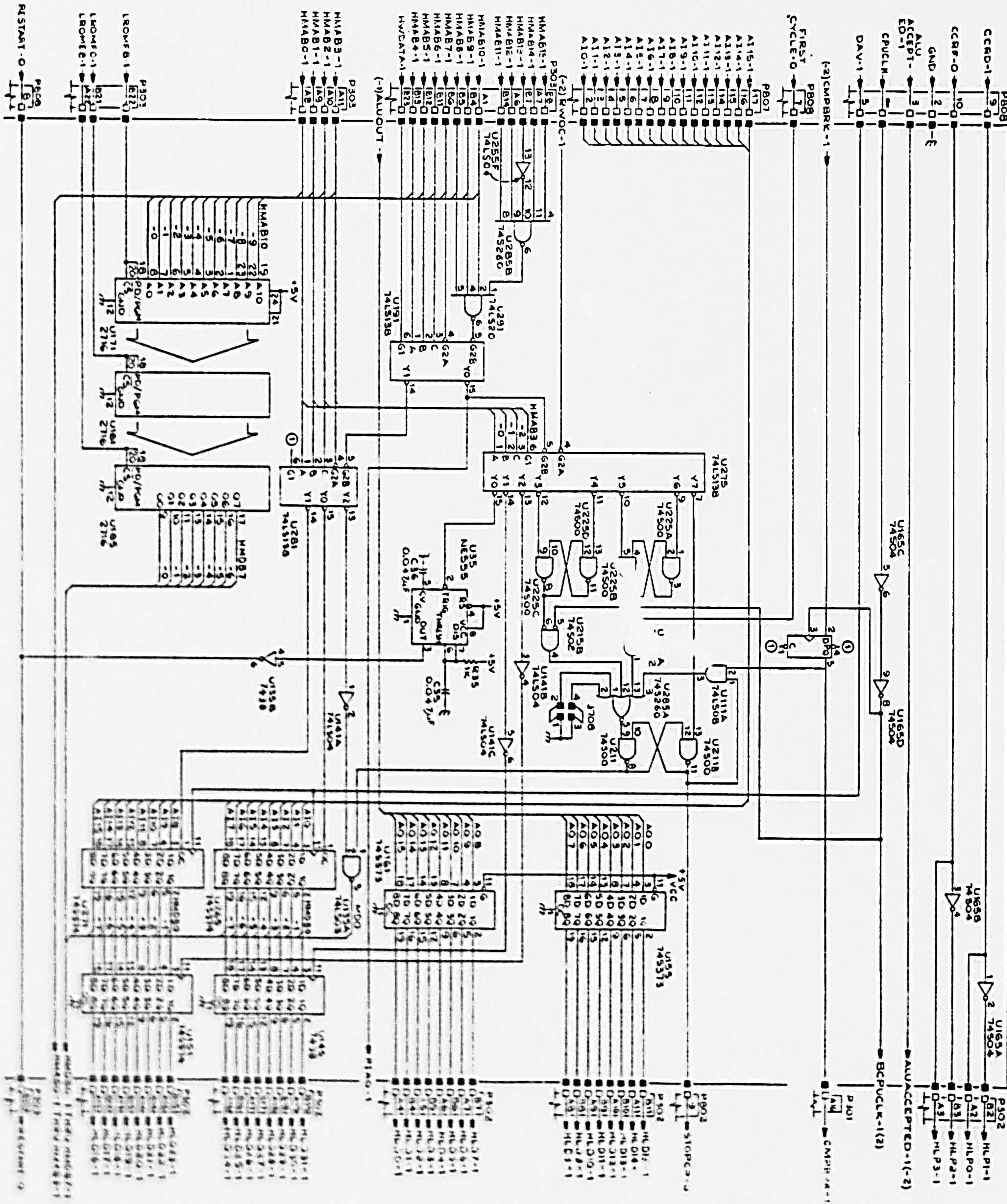


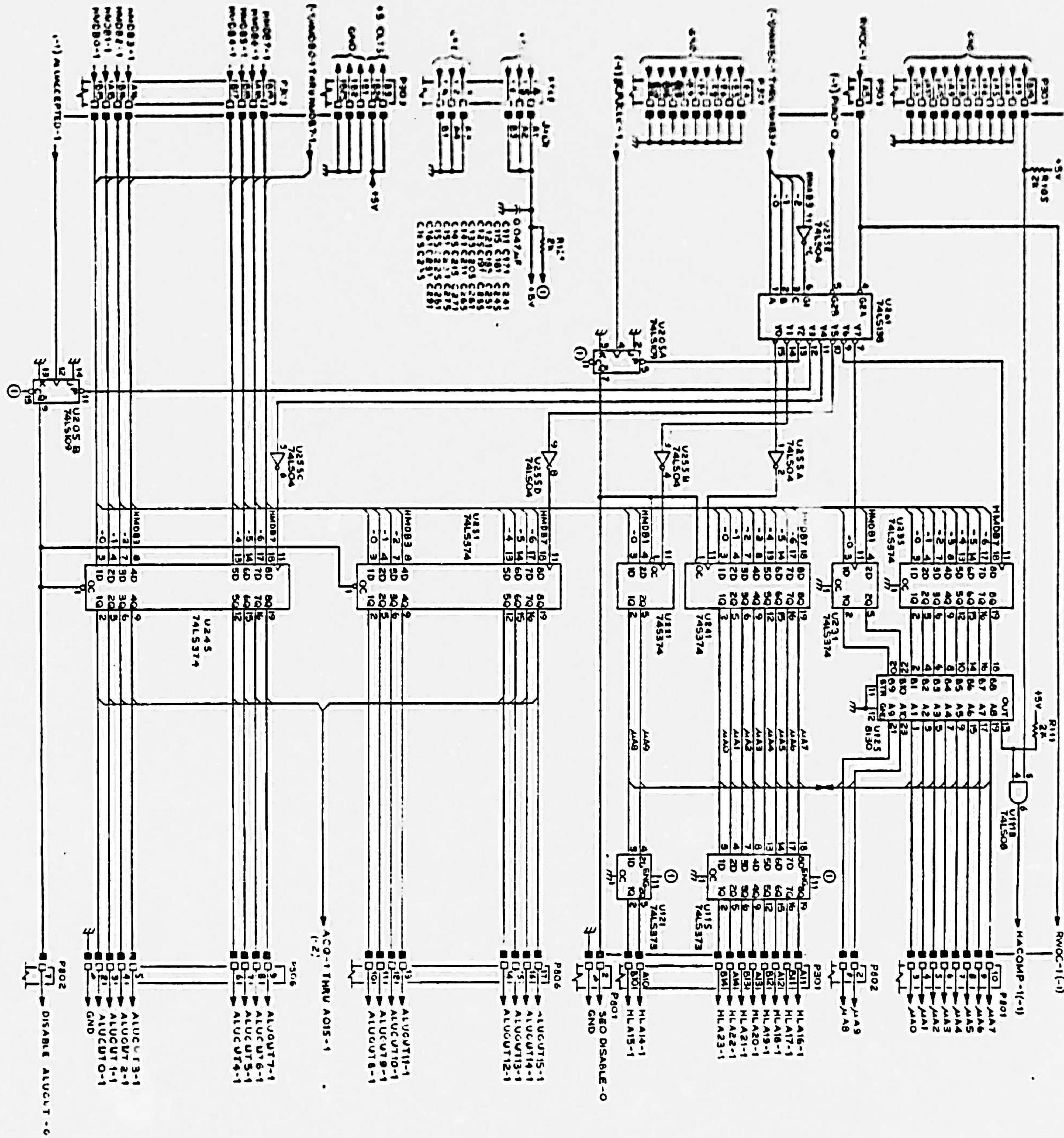
**067-0942-99**

**PERSONALITY BOARD FOR 4052/4054 GRAPHIC COMPUTING SYSTEMS**

**For Additional Copies Contact IOD Service Support  
© Tektronix, Inc. 1979**



067-0942-99  
ROWNE WITH 4032



ACTION

A - ADD  
C - CHANGE  
D - DELETE  
E - ESTIMATE  
R - REGULAR

030 1441 00

## BILL OF MATERIAL AND OPERATION RECORD

ITEM NUMBER:

1067-09412-99

METHOD:

DATE:

DESCRIPTION:  
1067-09412-99 PERSONALITY Bd.

