | * | | | MPC24820 |
| | | 2483 | * | | | MPC24830 |
| | | 2484 | * | | | MPC24840 |
| 20CC | 0000 | 2485 | WN | DC | X'0000' | MPC24850 |
| 20CE | 2020 | 2486 | | DC | X'2020' | MPC24860 |
| 20D0 | 0000 | 2487 | RN | DC | X'0000' | MPC24870 |
| 20D2 | 2020 | 2488 | | DC | X'2020' | MPC24880 |
| 20D4 | 0000 | 2489 | EN | DC | X'0000' | MPC24890 |
| 20D6 | 2020 | 2490 | | DC | X'2020' | MPC24900 |
| 20D8 | OD0A | 2491 | | DC | X'0D0A' | MPC24910 |
| | 0000 20D9 | 2492 | ENDA | EQU | *-1 | MPC24920 |
| | | 2493 | * | | | MPC24930 |
| | | 2494 | * | | | MPC24940 |
| | | 2495 | * | | | MPC24950 |
| 20DA | 5354 4154 5553 2020 | 2496 | STATMSG | DC | C'STATUS' | MPC24960 |
| 20E2 | 0000 | 2497 | STATUS | DC | X'0' | MPC24970 |
| 20E4 | OD0A | 2498 | | DC | X'0D0A' | MPC24980 |
| 20E6 | 4445 5649 4345 2020 | 2499 | | DC | C'DEVICE' | MPC24990 |

MESSAGES

| | | | | | | | | | |
|------|------|------|------|------|----------|-----|--------------------------------|--|----------|
| 20EE | 0000 | | | 2500 | DEVADD | DC | X'0' | | HPC25000 |
| 20FO | ODCA | | | 2501 | | DC | X'OD0A' | | HPC25010 |
| | 0000 | 20F1 | | 2502 | STATEND | EQU | *-1 | | HPC25020 |
| | | | | 2503 | * | | | | HPC25030 |
| | | | | 2504 | * | | | | HPC25040 |
| | | | | 2505 | * | | | | HPC25050 |
| 20F2 | OD0A | | | 2506 | TESTMSG | DC | X'OD0A',C'TEST | | HPC25060 |
| 20F4 | 5445 | 5354 | 2020 | | | | | | |
| 20FA | 0000 | | | 2507 | VALUE | DC | X'0000' | | HPC25070 |
| 20FC | OD0A | | | 2508 | | DC | X'OD0A' | | HPC25080 |
| | 0000 | 20FD | | 2509 | TESTED | EQU | *-1 | | HPC25090 |
| | | | | 2510 | * | | | | HPC25100 |
| | | | | 2511 | * | | | | HPC25110 |
| | | | | 2512 | * | | | | HPC25120 |
| 20FE | OD0A | 3F20 | | 2513 | QMARK | DC | Y'OD0A3F20' | | HPC25130 |
| 2102 | OD0A | | | 2514 | | DC | X'OD0A' | | HPC25140 |
| | 0000 | 2103 | | 2515 | QEND | EQU | *-1 | | HPC25150 |
| | | | | 2516 | * | | | | HPC25160 |
| | | | | 2517 | * | | | | HPC25170 |
| | | | | 2518 | * | | | | HPC25180 |
| 2104 | FFFF | | | 2519 | ASTERISK | DC | X'FFFF' | | HPC25190 |
| 2106 | OD0A | 2A20 | | 2520 | | DC | Y'OD0A2A20' | | HPC25200 |
| | 0000 | 2109 | | 2521 | ENDAST | EQU | *-1 | | HPC25210 |
| | | | | 2522 | * | | | | HPC25220 |
| | | | | 2523 | * | | | | HPC25230 |
| | | | | 2524 | * | | | | HPC25240 |
| 210A | OD0A | | | 2525 | TOTMSG | DC | X'OD0A' | | HPC25250 |
| 210C | 0000 | | | 2526 | TOTALMSG | DC | 0 | | HPC25260 |
| 210E | 0000 | | | 2527 | | DC | 0 | | HPC25270 |
| 2110 | 544F | 5441 | 4C20 | 2528 | | DC | C'TOTAL | | HPC25280 |
| | 0000 | 2117 | | 2529 | TOTALEND | EQU | *-1 | | HPC25290 |
| 2118 | 4552 | 524F | 5220 | 2530 | | DC | C'ERROR',X'OD0A' | | HPC25300 |
| 211E | OD0A | | | | | | | | |
| | 0000 | 211F | | 2531 | ERROREND | EQU | *-1 | | HPC25310 |
| | | | | 2532 | * | | | | HPC25320 |
| | | | | 2533 | * | | | | HPC25330 |
| | | | | 2534 | * | | | | HPC25340 |
| 2120 | 4E4F | 2045 | 5252 | 2535 | NOERR | DC | C'NO ERROR',X'OD0A' | | HPC25350 |
| 2128 | OD0A | | | | | | | | |
| | 0000 | 2129 | | 2536 | ERREND | EQU | *-1 | | HPC25360 |
| | | | | 2537 | * | | | | HPC25370 |
| | | | | 2538 | * | | | | HPC25380 |
| | | | | 2539 | * | | | | HPC25390 |
| 212A | OD0A | | | 2540 | ILGMSG | DC | X'OD0A',C'ILLEGAL INSTRUCTION' | | HPC25400 |
| 212C | 494C | 4C45 | 4741 | | | | | | |
| 2134 | 494E | 5354 | 5255 | | | | | | |
| 213C | 494F | 4E20 | | | | | | | |
| 2140 | OD0A | | | 2541 | | DC | X'OD0A' | | HPC25410 |
| 2142 | 0000 | | | 2542 | ADRS00 | DC | 0 | | HPC25420 |
| 2144 | 0000 | | | 2543 | | DC | 0 | | HPC25430 |
| 2146 | 2020 | | | 2544 | | DC | X'2020' | | HPC25440 |
| 2148 | 0000 | | | 2545 | ADRS0 | DC | 0 | | HPC25450 |
| 214A | 0000 | | | 2546 | | DC | 0 | | HPC25460 |