MOD NO:

DEPT:

2125

BY:

MICHAEL MERRAN

COMPONENT PARTS	QUANTITY	BATCH	CURRENT LABOR HOURS		CURRENT MACHINE HOURS		JIG	MATERIAL	METHOD	DATE
			OPERATION	TIME	OPERATION	TIME				
1067-09412-99 PERSONALITY Bd.	1	1067-09412-99 PERSONALITY Bd.								
1067-09412-99 PERSONALITY Bd.	1	1067-09412-99 PERSONALITY Bd.								

WHITE - COST SERVICES

YELLOW - MATER. CONTROL

PINK - ORIGINATOR

PAGE \_\_\_\_\_

REMARKS:

ITEM NUMBER:

067-09412-99

067-0942-99

Personality Board For  
4052/4054 Graphic Computing Systems

INTRODUCTION

The 067-0942-99 Personality Board is used with the 067-0902-99 System Test Fixture to troubleshoot 4052 and 4054 circuitry. The Personality Board allows the System Test Fixture to be connected to a 4052 or 4054's ALU (Arithmetic Logic Unit) and MCP (Memory Command Processor) Boards. ROM on the Personality Board defines the functions that the System Test Fixture can perform. The firmware listing for the Personality Board ROM is included in this manual as an aid in troubleshooting the System Test Fixture. The 067-0942-99 comes with an overlay that describes the meaning of LEDs and switches on the System Test Fixture. Unlabeled switches and LEDs are not used with 4052 and 4054 systems.

INSTALLATION

The Personality Board must first be installed in the 067-0902-99 System Test Fixture and then connected to the 4052 or 4054 being tested. If not already installed, mount the Personality Board within the System Test Fixture with screws and spacers. Attach the three ribbon cables between the Personality Board and the Fixture's Controller Board.

On a 4054 the connectors on the ALU and MCP Boards are accessible after removing the 4054's top cover.

For 4052 systems, first remove the 4-board CPU Assembly from the unit. A long 5V power cable (175-2431-00) must be used between the 4052's 5V supply and the board set.

Next, connect the ribbon cables from the Personality Board to the ALU and MCP according to Table 1.

Table 1  
Test Fixture and 4052/4054 Connections

Personality Board Connectors	Connector Color	4052/4054 Connectors
J801	Orange	J203 - ALU Board
J802	Yellow	J204 - ALU Board
J806	Red (2)	J272*- MCP Board
J807	Gray (2)	J268*- MCP Board
J808	Blue	J276 - MCP Board
J508	White (2 molex)	+5V and Ground** - MCP Board

\*Pin one of ten-pin cable connector lines up with pin one of MCP connector and pin one of seven-pin lines up with pin eleven of same MCP connector.

\*\*Molex connector with brown wire connects to ground pins. A different 5V source can be used instead of the system under test.

### OPERATION

You can use the System Test Fixture to do several kinds of operations with a 4052 or 4054. In general the fixture can control the ALU Board by forcing the processor to a micro-code address that you key in, by forcing the data you key in onto the ALU-OUT data bus, or by controlling the ALU's clock.

There are some things to keep in mind while using the fixture:

- If a micro-code address is forced on the bus, the current program sequence may be disturbed. The micro-code address you enter must be a proper entry point in order to establish a new micro-code sequence.
- Some micro-code sequences involving I/O or Rompacks must run at normal speed to produce valid results. In order to single step through these sequences use a jumper to connect the pins of J239 (slow.ALU) on the MAS Board. Be sure to remove this jumper after testing.
- Switches or LEDs that are not labelled on the System Test Fixture have no function in a 4052 or 4054 system.

## LEDs

### ADDRESS SPACE

- DATA LEDs - show whether the ALU data applies to memory space A or B.
- FETCH LEDs - show whether an instruction fetch is from memory space A or B.
- MICROADDRESS LEDs - display the current ALU micro-code address.
- ALU OUT LEDs - display the data from the ALU OUT lines.
- ADDRESS LEDs - display the address in memory that is affected by a READ or WRITE operation using the fixture.

## COMP SWITCH

The COMP switch (Break on COMPare) is located beneath the LED labelled COMP. The COMP switch, when on (down) stops the ALU clock if the micro-code address you key in (see COMP LOAD) is reached. If this happens the COMP LED glows. The COMP switch must be on while using the System Test Fixture to read or write in memory.

## KEYS

The following keys work by controlling the ALU clock:

- STOP - When you press the STOP key the ALU clock stops on the next trailing edge. The STOP key must be pressed before the other keys on the fixture can be used.
- START - Press the START key to make the ALU clock run. After pressing START, the System Test Fixture does not control the ALU. If COMP is on and you have keyed in a micro-code address, the ALU will stop if that address is reached.

## KEYS (cont)

- MICROSTEP - Each time you press the MICROSTEP key, the ALU does one micro-code instruction (the ALU clock makes one cycle).
- MACROSTEP - Each time you press the MACROSTEP key, the ALU does one complete 6800 type of instruction regardless of the number of micro-code instructions needed.

The following keys work by forcing the ALU to go to a given starting place and to then execute a set of micro-code instructions. The set of instructions performs the function. Some of these keys require you to enter hexadecimal data before performing the function. If you instead press a non-hexadecimal key, the previously started function is stopped and the new one started.

- MICRO ADDR - The Force MICRO ADDRess key places the three-digit hexadecimal address in micro-code onto the ALU's address bus. After pressing MICRO ADDR, press three hexadecimal keys to enter the address. You can then microstep, macrostep, or start the ALU.
- REG - Press the REGister key and another hexadecimal key to display the contents of the selected ALU register. There are 16 internal ALU registers. Register data is displayed on the ALU OUT LEDs.
- COMP LOAD - The COMPare LOAD key, followed by entering three hexadecimal digits, picks a stopping point in the micro-code. If the comp switch is on when that micro-code address is reached, the ALU clock stops.
- ADDR - Press the ADDRess key and four more hexadecimal keys to enter a memory address for a READ or WRITE operation. The ADDRESS LEDs show the address you entered.
- READ - To use the READ key, first set the COMP switch to ON, then use ADDR to enter the address you want to read. Press READ to display the contents of the addressed location on the ALU OUT LEDs.
- READ NEXT - The COMP switch must be on for READ NEXT to work. Pressing the READ NEXT key displays data on the ALU OUT LEDs from the next higher location in memory as shown in the ADDRESS LEDs.

KEYS (cont)

- DATA - Press the DATA key and then four hexadecimal keys to enter data for use with WRITE functions.
- WRITE - The WRITE function stores data in a memory location, reads it back, and displays it on the ALU OUT LEDs. To use WRITE, first set the COMP switch to ON, use the DATA key to enter the data you want to store, use ADDR to select the location address, and then press WRITE.
- WRITE NEXT - The WRITE NEXT function increments the memory address to the next location, writes data in that location, reads it back, and displays it on the ALU OUT LEDs. To use WRITE NEXT, first set the COMP switch to ON, use the DATA key to enter the data you want to store, and press WRITE NEXT. If you wish to write the same data to successive locations, enter the data, then press WRITE NEXT repeatedly.
- RESTART - Pressing the RESTART key causes the system under test to begin its restart routine as in power-up.
- INIT - INITIALIZES the System Test Fixture.
- FNCTN - The FUNCTION key is used with other keys to select special functions. The following functions are defined:
- FNCTN - WRITE: Writes the data entered using the DATA key to the program counter in the system under test. Enter the data first.
- FNCTN - READ: Reads the program counter into R5 (Register 5) of the ALU and displays it on the ALU OUT LEDs.
- FNCTN - WRITE NEXT: Writes the contents of R5 into the ALU's condition code register.
- FNCTN-READ NEXT: Reads the condition code register into R5 and displays it on the ALU OUT LEDs.
- FNCTN-MICROSTEP: Complete this function by entering the number of micro-code steps to be done. For example the sequence FNCTN MICROSTEP 5E causes the ALU to do 94 micro-code instructions.

## 4052/54 GENERAL FAULT ANALYSIS

The behavior of the instrument at power-on may help find any hardware malfunctions. At power-on, a number of tests are run both by special micro-code and by performing routine system initialization. The normal power-up sequence will turn off all of Front Panel lights except "POWER," will rewind the tape (if one is inserted) to the nearest beginning of a file, and will display a blinking cursor in the upper left corner of the display area. The screen will generally be flooded shortly after power is first applied. It may be necessary to use the PAGE key to erase the screen in order to observe the blinking cursor. If the blinking cursor cannot be observed, then a hardware failure exists.

The state of the front panel lights gives a little information as to the possible fault. If all lights (BUSY, I/O, and BREAK) are on, the failure could be in the ALU, MCP handshake, or I/O. If only BUSY is on, then a RAM memory error is likely. If all lights (except POWER) are off, the error is anywhere--except RAM memory is not the likely problem. ROM failure is a definite possibility. If the POWER light, as well as the other three are OFF, a blown fuse or power supply failure is likely.

If a blinking cursor is observed after power-on, but a failure in a previously successful operation subsequently occurs, then the error could be almost anywhere. If a standard error message is printed, it may be indicative of the failure: (A) Any error message associated with use of the magnetic tape may indicate a failure or needed adjustment of the tape drive or associated circuits; (B) A ROM failure may be the cause of otherwise inexplicable message; (C) A SYSTEM ERROR could be the result of a ROM failure or a firmware bug or of an intermittent RAM failure.

A memory parity error gives a SYS ERROR, "MESSAGE 41" and is non-fatal (program data are left intact). The most usual cause of the other type of SYS ERROR (without any reference to a Message Number or a Line) is an illegal stack entry resulting from a ROM error or firmware bug. This type of error is fatal. There are, at this time, no known firmware bugs which result in a SYS ERROR.

If the system locks up with none of the keyboard's keys (except the shift key's effect on screen hold-mode) operative, the failure could be anywhere. If there is apparent activity--display lights or cursor blinking, etc., then a F/W bug, ROM or RAM error may be responsible. Otherwise, an internal handshake failure could have occurred. A logic probe can be used to determine if CPU CLK is running to verify this possibility.

## SUMMARY

LIGHTS			LIKELY CAUSE
BUSY	BREAK & I/O	POWER	
OFF	OFF	OFF	Power supply failure
ON	ON	ON	ALU, MCP Handshake, I/O
ON	OFF	ON	Memory
OFF	OFF	ON	ALU, MCP, Optional RAM, ROM or I/O, High Voltages or Display.
			The high voltage, display board, and some of the I/O can be checked by typing "blind" the statement.
			PRI "G" CR
			If the bell rings, the problem is in the DISPLAY function.

## 4052/54 FAULT ISOLATION USING THE MICRO SELF TEST

The micro self test is a partial test of the CPU Board assembly (4 boards). It is intended to detect those faults which, if present, prevent the system from executing firmware code such as the power-up routine in system ROM, or the firmware of the Diagnostic Rompack (067-0900-XX). The System Test Fixture can be used to monitor the micro self test and aid in troubleshooting.

The results of the self test can often be interpreted without the aid of any test equipment. There are five LED (Light Emitting Diodes) and one Front Panel display light (BUSY light) which are used by the self test. Inspection of these LEDs (one on the ALU board and four on the MCP board) is possible by removal of the instrument's top cover. The "MICRO SELF TEST (SUMMARY)" flow chart illustrates how these lights are used to indicate passage of the major sections of the self test.

**NOTE:** Although the major subdivisions of the self test are associated with the various boards of the CPU, it should not be assumed that failure of a given test is necessarily indicative of a failure of the associated board. The failure may involve a signal connecting to some other board.

In some cases, knowing the micro address at which the self test appears to halt will be helpful. The System Test Fixture can display the micro address. Or a logic probe (e.g., Tektronix P6401) can be used to check the micro address lines. These lines are accessible on square pins on the edge of the ALU board closest to the front of the instrument.

The self test starts each time a restart pulse occurs. A restart pulse of several hundred milliseconds is generated when power is applied to the instrument. A restart pulse can also be generated by use of the momentary contact RESTART switch of the Diagnostic Rompack or the System Test Fixture. The RESTART switch of the Diagnostic Rompack is effective independent of the state of the ENABLE/DISABLE switch. Executing the firmware instruction opcode ~~00~~ also causes the self test to be executed.

**CAUTION:** Repeated power cycling of the instrument should be avoided. Instead, use the RESTART switch of the Diagnostic Rompack or System Test Fixture.

There are two modes of operation of the self test, Normal and Debug mode. In Normal mode, any failure detected by the self test will produce a HALT condition. In the Debug mode, a failing test will, in most cases, be repeated. If, in the case of an intermittent fault, a failing test is subsequently passed, the entire self test will be restarted after finishing the current pass of the self test. Thus, even in Debug mode, an intermittent fault would have to appear good two times in succession before control

could pass to any firmware. There are two exceptions to looping on a failing test: 1. The first subtest which checks the ability of the ALU to run micro tests, and 2. the RAM quick-check. These two tests always halt when an error is detected, independent of the Normal/Debug mode.

Debug mode is established using a jumper wire to ground the TESTOPP-0 signal at J204-8.

#### LOCATION OF LIGHTS & LEDs

HALF-CARRY LED  
(DS190, ALU BOARD)

This LED is located on the right-hand side of the instrument on the side edge of the ALU board. It is the rear most of the two LEDs located there.

OVERFLOW LED  
(DS195, ALU BOARD)

The forward LED, DS195 displays the state of the overflow bit of the condition code register.

PROGRAM COUNTER LEDs

The most significant (high-order) 4 bits of the program counter are displayed in 4 LEDs located on the left-side of the instrument on the side edge of the MCP board. The most significant bit is DS<sub>4</sub> located closest to the front of the instrument.

MEMORY PARITY ERROR

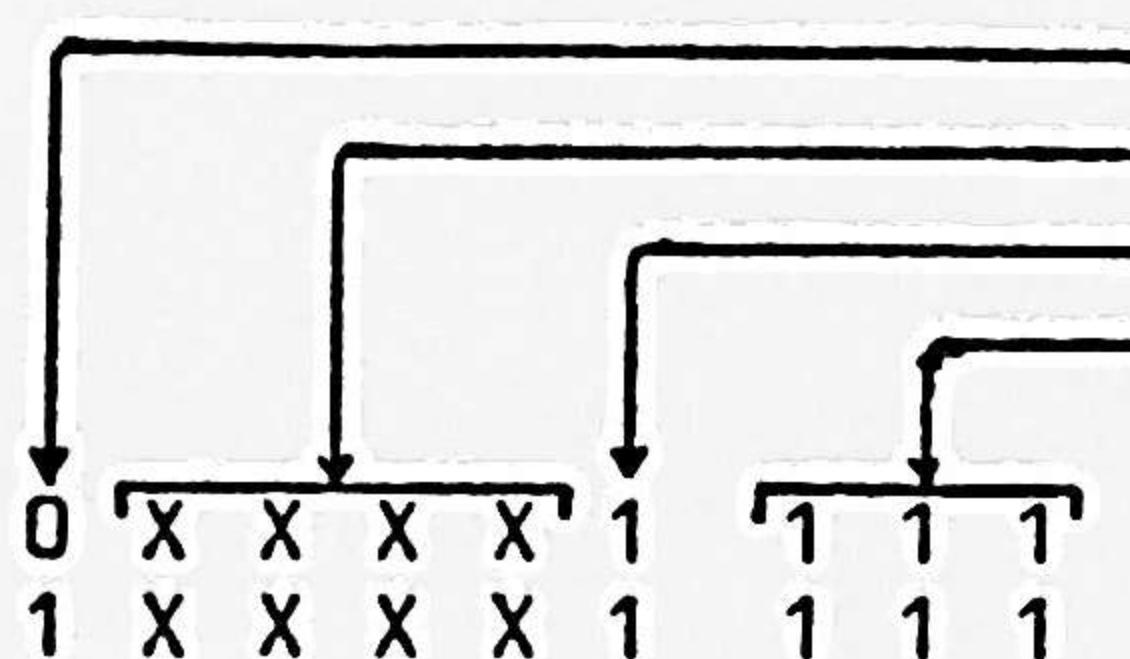
This LED is located on the left-side of the instrument on the side of the MAS BOARD.