MESSAGES

| | | | | | | | |
|------|------|----------------|------|----------|---------|--------------------------------|----------|
| 214C | OD0A | | 2547 | DC | X'0D0A' | | MPC25470 |
| | 0000 | 214D | 2548 | ILGEND | EQU | *-1 | MPC25480 |
| | | | 2549 | * | | | MPC25490 |
| | | | 2550 | * | | | MPC25500 |
| | | | 2551 | * | | | MPC25510 |
| 214E | OD0A | | 2552 | MACHMAL | DC | X'0D0A',C'MACHINE MALFUNCTION' | MPC25520 |
| 2150 | 4D41 | 4348 494E 4520 | | | | | |
| 2158 | 4D41 | 4C46 554E 4354 | | | | | |
| 2160 | 494F | 4E20 | | | | | |
| 2164 | OD0A | | 2553 | DC | X'0D0A' | | MPC25530 |
| 2166 | 0020 | | 2554 | CCADRS | DC | X'0020' | MPC25540 |
| 2168 | 0000 | | 2555 | MMADRS | DC | X'0000' | MPC25550 |
| 216A | 0020 | | 2556 | DC | X'0020' | | MPC25560 |
| 216C | OD0A | | 2557 | DC | X'0D0A' | | MPC25570 |
| | 0000 | 216D | 2558 | MMEND | EQU | *-1 | MPC25580 |
| | | | 2559 | * | | | MPC25590 |
| | | | 2560 | * | | | MPC25600 |
| | | | 2561 | * | | | MPC25610 |
| | | | 2562 | * | | | MPC25620 |
| | | | 2563 | * | | | MPC25630 |
| | | | 2564 | * | | | MPC25640 |
| 216E | OD0A | | 2565 | AUTOMSG | DC | X'0D0A' | MPC25650 |
| 2170 | 0000 | | 2566 | AUTOMSG1 | DC | 0 | MPC25660 |
| 2172 | 4630 | | 2567 | DC | X'4630' | | MPC25670 |
| | 0000 | 2173 | 2568 | AUTOEND | EQU | *-1 | MPC25680 |
| | | | 2569 | * | | | MPC25690 |
| | | | 2570 | * | | | MPC25700 |
| | | | 2571 | * | | | MPC25710 |
| 2174 | OD0A | | 2572 | TOCHESG | DC | X'0D0A',C'TOP OF MEMORY | MPC25720 |
| 2176 | 544F | 5020 4F46 204D | | | | | |
| 217E | 454D | 4F52 5920 2020 | | | | | |
| 2186 | 0000 | | 2573 | TOCMS | DC | 0 | MPC25730 |
| 2188 | 2020 | | 2574 | DC | X'2020' | | MPC25740 |
| 218A | 0000 | | 2575 | TOCLS | DC | 0 | MPC25750 |
| 218C | 0000 | | 2576 | DC | 0 | | MPC25760 |
| | 0000 | 218D | 2577 | TOCMESGE | EQU | *-1 | MPC25770 |
| 2190 | | | 2578 | ALIGN | 4 | | MPC25780 |
| 2190 | 0000 | | 2579 | ZERO | DC | X'0' | MPC25790 |

MEMORY ALLOCATION

| | | | | | | |
|------|-----------|------|----------|---------|-----------|----------|
| | | 2581 | * | | | MPC25810 |
| | | 2582 | * | | | MPC25820 |
| | | 2583 | * | | | MPC25830 |
| | | 2584 | | ALIGN 8 | | MPC25840 |
| 2198 | | 2585 | BLOCKMP | DC | 64 | MPC25850 |
| 2198 | 0040 | 2586 | BLOCKMZ | DC | 64 | MPC25860 |
| 219A | 0040 | 2587 | BLOCKREF | DC | 10 | MPC25870 |
| 219C | 000A | 2588 | BLOCKFLG | DC | 7 | MPC25880 |
| 219E | 00C7 | 2589 | BADRX | DC | X'01FE' | MPC25890 |
| 21A0 | 01FE | 2590 | | DC | X'0200' | MPC25900 |
| 21A2 | 0200 | 2591 | | DC | X'5FE' | MPC25910 |
| 21A4 | 05FE | 2592 | | DC | X'600' | MPC25920 |
| 21A6 | 06C0 | 2593 | | DC | X'9FE' | MPC25930 |
| 21A8 | 09FE | 2594 | | DC | Z(BADR6) | MPC25940 |
| 21AA | 0C12 | 2595 | | DC | Z(BADR7) | MPC25950 |
| 21AC | 0D0C | 2596 | | DC | Z(BADR8) | MPC25960 |
| 21AE | 0F36 | 2597 | | DC | Z(BADR9) | MPC25970 |
| 21B0 | 11C2 | 2598 | | DC | Z(BADR10) | MPC25980 |
| 21B2 | 1368 | 2599 | | DC | Z(BADR11) | MPC25990 |
| 21B4 | 1564 | 2600 | | DC | Z(BADR12) | MPC26000 |
| 21B6 | 17DA | 2601 | | DC | Z(BADR13) | MPC26010 |
| 21B8 | 190A | 2602 | | DC | Z(BADR14) | MPC26020 |
| 21BA | 1B2C | 2603 | | DC | Z(BADR15) | MPC26030 |
| 21BC | 1C86 | 2604 | | DC | Z(BADR16) | MPC26040 |
| 21BE | 1E7E | 2605 | | DC | Z(BADR17) | MPC26050 |
| 21C0 | 2076 | 2606 | | DC | X'2300' | MPC26060 |
| 21C2 | 2300 | 2607 | BADRZX | EQU | *-1 | MPC26070 |
| | 0000 21C3 | 2608 | * | | | MPC26080 |
| | | 2609 | | ALIGN 8 | | MPC26090 |
| 21C8 | | 2610 | TABLE | DS | 12 | MPC26100 |
| 21C8 | | 2611 | TOTAL | DC | 0 | MPC26110 |
| 21D4 | 00C0 | 2612 | TOTALERR | DC | 0 | MPC26120 |
| 21D6 | 0000 | 2613 | OPTSAV | DC | 0 | MPC26130 |
| 21D8 | 0000 | 2614 | TOC | DC | X'0' | MPC26140 |
| 21DA | 00C0 | 2615 | MPHF1 | DS | 2 | MPC26150 |
| 21DC | | 2616 | MPHF2 | DS | 2 | MPC26160 |
| 21DE | | 2617 | MPHF3 | DS | 2 | MPC26170 |
| 21E0 | | 2618 | MPHF4 | DS | 2 | MPC26180 |
| 21E2 | | 2619 | MPHF5 | DS | 2 | MPC26190 |
| 21E4 | | 2620 | MPHF6 | DS | 2 | MPC26200 |
| 21E6 | | 2621 | MPHF7 | DS | 2 | MPC26210 |
| 21E8 | | 2622 | MPHF8 | DS | 2 | MPC26220 |
| 21EA | | 2623 | RSAVE | DS | 32 | MPC26230 |
| 21EC | | 2624 | SAVEA | DSH | 5 | MPC26240 |
| 220C | | 2625 | SAVEC | DSH | 3 | MPC26250 |
| 2216 | | 2626 | SAVEE | DSH | 3 | MPC26260 |
| 221C | | 2627 | SAVEG | DSH | 5 | MPC26270 |
| 2222 | | 2628 | SAVEJ | DSH | 8 | MPC26280 |
| 222C | | 2629 | SSAVE | DSH | 16 | MPC26290 |
| 223C | | 2630 | SPSW | DS | 2 | MPC26300 |
| 225C | | 2631 | BLOCK | DS | 2 | MPC26310 |
| 225E | | 2632 | BLOCK2 | DS | 2 | MPC26320 |
| 2260 | | 2633 | BLOCK3 | DS | 2 | MPC26330 |
| 2262 | | | | | | |