BUSY  
BREAK  
I/O  
POWER

These lights are located just to the right of the CRT face plate.

#### ACTION OF LIGHTS & LEDs DURING SELF TEST

The table on the next page shows how to interpret the results of the self test through various stages.



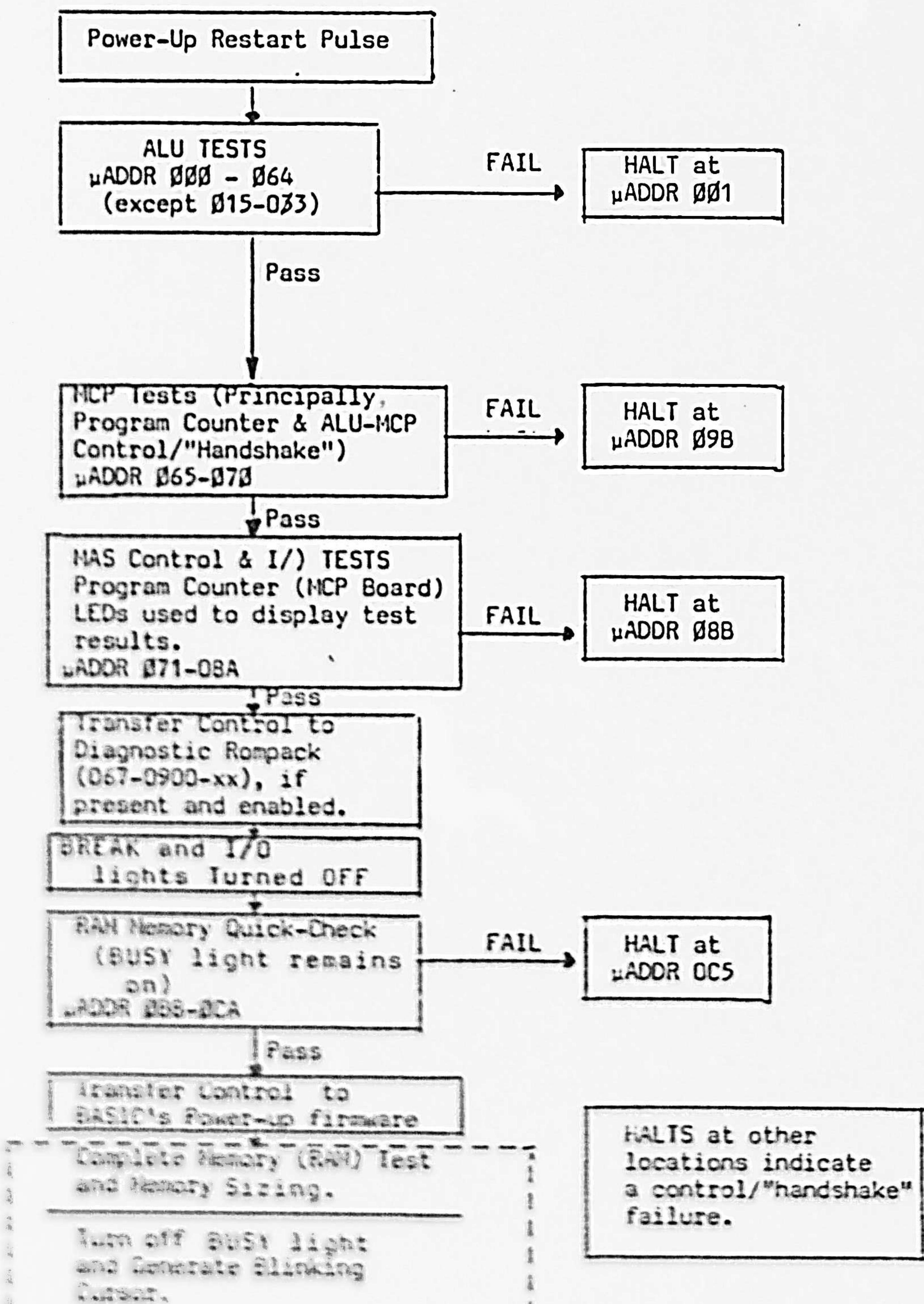
Half Carry DS190  
Prog. Counter DS4, DS3, DS2, DS1  
Busy  
Break, I/O, & Power

							<u>Comments</u>		
0	X	X	X	X	1	1	During, and immediately after restart. Test is checking ALU.		
1	X	X	X	X	1	1	Any change in the Program Counter LEDs since RESTART indicates that ALU portion of the SELF TEST is successful.		
1	0	0	0	0	1	1	During the MCP test, this sequence will appear in the program counter as a means of testing each bit of the 16 bit data path out of and into the ALU. The last two patterns occur while testing the count operation of FFFF+1→0000.		
1	0	0	0	1	1	1			
1	0	0	1	0	1	1			
1	0	1	0	0	1	1			
1	1	0	0	0	1	1			
1	1	1	1	1	1	1	If the above tests are passed, then the Program Counter LEDs are used to indicate the next set of tests.		
							In each of the following five tests, failure at a given test is indication that all preceeding tests were passed.		
1	0	0	1	1	1	1	Means PC test passed. If halted (NO CPUCLK), it means that there was a failure in ALU-MCP-MAS handshake during a write operation to RAM memory.		
1	0	1	0	1	1	1	Ram write handshake works. If halted, read RAM handshake has failed.		
1	0	1	1	0	1	1	RAM read handshake works. If halted, there was a handshake failure during a write at an I/O address.		
1	0	1	1	1	1	1	Write I/O handshake works. If halted, read I/O handshake failure has occurred.		
1	1	0	0	1	1	1	Read I/O handshake works. If halted, then the I/O data validity check has failed.		
1	1	0	0	1	0	1	The busy light is turned off if the diagnostic Rompack is not entered and all tests so far have been passed.		
							The parity error is cleared and disabled, and quick check of memory begins. If this quick check fails, it halts at 0C5. Otherwise, control is passed to the power up firmware of BASIC.		
1	X	X	X	X	1	0	0	1	BASIC does a complete check of memory. If failure is detected, it will never go on to its initialization phase.
1	X	X	X	X	0	0	0	1	BASIC's initialization in progress or completed and is blinking the cursor. Activity in some of the LEDs (indicated by "B"), including the overflow LED, DS190, at the same rate as the cursor blink can be observed.
B	B	1	1	B	0	0	0	1	

**LEGEND:**

- 0= LIGHT IS OFF**
- 1= LIGHT IS ILLUMINATED**
- X= CONDITION OF LIGHT IS UNPREDICTABLE  
OR IS RAPIDLY CHANGING**
- B= OBSERVABLE BLINKING ACTIVITY**

## MICRO SELF TEST (SUMMARY)



## 4052/54 MICRO-CODE LISTING

The following pages give the micro-code contained in ROM on the ALU Board. The listing shows the micro-code address and ROM data according to the following format:

		Line # of code listing
		Micro-code address
1	000	U330 (MSB=pin 17)
80		U315
20		U300
00		U340
98		U335
50		U320
D1		U305
DD		Label used in comments
		*Restart



```

12 00B 80 10 02 9F C0 40 00*      *;+ If N is not set, goto to 001 (and HALT).
13 00C 80 00 80 74 00 40 00*      *;+ 0->N. Try to clear the N bit.
14 00D 80 10 02 1F C0 40 00*      *;+ If the N bit is set, goto 001 (and halt).
15 00E 83 60 00 9F C0 40 00*      *
*;+ Goto to 036 to skip over the interrupt vectors and the system
*;+ test fixture code.
*;+-
*
*
*;+ The following 16 branches perform priority encoding and
*;+ dispatching for interrupts. The alu forces a microbranch
*;+ to address 01X on interrupts, where 'X' is the logical 'OR' of all
*;+ pending interrupt conditions. A weight of 1 is given to
*;+ memory parity errors, a weight of 2 is given to NMI, a weight
*;+ of 4 to IRQ, and a spare, unused interrupt is given a weight
*;+ of 8. When several interrupts are pending a branch to the highest
*;+ priority interrupt is taken.
*;+-
*
*
*
*
16 010 8C E0 00 9F C0 40 00*      *;+ X is 0, there are no interrupts, and we should not have gotten here,
*;+ so goto the invalid instruction trap.
17 011 90 B0 00 9F C0 40 00*      *;+ X is 1, it's a parity error alone, goto to parity error trap.
18 012 8F 40 18 9F C0 40 00*      *;+ X is 2, NMI only, so goto NMI routine.
19 013 90 B0 00 9F C0 40 00*      *;+ X is 3, parity error and NMI together, so take the higher-priority
*;+ parity error and goto the parity error routine.
20 014 8F 20 00 9F C0 40 00*      *;+ X is 4, IRQ alone, so goto the IRQ routine.
21 015 90 B0 00 9F C0 40 00*      *;+ X is 5, IRQ and parity error, parity wins.
22 016 8F 40 18 9F C0 40 00*      *;+ X is 6, IRQ and NMI, NMI wins.
23 017 90 B0 00 9F C0 40 00*      *;+ X is 7, parity error, NMI and IRQ--parity wins.
24 018 90 90 00 9F C0 40 00*      *;+ X is 8, the unused interrupt has been used, goto DSPINT to
*;+ dispose of the unused interrupt.
25 019 90 B0 00 9F C0 40 00*      *;+ In the next eight instructions, the unused interrupt is
*;+ always present, but it is lower priority than any of the
*;+ other interrupts, and will never win here, so it will not
*;+ be mentioned again.
*
*;+ Parity error and you-know-what, parity error wins.
26 01A 8F 40 18 9F C0 40 00*      *;+ NMI and um-um, take NMI.
27 01B 90 B0 00 9F C0 40 00*      *;+ NMI and parity, take parity.
28 01C 8F 20 00 9F C0 40 00*      *;+ IRQ only, take it.

```

24 01D 90 60 00 9F C0 40 00\*  
 \*;+ IRQ and parity, parity always wins!  
 30 01E 4F 40 18 9F C0 40 00\*  
 \*;+ IPO and NMI, NMI wins.  
 31 01F 90 60 00 9F C0 40 00\*  
 \*;+ X=15. IPO, NMI, and Parity, parity wins.  
 •  
 •  
 •  
 •  
 •  
 •;+ Display registers for system test fixture  
 •  
 •  
 •  
 32 020 82 00 00 9F C0 44 00\*DPU:  
 \*;+  
 33 021 82 10 00 4F C0 44 01\*DR1:  
 \*;+  
 34 022 82 20 00 9F C0 44 02\*DPU2:  
 \*;+  
 35 023 82 30 00 9F C0 44 03\*DPU3:  
 \*;+  
 36 024 82 40 00 9F C0 44 04\*DPU4:  
 \*;+  
 37 025 82 50 00 9F C0 44 05\*DPU5:  
 \*;+  
 38 026 82 60 00 9F C0 44 06\*DPU6:  
 \*;+  
 39 027 82 70 00 9F C0 44 07\*DPU7:  
 \*;+  
 40 028 82 80 00 9F C0 44 08\*DPU8:  
 \*;+  
 41 029 82 90 00 9F C0 44 09\*DPU9:  
 \*;+  
 42 02A 82 A0 00 9F C0 44 0A\*DPU10:  
 \*;+  
 43 02B 82 B0 00 9F C0 44 0B\*DPU11:  
 \*;+  
 44 02C 82 C0 00 9F C0 44 0C\*DPU12:  
 \*;+  
 45 02D 82 D0 00 9F C0 44 0D\*DPU13:  
 \*;+  
 46 02E 82 E0 00 9F C0 44 0E\*DPU14:  
 \*;+  
 47 02F 82 F0 00 9F C0 44 0F\*DPU15:  
 \*;+  
 •  
 •  
 •;+ Next two instructions are used by the system test fixture  
 •;+ to write a memory location. The microcode specifies  
 •;+ "R0 to memory address" and "R0 to (or from) memory data" but the  
 •;+ test fixture overrides the alu outputs and substitutes its  
 •;+ own values for data and address.  
 •;  
 48 030 93 10 40 9F C0 44 00\*DPU16:  
 \*;+ R0 to write address  
 49 031 91 20 40 9F C0 44 00\*DPU17:  
 \*;+ R0 to memory data

```

      *
      *
50 032 83 30 40 9F CC 44 00*READ:
      *;+    R0 to memory address
51 033 00 00 00 77 CA 47 00*RDNEXT:
      *;+    data from memory
      *
      *
      *
      *;+    continue general alu arithmetic diagnostics
      *
      *
52 034 80 10 38 1F C0 40 00*GA2:
      *;+    The several tests below come here when an error is detected.
      *;+    If the 'debug' jumper is not in place, goto 'ALUBAD'
      *;+    and HALT. Otherwise go to the next statement...
      *;+    ...which sets R13 to -1 to indicate that at least one failure
      *;+    was detected. Then fall through to repeat these tests.
53 035 83 60 00 9F C0 D1 DD*
      *;+    Start here. Clear R0 by subtracting it from itself. Set condition codes.
      *;+    System test fixture can be used to start this test here.
54 036 83 70 00 9B 50 D1 00*GA0:
      *;+    If the 'zero' condition is not true, go to GA2 (error).
      *;+    Subtract 1 from R0 (result should be negative) and set cond. codes.
55 037 83 40 08 9B 40 CC 00*GA1:
      *;+    Carry should be cleared, if not go to GA2 (error).
      *;+    If not negative result, goto to GA2 (error).
      *;+    If "zero" is set goto GA2 (error).
      *;+    Add 1 back to R0 and set condition codes.
56 038 83 40 01 1F C0 40 00*
      *;+    If result is not zero now, goto GA2 (error).
57 039 83 40 02 9F C0 40 00*
      *;+    If result is negative, goto GA2 (error)
58 03A 83 40 08 1E 50 C4 00*
      *;+    If carry is not set, goto GA2 (error).

      *
      *
      *
      *;+    The following tests check for stuck or lin-to-line shorts
      *;+    on ALU in.
62 03E 83 F0 00 9F C0 40 00*SHORT0:
      *;+    Test fixture can start test here. (Dummy goto).
63 03F AA AA 80 77 C0 C7 00*SHORT1:
64 040 55 55 80 77 C0 C7 10*
      *;+    Load an alternate 1-0 pattern in R0 and its complement
      *;+    in R1.
65 041 80 00 00 77 C0 C1 11*
      *;+    Double R1 to shift it left.
66 042 80 00 00 72 00 71 01*
      *;+    EOR R1 and R0 and set cond. codes. Result should be zero.
67 043 84 80 08 1F C0 40 00*
      *;+    If result is zero, goto to register tests.
68 044 80 10 38 1F C0 40 00*
      *;+    Error was detected. If debug jumper is not on, goto ALUBAD.
69 045 83 F0 00 9F C0 D1 DD*