391 398 403 415 426 435 438

536*

538 1540 1564 2196

1004 1006 1008 1123

1724 1733 1744 1757 1764 1778 1786
1932 2041 2055 2099 2164 2189 2291*

1652 2200 2641*

783 1839 1843 1904

1212 1221 1230 1250 1418 2108

2415

2368 2401 2403 2405
2382 2386 2390 2399 2419 2421

MPC27000
MPC27010
MPC27020
MPC27030
MPC27040
MPC27050
MPC27060
MPC27070
MPC27080

CHKSUM/M17 PUNCHER

| | | | | | | | | | | | | | | |
|-----|-----------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| | | 2247 | 2248 | 2250 | 2251 | 2256 | 2257 | 2260 | 2265 | 2357 | 2359 | 2361 | 2363 | 2365 |
| | | 2367 | 2369 | 2371 | 2373 | 2375 | 2379 | 2381 | 2383 | 2387 | 2391 | 2393 | 2396 | 2400 |
| | | 2402 | 2404 | 2406 | 2408 | 2411 | 2413 | 2416 | 2420 | 2652 | 2654 | 2658 | 2660 | 2663 |
| | | 2664 | 2667 | 2667 | 2668 | 2675 | 2675 | 2677 | 2680 | 2684 | 2686 | 2689 | 2693 | |
| R10 | 0000 000A | 88* | 101 | 233 | 257 | 266 | 268 | 275 | 508 | 614 | 615 | 615 | 715 | 888 |
| | | 965 | 982 | 1143 | 1249 | 1252 | 1253 | 1270 | 1271 | 1274 | 1303 | 1308 | 1836 | 1838 |
| | | 2102 | 2107 | 2110 | 2111 | 2138 | 2140 | 2167 | 2221 | 2223 | 2231 | 2235 | 2237 | 2292 |
| | | 2293 | 2293 | 2301 | 2305 | 2312 | 2314 | 2315 | 2317 | 2318 | 2319 | 2323 | 2324 | 2328 |
| | | 2336 | 2339 | 2340 | 2346 | 2347 | 2348 | 2376 | 2384 | 2388 | 2397 | 2418 | | |
| R11 | 0000 000B | 89* | 236 | 279 | 282 | 282 | 287 | 293 | 361 | 509 | 510 | 530 | 533 | 534 |
| | | 535 | 537 | 540 | 541 | 547 | 548 | 550 | 633 | 634 | 642 | 710 | 727 | 733 |
| | | 1336 | 1339 | 1363 | 1364 | 1365 | 1369 | 1672 | 2102 | 2116 | 2120 | 2167 | 2224 | 2225 |
| | | 2226 | 2227 | 2228 | 2230 | 2231 | 2277 | 2280 | 2283 | 2291 | 2296 | 2303 | 2304 | 2305 |
| | | 2311 | 2312 | 2318 | 2319 | 2322 | 2326 | 2327 | 2328 | 2330 | 2335 | 2336 | 2342 | 2347 |
| | | 2348 | 2350 | | | | | | | | | | | |
| R12 | 0000 000C | 90* | 239 | 241 | 243 | 268 | 271 | 273 | 440 | 452 | 455 | 531 | 541 | 548 |
| | | 556 | 627 | 628 | 630 | 636 | 637 | 639 | 645 | 649 | 653 | 703 | 706 | 706 |
| | | 707 | 707 | 710 | 733 | 800 | 807 | 808 | 875 | 877 | 884 | 942 | 950 | 957 |
| | | 960 | 1047 | 1049 | 1057 | 1130 | 1133 | 1135 | 1138 | 1355 | 1357 | 1361 | 1364 | 1455 |
| | | 1463 | 1467 | 1469 | 1497 | 1519 | 1520 | 1523 | 1526 | 1528 | 1529 | 1672 | 1679 | 1681 |
| | | 1891 | 1902 | 1903 | 2114 | 2115 | 2116 | 2117 | 2121 | 2121 | 2149 | 2150 | 2209 | 2210 |
| | | 2210 | 2211 | 2222 | 2225 | 2232 | 2232 | 2234 | 2295 | 2296 | 2298 | 2299 | 2309 | 2311 |
| | | 2333 | 2335 | 2394 | 2409 | 2414 | 2430 | 2431 | 2440 | | | | | |
| R13 | 0000 000D | 91* | 289 | 289 | 290 | 291 | 400 | 437 | 441 | 477 | 479 | 711 | 729 | 735 |
| | | 795 | 800 | 877 | 878 | 880 | 881 | 950 | 951 | 953 | 954 | 1042 | 1047 | 1345 |
| | | 1345 | 1349 | 1354 | 1363 | 1365 | 1398 | 1403 | 1403 | 1404 | 1414 | 1414 | 1415 | 1416 |
| | | 1417 | 1417 | 1421 | 1422 | 1423 | 1425 | 1427 | 1428 | 1429 | 1456 | 1465 | 1465 | 1476 |
| | | 1484 | 1484 | 1486 | 1487 | 1517 | 1521 | 1522 | 1522 | 1523 | 1531 | 1554 | 1555 | 1607 |
| | | 1619 | 1623 | 1625 | 1636 | 1671 | 1673 | 1673 | 1674 | 1683 | 1684 | 1685 | 1831 | 1833 |
| | | 1862 | 1864 | 1883 | 1886 | 1890 | 1890 | 1893 | 1894 | 1895 | 1896 | 1900 | 2119 | 2120 |
| | | 2207 | 2208 | 2211 | 2212 | 2310 | 2334 | | | | | | | |
| R14 | 0000 000E | 92* | 102 | 251 | 252 | 253 | 254 | 255 | 258 | 259 | 262 | 263 | 286 | 288 |
| | | 292 | 294 | 299 | 300 | 304 | 343 | 344 | 350 | 352 | 393 | 513 | 520 | 546 |
| | | 559 | 566 | 614 | 617 | 647 | 651 | 655 | 704 | 708 | 708 | 709 | 709 | 711 |
| | | 735 | 876 | 880 | 885 | 943 | 953 | 958 | 961 | 1131 | 1134 | 1136 | 1139 | 1252 |
| | | 1310 | 1312 | 1315 | 1346 | 1347 | 1350 | 1358 | 1359 | 1374 | 1376 | 1377 | 1378 | 1380 |
| | | 1407 | 1410 | 1411 | 1412 | 1412 | 1413 | 1419 | 1469 | 1470 | 1471 | 1494 | 1504 | 1506 |
| | | 1507 | 1508 | 1509 | 1609 | 1614 | 1617 | 1620 | 1621 | 1632 | 1675 | 1676 | 1678 | 1695 |
| | | 1707 | 1711 | 1715 | 1761 | 1783 | 1824 | 1839 | 1843 | 1904 | 1908 | 2092 | 2096 | 2105 |
| | | 2110 | 2132 | 2157 | 2161 | 2170 | 2180 | 2186 | 2201 | 2213 | 2222 | 2223 | 2238 | 2377 |
| | | 2385 | 2389 | 2394 | 2398 | 2409 | 2414 | 2417 | | | | | | |
| R15 | 0000 000F | 93* | 238 | 245 | 245 | 246 | 247 | 249 | 249 | 250 | 252 | 255 | 259 | 263 |
| | | 270 | 270 | 295 | 296 | 296 | 297 | 298 | 307 | 310 | 344 | 352 | 363 | 389 |
| | | 459 | 475 | 512 | 513 | 514 | 515 | 519 | 520 | 554 | 562 | 569 | 605 | 606 |
| | | 607 | 608 | 609 | 612 | 613 | 619 | 620 | 623 | 625 | 627 | 628 | 630 | 635 |
| | | 636 | 637 | 639 | 644 | 645 | 649 | 653 | 690 | 691 | 692 | 693 | 694 | 697 |
| | | 698 | 701 | 702 | 703 | 704 | 727 | 729 | 740 | 743 | 744 | 746 | 747 | 748 |
| | | 749 | 752 | 753 | 759 | 762 | 763 | 788 | 790 | 791 | 793 | 794 | 801 | 803 |
| | | 804 | 807 | 808 | 812 | 834 | 859 | 860 | 862 | 863 | 867 | 868 | 871 | 874 |
| | | 875 | 876 | 884 | 885 | 896 | 925 | 926 | 928 | 929 | 934 | 936 | 937 | 940 |
| | | 941 | 942 | 943 | 956 | 957 | 958 | 960 | 961 | 996 | 999 | 1000 | 1001 | 1002 |
| | | 1005 | 1006 | 1007 | 1008 | 1009 | 1033 | 1034 | 1039 | 1040 | 1048 | 1049 | 1050 | 1052 |
| | | 1053 | 1057 | 1061 | 1101 | 1102 | 1104 | 1105 | 1117 | 1119 | 1120 | 1124 | 1125 | 1127 |

CHKSUM/M17 PUNCHER

| | | | | | | | | | | | | | | |
|--------|-----------|-------|-------|------|------|-------|------|-------|------|-------|------|------|------|------|
| | | 1128 | 1129 | 1130 | 1131 | 1133 | 1134 | 1135 | 1136 | 1138 | 1139 | 1155 | 1205 | 1206 |
| | | 1214 | 1215 | 1215 | 1223 | 1224 | 1231 | 1251 | 1255 | 1257 | 1266 | 1267 | 1268 | 1269 |
| | | 1272 | 1272 | 1305 | 1310 | 1337 | 1343 | 1347 | 1358 | 1370 | 1375 | 1378 | 1381 | 1400 |
| | | 1401 | 1402 | 1405 | 1406 | 1413 | 1415 | 1418 | 1421 | 1426 | 1427 | 1451 | 1460 | 1461 |
| | | 1472 | 1473 | 1475 | 1493 | 1494 | 1495 | 1496 | 1503 | 1503 | 1506 | 1518 | 1524 | 1525 |
| | | 1525 | 1526 | 1527 | 1527 | 1530 | 1542 | 1552 | 1553 | 1554 | 1562 | 1563 | 1574 | 1575 |
| | | 1575 | 1576 | 1577 | 1587 | 1591 | 1593 | 1594 | 1608 | 1608 | 1616 | 1619 | 1627 | 1627 |
| | | 1630 | 1634 | 1637 | 1638 | 1648 | 1655 | 1657 | 1659 | 1678 | 1680 | 1682 | 1683 | 1684 |
| | | 1686 | 1698 | 1718 | 1724 | 1733 | 1744 | 1757 | 1764 | 1778 | 1786 | 1810 | 1836 | 1837 |
| | | 1846 | 1849 | 1867 | 1868 | 1869 | 1872 | 1889 | 1911 | 1926 | 1927 | 1932 | 2041 | 2055 |
| | | 2099 | 2109 | 2127 | 2130 | 2132 | 2141 | 2164 | 2186 | 2187 | 2189 | 2192 | 2193 | 2194 |
| | | 2255 | 2278 | 2285 | 2309 | 2310 | 2331 | 2333 | 2334 | 2338 | 2343 | 2344 | 2345 | 2377 |
| | | 2385 | 2389 | 2398 | 2417 | 2430 | 2431 | 2440 | 2674 | 2681 | 2694 | 2701 | | |
| R2 | 0000 0002 | 80* | 108 | 120 | 130 | 221 | 227 | 610 | 694 | 717 | 717 | 757 | 764 | 791 |
| | | 816 | 832 | 836 | 945 | 945 | 969 | 969 | 991 | 991 | 993 | 1107 | 1107 | 1110 |
| | | 1111 | 1112 | 1116 | 1408 | 1408 | 1748 | 1751 | 1774 | 1776 | 2125 | 2125 | 2198 | 2248 |
| | | 2250 | 2251 | 2252 | 2252 | 2257 | 2258 | 2260 | 2265 | 2266 | 2281 | 2655 | 2664 | 2666 |
| | | 2690 | | | | | | | | | | | | |
| R3 | 0000 0003 | 81* | 94 | 109 | 222 | 228 | 229 | 234 | 256 | 264 | 269 | 277 | 373 | 373 |
| | | 377 | 380 | 381 | 446 | 449 | 449 | 452 | 930 | 971 | 989 | 998 | 1004 | 1108 |
| | | 1480 | 1483 | 1491 | 2261 | 2262 | 2263 | 2656 | 2676 | 2685 | 2702 | 2703 | | |
| R4 | 0000 0004 | 82* | 95 | 110 | 111 | 113 | 115 | 123 | 125 | 239 | 240 | 242 | 269 | 272 |
| | | 274 | 374 | 375 | 379 | 1109 | 1579 | 1580 | 1581 | 1582 | 1589 | 1750 | 1760 | 1767 |
| | | 1768 | 1802 | 1804 | 1814 | 2657 | 2659 | 2661 | 2665 | 2665 | 2666 | 2683 | 2687 | |
| R5 | 0000 0005 | 83* | 96 | 113 | 123 | 242 | 274 | 375 | 377 | 1895 | 1899 | 1907 | 2658 | 2659 |
| | | 2686 | 2687 | 2688 | | | | | | | | | | |
| R6 | 0000 0006 | 84* | 97 | 115 | 116 | 116 | 118 | 125 | 261 | 266 | 272 | 277 | 401 | 403 |
| | | 404 | 405 | 406 | 407 | 408 | 409 | 410 | 411 | 413 | 415 | 416 | 417 | 418 |
| | | 419 | 420 | 421 | 422 | 424 | 426 | 427 | 428 | 429 | 430 | 431 | 432 | 433 |
| | | 438 | 442 | 444 | 444 | 447 | 450 | 460 | 460 | 473 | 474 | 1706 | 1770 | 1770 |
| | | 1771 | 1774 | 1782 | 1789 | 1796 | 1796 | 1800 | 1802 | 1821 | 1822 | 2670 | 2671 | 2672 |
| | | 2677 | 2678 | 2688 | 2691 | 2703 | 2704 | | | | | | | |
| R7 | 0000 0007 | 85* | 98 | 119 | 120 | 121 | 122 | 129 | 129 | 130 | 545 | 1710 | 1803 | 1803 |
| | | 1813 | 1815 | 1818 | 2104 | 2169 | 2209 | | | | | | | |
| R8 | 0000 0008 | 86* | 99 | 112 | 112 | 122 | 127 | 260 | 264 | 271 | 275 | 355 | 356 | 484 |
| | | 484 | 485 | 486 | 487 | 488 | 489 | 492 | 497 | 502 | 504 | 507 | 508 | 521 |
| | | 522 | 1714 | 1806 | 1808 | 1809 | 2704 | | | | | | | |
| R9 | 0000 0009 | 87* | 100 | 241 | 273 | 351 | 376 | 397 | 425 | 443 | 445 | 464 | 468 | 470 |
| | | 2152 | 2156 | | | | | | | | | | | |
| RDCHR | 0000 OCA6 | 363* | 371 | | | | | | | | | | | |
| RDCHD | 0000 OA30 | 162* | 361 | 510 | 540 | 547 | 1401 | 1957 | 1966 | 1977 | 2342 | | | |
| READ | 0000 15C4 | 650 | 724 | 818 | 979 | 1284* | | | | | | | | |
| REGSN | 0000 190C | 1706* | 1868 | | | | | | | | | | | |
| RET1 | 0000 0003 | 94* | 824 | 826 | 1067 | 1069 | 1481 | 1510 | 1556 | | | | | |
| RET2 | 0000 000E | 102* | 828 | 830 | 1071 | 1073 | 1489 | 1532 | 1567 | | | | | |
| RETURN | 0000 1F3A | 2342* | 2351 | | | | | | | | | | | |
| RN | 0000 20D0 | 1713 | 2487* | | | | | | | | | | | |
| RORW | 0000 15D4 | 629 | 638 | 732 | 823 | 894 | 981 | 1060 | 1145 | 1292* | | | | |
| RSAVE | 0000 21EC | 175 | 2623* | | | | | | | | | | | |
| RTN1 | 0000 1B28 | 1931 | 1935* | | | | | | | | | | | |
| RUN | 0000 0C60 | 333* | 383 | | | | | | | | | | | |
| SAVEA | 0000 220C | 1336 | 1339 | 1369 | 1480 | 1483 | 1491 | 2624* | | | | | | |
| SAVEC | 0000 2216 | 1374 | 1380 | 1648 | 1649 | 1657 | 1658 | 1671 | 1685 | 2625* | | | | |