```

```

*;+      Otherwise, set R13 to -1 and repeat the test.
*
*
*
70 046 80 10 38 1F C0 40 00*REGBAD:
*;+      Register test failures come here. If debug jumper is on
*;+      repeat the test, otherwise, goto ALUBAD.
71 047 84 80 00 9F C0 D1 DD*
*
72 048 88 C0 00 8F C0 40 00*REG1:
*;+      Call a subroutine to put 0000 in R0, 1111 in R1, 2222 in R2,
*;+      4444 in R4, 8888 in R8, and FFFF in all other registers (except R13).
73 049 80 00 00 77 C0 D9 10*
*;+      EUR R0 and R1.
74 04A 11 11 80 72 00 75 01*
*;+      Result should be 1111.
75 04B 84 60 08 9F C0 40 00*
*;+      If not, goto REGBAD above.
*;+      Perform the test with R1 and R2, R2 and R4,
*;+      R4 and R8, and R8 and R0. Any failures indicate
*;+      a possible short in the register A&B lines of the
*;+      2901, a bad 2901, or a bad microcode bit.
*
76 04C 88 C0 00 8F C0 40 00*
77 04D 80 C0 00 77 C0 D9 21*
78 04E 33 33 80 72 00 75 02*
79 04F 84 60 08 9F C0 40 00*
*
80 050 88 C0 00 RF C0 40 00*
81 051 80 00 00 77 C0 D9 42*
82 052 66 66 80 72 00 75 04*
83 053 84 60 08 9F C0 40 00*
*
84 054 88 C0 00 RF C0 40 00*
85 055 80 00 00 77 C0 D9 R4*
86 056 CC CC 80 72 00 75 08*
87 057 84 60 08 9F C0 40 00*
*
88 058 88 C0 00 8F C0 40 00*
89 059 80 00 00 77 C0 D9 08*
90 05A 88 88 80 72 00 75 00*
91 05B 84 60 08 9F C0 40 00*
*
*
*
*
92 05C 85 D0 00 9E 90 51 55*P0:
*;+      The "P" (condition code) register is outside the 2901's.
*;+      Test the ALU's ability to communicate with it.
*;+
*;+
*;+
93 05D 00 00 81 72 C0 47 00*P1:
*;+      Generate a 0 (R5-R5) and stuff it in P.
*;+      Test fixture can start here.
*
94 05E 86 30 08 9F C0 40 00*
*;+      Pass P back to the 2901's and set the zero bit
*;+      according to the result.
95 05F 80 00 84 74 C0 40 00*
*;+      Go to error      ' ' the zero condition is not set now.
*;+      Force the I      SK on.

```

```

96 060 00 00 81 77 C0 C7 50*      *;+ Get P back in RS.
97 061 00 14 80 72 C0 75 05*      *;+ EOR RS with hex 14 and set condition codes.
98 062 86 50 08 1F C0 40 00*      *;+ If RS was 14 (result of EUR is zero) go to next test (PC).
99 063 80 10 38 1F C0 40 00*PBAD: *;+ Error was detected. If debug jumper is not on, goto ALUBAD.
100 064 85 C0 00 9F C0 D1 DD*      *;+ Otherwise, repeat the test.
*;+
*;+
*;+ ALU tests have determined that the ALU is reasonably free of
*;+ faults. Testing of the rest of the boards begins.
*;+
*;+
*;+
*;+
*;+
*;+
*;+
*;+ Begin test of PC on MCP. Dummy goto next statement so
*;+ test fixture can start the test here.
*;+
*;+
*;+
*;+ Load a single bit into RS.
*;+
*;+ Transfer RS to the PC.
*;+
*;+ EOR RS and PC. Result should be zero.
*;+
*;+ If the result is not zero, goto PCBAD. Shift RS left
*;+ to ripple the single bit through all 16 PC bits.
*;+
*;+ Go repeat the test with the next bit if no bit has
*;+ shifted into the carry bit.
*;+
*;+ Continue PC test. Load -1 (all ones) into PC.
*;+
*;+ Increment the PC by 1 via the increment circuitry
*;+ on the MCP.
*;+
*;+ Read the PC back. It should be zero now.
*;+
*;+ If PC is now zero, go to memory tests.
*;+
*;+ PC increment or load error. If debug jumper is on,
*;+ repeat the test otherwise goto MPCBAD.
*;+
*;+
*;+ Any hangup problems in the above tests indicate an ALU-MCP
*;+ handshake problem.
*;+
*;+ Begin test of memory function.
*;+
*;+
*;+
*;+
*;+
*;+
*;+
*;+ Dummy goto next statement for system test panel start.
*;+
*;+ MEMWRT:

```

			;;+	Store 3 in top 4 bits of PC (LED's on MCP board). PC is determined by preceding tests to be reliable enough to use as progress indicator for the following tests. Note value in PC LED's if a hang occurs because of ALU-MCP-MAS handshake problems.
115	073	00 01 80 A7 00 47 00	*PUMP:	
116	074	40 00 C0 77 CE 47 00		
117	075	00 00 40 6F CA 47 00		
			;;+	The above three instructions are a loop executed 15 times. Data of zeroes is written to memory address 4000 to pump up internal bias voltages in memory chips. Handshake problems may cause the loop to hang.
			;;+	
118	076	97 70 00 4F C0 40 00		Dummy goto for system test fixture startup.
119	077	50 00 80 77 C6 47 00	*MEMRD:	Put 5 in PC LED's to show progress.
120	078	40 00 C0 77 CC 47 00		Put 4000 to memory write address.
121	079	00 00 00 77 CA 47 00		Read memory.
122	07A	50 00 80 77 C6 47 00		Put 6 to PC to indicate previous read did not hang.
			;;+	
			;;+	Begin tests of I/O
			;;+	
123	07B	67 C0 00 9F C0 40 00		Dummy goto next statement for system test fixture. Test I/U by rippling a single 1 bit thru the data direction register of a PIA.
124	07C	FF 0F 80 77 CE 47 00	*I00:	Clear the control register of PIA at FFUF
125	07D	00 00 80 77 CA 47 00		
126	07E	00 01 80 77 C0 C7 00		Initialize the loop with a single "1" bit in R0.
			;;+	
127	07F	FF 0E 80 77 CE 47 00	*I01:	Write R0 to the data direction register.
128	080	80 00 00 77 C8 44 00		
129	081	70 00 80 77 C6 47 00		Put 7 in the PC LED's to show we haven't hung yet.
130	082	FF 0E 80 77 CC 47 00		
131	083	00 00 00 72 C6 75 00		Read the PIA data direction back and XOR it with R5
132	084	90 00 80 77 C6 47 00		Put 9 in PC LED's.
133	085	88 90 08 3F C0 40 00		If the PIA and R5 didn't match, goto IOBAD.
134	086	80 00 00 72 C0 C1 00		Shift R5, the test bit, left one place.
135	087	87 F0 08 9F C0 40 00		If there's still a bit in R5 (lower 8 bits) continue the test.
136	088	89 C0 00 9F C0 40 00		Test passed, goto DIAGCOUNT to skip over some stuff.
			;;+	
137	089	88 80 38 1F C0 40 00	*IOBAD:	Goto IOBAD1 if debug jumper not connected.
			;;+	



```

165 0A5 00 04 80 77 CA 47 00*
    *;+ Initialize the PIA used for the 4052/4054 identity
    *;+ determination.
166 0A6 80 00 00 76 00 40 00*
167 0A7 00 0C C0 77 CC 47 00*
168 0A8 00 00 00 77 CA C7 00*
169 0A9 0B 52 80 72 00 75 00*
    *;+ Look for "DB52" in location C of the rompack
    *;+ to see if it's the diagnostic rompack.
170 0AA 88 00 00 9F C0 40 00*
    *;+ Go to the regular restart at NODEBUG1 if diagnostic
    *;+ rompack is not in place.
171 0AB 00 F0 80 77 C0 E5 0D*
    *;+ Set accumulator A to "F0" if any errors occurred.
    *;+ (As noted in R13).
172 0AC 00 2E C0 77 CC 47 00*
173 0AD 00 00 00 77 CA C7 40*
174 0AE 00 F0 80 76 80 47 00*
175 0AF 80 00 50 7F C6 44 04*
    *;+ Get the entry point to the diagnostic rompack
    *;+ from location 000E and go there.
    *
    *
176 0B0 80 00 00 72 00 44 CD*NODEBUG1:
    *;+ This is the regular (no diagnostic rompack)
    *;+ restart point. Test R13 to see if any error occurred during
    *;+ microdiagnostics.
177 0B1 80 00 08 9F C0 40 00*
    *;+ If any errors were found, go to RESTART and repeat
    *;+ all test.
    *
    *
178 0B2 FF 26 60 77 C8 47 00*
179 0B3 00 00 60 77 C8 47 00*
180 0B4 FF 2A C0 77 C8 47 00*
181 0B5 F0 04 80 77 CA 47 00*
182 0B6 FF 2A 80 77 C8 47 00*
183 0B7 00 60 80 77 C8 47 00*
    *;+ Turn off the busy light and begin a more complete test
    *;+ of the basic (32K) ram.
    *
184 0B8 88 90 00 9F 10 D1 55*
    *
185 0B9 80 00 40 77 DE 44 05*MEM1:
186 0BA 00 03 80 77 CA 85 55*
187 0BB 7F FE 80 72 00 75 05*
188 0BC 88 90 08 9F C0 40 00*
    *;+ The above four instructions store values at
    *;+ ram addresses 1,4,7,10 (incrementing by 3). The value
    *;+ stored is the address plus one.
    *
189 0BD 88 E0 00 9F D0 D1 55*
    *
190 0BE 80 00 40 77 DC 44 05*MEM2:
191 0BF 00 00 00 72 0A F5 65*
    *;+ Read the locations and compare to what should
    *;+ be stored.
192 0C0 AC 60 08 1F C0 40 00*
    *;+ Go to ME. : no error is found.