CHKSUM/M17 PUNCHER

| | | | | | | |
|----------|------|------|-------|-------|------|-------|
| TOTALERR | 0000 | 21D6 | 487 | 565 | 1853 | 2612* |
| TOTALMSG | 0000 | 210C | 561 | 568 | 570 | 2526* |
| TOTMSG | 0000 | 210A | 563 | 2525* | | |
| TST | 0000 | 0EAO | 521 | 523* | | |
| TST00 | 0000 | 0EBO | 441* | 454 | | |
| TST01 | 0000 | 0DC0 | 447* | 451 | | |
| TST1A | 0000 | 0F70 | 622* | 657 | | |
| TST1B | 0000 | 0FF8 | 619 | 658* | | |
| TST2 | 0000 | 0DCC | 448 | 452* | | |
| TST2A | 0000 | 1008 | 693* | 765 | | |
| TST2B | 0000 | 1012 | 696* | 741 | | |
| TST2C | 0000 | 101A | 698* | 756 | | |
| TST2D | 0000 | 1024 | 701* | 731 | 737 | |
| TST2E | 0000 | 1076 | 718 | 724* | | |
| TST2EE | 0000 | 1082 | 723 | 727* | | |
| TST2F | 0000 | 1092 | 716 | 732* | | |
| TST2G | 0000 | 10A6 | 691 | 738* | | |
| TST2H | 0000 | 10B2 | 739 | 742* | | |
| TST2J | 0000 | 10D0 | 745 | 752* | | |
| TST2K | 0000 | 10D2 | 751 | 753* | | |
| TST2Z | 0000 | 10E2 | 750 | 757* | | |
| TST3A | 0000 | 1108 | 789* | 837 | | |
| TST3B | 0000 | 110A | 790* | | | |
| TST3C | 0000 | 1112 | 792* | 831 | | |
| TST3D | 0000 | 111E | 795* | 829 | | |
| TST3E | 0000 | 1122 | 797* | 825 | | |
| TST3F | 0000 | 114E | 810* | 2433 | | |
| TST3G | 0000 | 1170 | 817 | 820* | | |
| TST3H | 0000 | 117A | 813 | 823* | | |
| TST3J | 0000 | 117E | 819 | 822 | 824* | |
| TST3Z | 0000 | 1194 | 827 | 830* | | |
| TST4A | 0000 | 11CA | 865* | 897 | | |
| TST4B | 0000 | 11DE | 871* | 895 | | |
| TST4C | 0000 | 122A | 889 | 893* | | |
| TST4D | 0000 | 122E | 892 | 894* | | |
| TST4Z | 0000 | 1236 | 862 | 896* | | |
| TST5A | 0000 | 1256 | 930* | 995 | 997 | |
| TST5B | 0000 | 126E | 937* | 1010 | | |
| TST5C | 0000 | 1278 | 940* | 988 | | |
| TST5D | 0000 | 1296 | 946 | 949* | | |
| TST5E | 0000 | 129A | 948 | 950* | | |
| TST5F | 0000 | 12F2 | 972 | 976* | | |
| TST5G | 0000 | 12FC | 970 | 979* | | |
| TST5H | 0000 | 1302 | 966 | 981* | | |
| TST5J | 0000 | 1306 | 975 | 978 | 980 | 982* |
| TST5K | 0000 | 1316 | 983 | 987* | | |
| TST5L | 0000 | 131A | 986 | 988* | | |
| TST5M | 0000 | 1334 | 990 | 996* | | |
| TST5Z | 0000 | 131E | 928 | 989* | | |
| TST6A | 0000 | 1378 | 1037* | 1074 | | |
| TST6B | 0000 | 138A | 1042* | 1072 | | |
| TST6C | 0000 | 138E | 1044* | 1068 | | |
| TST6D | 0000 | 13BC | 1059* | | | |

MEMORY ALLOCATION

| | | | | | | |
|------|-----------|------|----------|-------|-----------|----------|
| | 2581 | * | | | | MPC25810 |
| | 2582 | * | | | | MPC25820 |
| | 2583 | * | | | | MPC25830 |
| 2198 | 2584 | | ALIGN | 8 | | MPC25840 |
| 2198 | 0040 | 2585 | BLOCKMP | DC | 64 | MPC25850 |
| 219A | 0040 | 2586 | BLOCKMZ | DC | 64 | MPC25860 |
| 219C | 000A | 2587 | BLOCKREF | DC | 10 | MPC25870 |
| 219E | 0007 | 2588 | BLOCKFLG | DC | 7 | MPC25880 |
| 21A0 | 01FE | 2589 | BADRX | DC | X'01FE' | MPC25890 |
| 21A2 | 0200 | 2590 | | DC | X'0200' | MPC25900 |
| 21A4 | 05FE | 2591 | | DC | X'5FE' | MPC25910 |
| 21A6 | 06C0 | 2592 | | DC | X'600' | MPC25920 |
| 21A8 | 09FE | 2593 | | DC | X'9FE' | MPC25930 |
| 21AA | 0C12 | 2594 | | DC | Z(BADR6) | MPC25940 |
| 21AC | 0D0C | 2595 | | DC | Z(BADR7) | MPC25950 |
| 21AE | 0F36 | 2596 | | DC | Z(BADR8) | MPC25960 |
| 21B0 | 11C2 | 2597 | | DC | Z(BADR9) | MPC25970 |
| 21B2 | 1368 | 2598 | | DC | Z(BADR10) | MPC25980 |
| 21B4 | 1564 | 2599 | | DC | Z(BADR11) | MPC25990 |
| 21B6 | 17DA | 2600 | | DC | Z(BADR12) | MPC26000 |
| 21B8 | 190A | 2601 | | DC | Z(BADR13) | MPC26010 |
| 21BA | 1B2C | 2602 | | DC | Z(BADR14) | MPC26020 |
| 21BC | 1C86 | 2603 | | DC | Z(BADR15) | MPC26030 |
| 21BE | 1E7E | 2604 | | DC | Z(BADR16) | MPC26040 |
| 21C0 | 2076 | 2605 | | DC | Z(BADR17) | MPC26050 |
| 21C2 | 2300 | 2606 | | DC | X'2300' | MPC26060 |
| | 0000 21C3 | 2607 | BADRXZ | EQU | *-1 | MPC26070 |
| | | 2608 | * | | | MPC26080 |
| 21C8 | | 2609 | | ALIGN | 8 | MPC26090 |
| 21C8 | | 2610 | TABLE | DS | 12 | MPC26100 |
| 21D4 | 0000 | 2611 | TOTAL | DC | 0 | MPC26110 |
| 21D6 | 0000 | 2612 | TOTALERR | DC | 0 | MPC26120 |
| 21D8 | 0000 | 2613 | OPTSAV | DC | 0 | MPC26130 |
| 21DA | 00C0 | 2614 | TOC | DC | X'0' | MPC26140 |
| 21DC | | 2615 | MPHF1 | DS | 2 | MPC26150 |
| 21DE | | 2616 | MPHF2 | DS | 2 | MPC26160 |
| 21E0 | | 2617 | MPHF3 | DS | 2 | MPC26170 |
| 21E2 | | 2618 | MPHF4 | DS | 2 | MPC26180 |
| 21E4 | | 2619 | MPHF5 | DS | 2 | MPC26190 |
| 21E6 | | 2620 | MPHF6 | DS | 2 | MPC26200 |
| 21E8 | | 2621 | MPHF7 | DS | 2 | MPC26210 |
| 21EA | | 2622 | MPHF8 | DS | 2 | MPC26220 |
| 21EC | | 2623 | RSAVE | DS | 32 | MPC26230 |
| 220C | | 2624 | SAVEA | DSH | 5 | MPC26240 |
| 2216 | | 2625 | SAVEC | DSH | 3 | MPC26250 |
| 221C | | 2626 | SAVEE | DSH | 3 | MPC26260 |
| 2222 | | 2627 | SAVEG | DSH | 5 | MPC26270 |
| 222C | | 2628 | SAVEJ | DSH | 8 | MPC26280 |
| 223C | | 2629 | SSAVE | DSH | 16 | MPC26290 |
| 225C | | 2630 | SPSW | DS | 2 | MPC26300 |
| 225E | | 2631 | BLOCK | DS | 2 | MPC26310 |
| 2260 | | 2632 | BLOCK2 | DS | 2 | MPC26320 |
| 2262 | | 2633 | BLOCK3 | DS | 2 | MPC26330 |

MEMORY ALLOCATION

| | | | | | | |
|------|-----------|------|----------|-----|-----|----------|
| 2264 | | 2634 | BLOCKMAX | DS | 2 | MPC26340 |
| 2266 | | 2635 | BLKPROT | DS | 2 | MPC26350 |
| 2268 | | 2636 | BLKPT | DS | 2 | MPC26360 |
| 226A | | 2637 | BLKPT | DS | 2 | MPC26370 |
| 226C | | 2638 | MAPSAVE | DS | 2 | MPC26380 |
| 226E | | 2639 | MESSAV | DS | 4 | MPC26390 |
| 2272 | | 2640 | PSWSAVE | DS | 2 | MPC26400 |
| 2274 | | 2641 | PROTPATT | DS | 2 | MPC26410 |
| 2276 | | 2642 | PSW811 | DS | 2 | MPC26420 |
| 2278 | | 2643 | TESTEND | DS | 2 | MPC26430 |
| 227A | | 2644 | TTYBUF | DS | 6 | MPC26440 |
| 2280 | | 2645 | ACTTOCHS | DS | 2 | MPC26450 |
| 2282 | | 2646 | ACTTCCLS | DS | 2 | MPC26460 |
| 2284 | | 2647 | ACTADUP | DS | 2 | MPC26470 |
| 2286 | | 2648 | LINKSAV | DS | 2 | MPC26480 |
| | 0000 2287 | 2649 | PROGEND | EQU | *-1 | MPC26490 |