```

```

193 0C1 80 00 00 77 D6 44 05*      *
194 0C2 80 00 00 77 C6 44 06*      *;+ Write out failing address to PC
195 0C3 80 00 00 77 C0 D1 DD*      *;+ Write failing bits to PC (bad bits are ones).
196 0C4 8C 60 38 9F C0 40 00*      *;+ Put -1 in R13 to indicate at least one error occurred.
197                                     *;+ If the debug jumper is in place, continue
198                                     *;+ with the test.
199                                     *
200                                     *
201 0C5 0C 50 00 9F C0 47 00*MEMBAD:
202                                     *;+ Otherwise, hang, bad memory.
203                                     *
204                                     *
205                                     *
206 0C6 00 03 80 77 C0 C5 55*MEM3:
207 0C7 7F FE 80 72 00 75 05*      *
208 0C8 8B E0 08 9F C0 40 00*      *
209 0C9 80 00 00 72 00 44 0D*      *;+ Increment the address, test for end, and go back
210 0CA 8C 50 08 9F C0 40 00*      *;+ to MEM2 if not finished.
211 0CB FE FE C4 74 CC 47 00*      *;+ Test R13 to see if any errors occurred. If so, goto MEMBAD.
212 0CC 00 00 00 77 4A C7 40*      *
213 0CD 80 00 50 7F C6 44 04*      *;+ Fetch the firmware restart address from FEFE and start BASIC.
214                                     *
215                                     *
216                                     *
217                                     *
218                                     *
219                                     *
220                                     *
221                                     *
222                                     *
223                                     *
224                                     *
225 0CE 10 D0 81 9F C0 C7 40*TRAP:
226                                     *
227 0CF 00 00 50 7F C8 C7 00*PULACONT:
228 0D0 00 00 50 7F C8 C7 10*PULBCONT:
229 0D1 80 00 50 7F C 44 00*PSHACONT:
230 0D2 80 00 50 7F C8 44 01*PSHBCONT:
231                                     *
232                                     *
233 0D3 00 00 00 77 CA C7 40*RTSCONT:
234 0D4 80 00 50 7F D6 83 34*      *
235 0D5 80 00 00 77 CC 44 04*NEGCCONT:
236 0D6 00 00 00 70 98 CF 50*      *
237 0D7 80 00 00 77 CE 44 04*STUFFIT:
238 0D8 80 00 50 7A C8 44 05*      *
239                                     *
240 0D9 80 00 00 77 CC 44 04*COMCONT:
241 0DA 0D 70 00 99 08 FF 50*      *
242                                     *
243 0DB 80 00 00 77 CC 44 04*DECCONT:
244 0DC 0D 70 00 98 C8 D7 50*      *
245                                     *
246 0DD 80 00 00 77 CC 44 04*INCCONT:
247 0DE 0D 70 00 98 D8 C7 50*      *
248                                     *
249 0DF 80 00 00 77 CC 44 04*LSRCONT:
250 0E0 0D 70 00 98 09 47 50*      *
251                                     *
252 0E1 80 00 00 77 CC 44 04*RORCONT:

```

226 0E2 0D 71 00 9A 89 47 50\*  
 227 0E3 80 00 00 77 C0 44 04\*ASRCONT:  
 228 0E4 00 00 00 70 88 C7 50\*  
 229 0E5 8D 72 00 98 81 44 55\*  
 230 0E6 80 00 00 77 C0 44 04\*ASLCONT:  
 231 0F7 00 00 00 77 C8 C7 50\*  
 232 0E8 80 70 00 98 80 C1 55\*  
 233 0E9 80 00 00 77 C0 44 04\*RULCONT:  
 234 0EA 00 00 00 77 C8 C7 50\*  
 235 0EB 8D 70 00 98 A0 C1 55\*  
 236 0EC 00 00 50 76 88 47 00\*TSTCONT:  
 237 0ED 80 00 50 79 5A 51 00\*CLRIXCON:  
 \*  
 238 0EE 80 02 00 73 A1 44 00\*ASRACONT:  
 239 0EF 80 00 50 7A C0 44 00\*SETZA:  
 240 0F0 80 02 00 73 91 44 11\*ASPBCONT:  
 241 0F1 80 00 50 7A C0 44 01\*SETZB:  
 \*  
 \*  
 242 0F2 FE F2 60 77 C0 C7 50\*ING:  
 243 0F3 8F 50 00 9F C0 40 00\*  
 244 0F4 FE FC 80 77 C0 C7 50\*H4I:  
 245 0F5 80 00 83 77 C0 40 00\*CLINT:  
 246 0F6 00 00 61 77 00 C7 60\*STACK:  
 247 0F7 00 06 84 74 DE 40 03\*  
 248 0F8 80 00 00 77 C8 44 06\*  
 249 0F9 00 00 00 77 C4 C7 40\*  
 250 0FA 80 00 40 77 C0 44 05\*  
 251 0FB 00 00 60 77 C4 C7 70\*  
 252 0FC 80 00 40 77 C0 40 03\*  
 253 0FD 60 00 00 77 C4 44 04\*  
 254 0FE 00 03 C0 77 DE 40 03\*  
 255 0FF 80 00 00 77 C4 44 02\*  
 256 100 00 04 80 77 DE 40 03\*  
 257 101 80 00 00 77 C8 44 00\*  
 258 102 00 05 60 77 DE 40 03\*  
 259 103 40 00 00 77 C8 44 01\*  
 260 104 00 07 80 77 D0 C0 33\*  
 261 105 60 00 00 73 40 44 0F\*  
 262 106 91 A0 02 1F C0 40 00\*  
 263 107 00 D0 90 76 80 50 05\*  
 264 108 80 00 50 7F C6 44 07\*  
 \*  
 \*  
 \*  
 265 109 FE F4 80 77 C0 C7 50\*DSPINFO:  
 266 10A 8F 60 00 9F C0 40 00\*  
 267 10B FE F2 80 77 C0 C7 50\*HWFAIL:  
 268 10C 9F 50 00 9F C0 40 00\*  
 \*  
 \*  
 269 10D FE FA 80 77 C0 C7 50\*SWICONT:  
 270 10E 8F 60 00 9F C0 40 00\*  
 271 10F 00 00 00 77 C8 C7 50\*RTSCOUNT:  
 272 110 80 00 00 77 DC C4 33\*  
 273 111 00 00 00 77 C8 C7 10\*  
 274 112 80 00 00 77 DC C4 33\*  
 275 113 00 00 00 77 C8 C7 00\*  
 276 114 80 00 40 77 DC C4 33\*