CHKSUM/M17 PUNCHER

| | | | | | | | |
|------|-----------|------|----------|------|--------------|----------------------------------|----------|
| 2288 | 2400 | 2651 | \$CHKSUM | LIS | R0,0 | PUNCH M17 TAPE WITH CHECKSUM | MPC26510 |
| 228A | 9510 | 2652 | | EPSR | R1,R0 | CLEAR PSW | MPC26520 |
| | | 2653 | * | | | | MPC26530 |
| 228C | C810 0A00 | 2654 | | LHI | R1,X'0A00' | START ADDRESS | MPC26540 |
| 2290 | 2421 | 2655 | | LIS | R2,1 | INCREMENT | MPC26550 |
| 2292 | C830 2287 | 2656 | | LHI | R3,PROGEND | FINAL ADDRESS | MPC26560 |
| 2296 | 2440 | 2657 | | LIS | R4,0 | CHECKSUM BYTE | MPC26570 |
| 2298 | D351 0000 | 2658 | \$GEN | LB | R5,0(R1) | | MPC26580 |
| 229C | 0745 | 2659 | | XHR | R4,R5 | | MPC26590 |
| 229E | C110 2298 | 2660 | | BXLE | R1,\$GEN | | MPC26600 |
| 22A2 | D240 00B9 | 2661 | | STB | R4,MN+3 | CHECKSUM BYTE TO BOOT LOADER | MPC26610 |
| | | 2662 | * | | | | MPC26620 |
| 22A6 | C810 0080 | 2663 | \$TAPE | LHI | R1,X'0080' | | MPC26630 |
| 22AA | 9E21 | 2664 | | OCR | R2,R1 | DISPLAY : NORMAL MODE | MPC26640 |
| 22AC | 9444 | 2665 | | EXBR | R4,R4 | | MPC26650 |
| 22AE | 9824 | 2666 | | WHR | R2,R4 | CHECKSUM BYTE TO D1 | MPC26660 |
| 22B0 | 9411 | 2667 | | EXBR | R1,R1 | | MPC26670 |
| 22B2 | 9501 | 2668 | | EPSR | R0,R1 | HALT PROCESSOR. | MPC26680 |
| | | | | | | | |
| 22B4 | D360 007A | 2670 | \$PUNCH | LB | R6,X'7A' | GET BOUTDV (PUNCH) ADDRESS. | MPC26700 |
| 22B8 | DE60 007B | 2671 | | OC | R6,X'7B' | START TAPE PUNCH | MPC26710 |
| 22BC | 9D60 | 2672 | | SSR | R6,R0 | | MPC26720 |
| 22BE | 2081 | 2673 | | BTBS | 8,1 | | MPC26730 |
| 22C0 | 41F0 2302 | 2674 | | BAL | R15,\$STAPL | PUNCH LEADER | MPC26740 |
| 22C4 | 9411 | 2675 | | EXBR | R1,R1 | (R1) = X'8000' | MPC26750 |
| 22C6 | C830 00CF | 2676 | | LHI | R3,X'CF' | | MPC26760 |
| 22CA | DA61 0000 | 2677 | \$PNCH1 | WD | R6,0(R1) | PUNCH BOOT LOADER | MPC26770 |
| 22CE | 9D60 | 2678 | | SSR | R6,R0 | | MPC26780 |
| 22D0 | 2081 | 2679 | | BTBS | 8,1 | | MPC26790 |
| 22D2 | C110 22CA | 2680 | | BXLE | R1,\$PNCH1 | | MPC26800 |
| 22D6 | 41F0 2308 | 2681 | | BAL | R15,\$STAPL1 | PUNCH ONE-FOLD GAP. | MPC26810 |
| | | 2682 | * | | | | MPC26820 |
| 22DA | D340 00B9 | 2683 | | LB | R4,MN+3 | GET CHECKSUM BYTE | MPC26830 |
| 22DE | C810 0A00 | 2684 | | LHI | R1,X'A00' | START ADDRESS | MPC26840 |
| 22E2 | C830 2287 | 2685 | | LHI | R3,PROGEND | END ADDRESS | MPC26850 |
| 22E6 | D351 0000 | 2686 | \$PNCH2 | LB | R5,0(R1) | PUNCH PROGRAM | MPC26860 |
| 22EA | 0745 | 2687 | | XHR | R4,R5 | | MPC26870 |
| 22EC | 9A65 | 2688 | | WDR | R6,R5 | | MPC26880 |
| 22EE | 9401 | 2689 | | EXBR | R0,R1 | | MPC26890 |
| 22F0 | 9820 | 2690 | | WHR | R2,R0 | DATA ADDRESS TO DISPLAY | MPC26900 |
| 22F2 | 9D60 | 2691 | | SSR | R6,R0 | | MPC26910 |
| 22F4 | 2081 | 2692 | | BTBS | 8,1 | | MPC26920 |
| 22F6 | C110 22E6 | 2693 | | BXLE | R1,\$PNCH2 | | MPC26930 |
| 22FA | 41F0 2302 | 2694 | | BAL | R15,\$STAPL | PUNCH TRAILER. | MPC26940 |
| 22FE | 4300 22A6 | 2695 | | B | \$TAPE | DISPLAY CHECKSUM, HALT PROCESSOR | MPC26950 |
| | | | | | | | |
| 2302 | C800 0100 | 2697 | \$STAPL | LHI | R0,256 | TO PUNCH BLANK LEADER | MPC26970 |
| 2306 | 2303 | 2698 | | BS | \$STAPLP | | MPC26980 |
| 2308 | C800 0055 | 2699 | \$STAPL1 | LHI | R0,85 | TO PUNCH 1-FOLD GAP | MPC26990 |