277	115	00 00 00 77 C8 C7 200
278	116	00 02 C0 77 C8 C7 330
279	117	00 00 00 77 C8 C7 400
280	118	00 00 C0 77 01 04 030
281	119	26 00 50 7F C6 01 340
		•
		•
282	11A	F7 03 00 77 C8 C7 000ALT1ECC71
283	11B	30 29 66 77 C8 C7 000
284	11C	00 00 00 77 02 00 000
285	11D	FL F2 C0 77 0C 67 190
286	11E	00 00 00 77 C8 C7 400
287	11F	00 00 00 76 02 04 060
288	120	30 00 50 7F C8 01 460
		•
		•
289	121	00 00 00 77 C8 C9 000BSNC0111
290	122	00 00 00 77 C8 C7 000J3F14C001
291	123	40 00 63 77 C8 C7 330
292	124	26 00 60 77 C8 64 060
293	125	00 00 00 78 C8 01 320
		•
294	126	00 C0 60 78 C8 01 000174C0111
295	127	20 3F 02 77 C8 69 00036PC0111
296	128	00 C0 02 77 C8 65 000
297	129	00 00 00 76 02 00 000
298	12A	00 12 30 78 C8 01 340
		•
		•
299	12B	30 00 50 79 50 20 0003714AC0111
300	12C	30 00 50 79 50 40 000C074AC0011
301	12D	50 00 50 79 70 20 00038C8AC0111
302	12E	30 00 50 78 68 63 000ALS4AC0111
303	12F	00 00 50 78 68 63 0000174AC0111
304	130	00 30 96 78 68 C7 000LCS4AC0111
305	131	00 20 90 78 64 F9 000LUM4AC0111
306	132	20 40 90 78 68 29 000ARC4AC0111
307	133	30 00 90 78 64 00 000CHA4AC0111
308	134	00 00 90 79 60 C9 000ADG4AC0111
309	135	00 30 90 78 50 40 020CP4AC0111
310	136	30 00 61 77 C8 C7 000CP4C0111
311	137	00 01 00 78 69 73 000
		•
312	138	00 00 50 78 94 C7 000LGBAC0111
		•
313	139	00 00 50 79 58 C0 000SHP4AC0111
314	13A	20 00 50 79 56 40 000C409AC0111
315	13B	00 00 50 79 78 C0 000SAC9AC0111
316	13C	00 00 50 78 68 63 000ALG8AC0111
317	13D	00 00 50 78 68 65 0000178AC0111
318	13E	00 00 50 78 64 C7 000LCS8AC0111
319	13F	00 00 50 78 64 F9 000LGB8AC0111
320	140	00 00 50 78 68 C9 000ACG8AC0111
321	141	00 00 50 74 64 00 000CHA8AC0111
322	142	00 00 50 78 68 C9 000ADG8AC0111

323 143 00 00 50 7F 04 C7 20•LDXXCON:  
 324 144 80 00 50 7A 0A 44 03•STSAC74:  
 325 145 80 00 50 7A 0A 44 02•STXXCON:  
 326 146 80 00 50 7A 48 44 01•STAAXCON:  
 327 147 80 00 50 7A 48 44 00•STAAXCON:  
 \*  
 328 148 00 FF 80 77 C0 E5 000  
 329 149 80 00 40 77 C0 11 200  
 330 14A 00 00 50 7A 04 C7 200  
 \*  
 331 14B 00 FF 80 77 C0 E5 000•YLUXCON:  
 332 14C 80 00 00 77 C0 41 200  
 333 14D 00 00 50 7A 48 C7 000  
 \*  
 334 14E 00 FF 80 77 C0 E5 000•YLUXCON:  
 335 14F 80 00 00 77 C0 41 200  
 336 150 00 00 50 7A 48 C7 100  
 \*  
 337 151 00 FF 80 77 C0 E5 000•STAXCON:  
 338 152 80 00 00 77 C0 41 200  
 339 153 80 00 50 7A 48 14 010  
 \*  
 340 154 00 FF 80 77 C0 E5 000•JTPAXCON:  
 341 155 80 00 50 7F C4 11 200  
 \*  
 342 156 95 00 00 8F C0 10 00•FPMCHINT:  
 343 157 95 10 00 FF C0 10 000  
 344 158 80 00 50 7F C0 40 000  
 \*  
 345 159 00 00 C0 77 C0 45 04•FPMCHINT:  
 346 15A 00 00 00 77 C0 C7 000  
 347 15B 00 01 C0 77 C0 45 040  
 \*  
 348 15C 00 00 00 77 C0 C7 000  
 349 15D 00 02 C0 77 C0 45 040  
 350 15E 00 00 00 77 C0 C7 700  
 351 15F 80 00 00 77 C0 44 040  
 352 160 00 00 00 07 C0 C7 800  
 \*  
 353 161 40 00 80 77 C0 4C 03•FPMCHINT:  
 354 162 80 00 00 77 C0 44 050  
 355 163 00 03 C0 77 C0 40 030  
 356 164 80 00 00 77 C0 44 060  
 357 165 00 05 C0 77 C0 40 030  
 358 166 80 00 00 77 C0 44 070  
 359 167 00 07 C0 77 DF 40 030  
 360 168 80 00 00 77 C0 44 060  
 361 169 00 08 80 77 C0 44 040  
 362 170 96 10 00 8E 80 44 040  
 363 171 00 08 80 77 C0 45 040  
 364 172 80 00 50 7F C0 40 000  
 \*  
 371 173 00 00 00 77 D0 C4 33•FPMULCONT

372	174	00 0C 80 A7 C0 47 00*	
373	175	80 00 40 77 DC C4 33*	
374	176	00 00 00 77 CA C7 50*	
375	177	80 00 40 77 CE 44 04*	
376	178	80 00 00 77 DA 83 35*	
377	179	00 02 A0 6F C0 C5 44*	
		*	
378	17A	80 00 50 7F C0 40 00*	
		*	
379	17B	00 0B 80 A7 C0 47 00*FDUPCONT:	
380	17C	00 08 C0 77 CC 45 03*	
381	17D	00 00 00 77 CA C7 50*	
382	17E	80 00 40 77 CE 4C 03*	
383	17F	80 00 00 77 CA 44 05*	
384	180	00 02 A0 6F D0 CD 33*	
		*	
385	181	80 00 50 7F D0 C4 33*	
		*	
386	182	00 02 80 77 C0 C5 43*FSWPCONT:	
387	183	00 0C 80 A7 C0 47 00*	
388	184	80 00 40 77 CC 44 04*	
389	185	00 00 00 77 CA C7 50*	
390	186	00 09 C0 77 CC 45 04*	
		*	
391	187	00 00 00 77 CA C7 60*	
392	188	00 09 C0 77 CE 45 04*	
393	189	80 00 00 77 CA 44 05*	
394	18A	80 00 40 77 CE 44 04*	
395	18B	80 00 00 77 CA 44 06*	
396	18C	00 02 A0 6F C0 C5 44*	
		*	
397	18D	80 00 50 7F C0 40 00*	
398	18E	A2 30 00 8F C0 40 00*FMULCON:	
399	18F	80 00 00 77 C0 C1 01*	
400	190	04 00 80 77 D0 CD 00*	
401	191	00 00 80 77 C0 C7 E0*	
402	192	00 00 80 77 C0 C7 D0*	
403	193	00 00 80 77 C0 C7 C0*	
404	194	99 80 00 88 40 C4 46*	
405	195	99 80 00 88 40 C4 47*	
406	196	99 80 00 88 40 C4 48*	
407	197	9F E0 00 9F C0 40 00*	
		*	
408	198	9A 30 08 1F C0 40 00*SHIFTADD:	
409	199	00 00 80 A7 C0 47 00*	
410	19A	80 04 00 73 01 44 44*	
411	19B	99 F0 01 9F C0 40 00*	
412	19C	80 00 00 73 40 C1 C9*	
413	19D	80 00 00 73 60 C1 DA*	
414	19E	80 00 00 73 60 C1 EB*	
		*	
415	19F	80 05 00 72 81 44 EE*	SHFR7:
416	1A0	80 05 00 72 81 44 DD*	
417	1A1	80 05 20 6A 81 44 CC*	
		*	
418	1A2	80 00 00 D7 C0 40 00*	
		*	
419	1A3	80 00 00 77 C0 C4 CD*WSHIFT:	

```

420 1A4 80 00 00 77 C0 C4 DE*
421 1A5 00 00 80 D7 C0 C7 E0*
422 1A6 A2 30 00 8F C0 40 00*FDIVCON:
423 1A7 80 00 00 77 D0 D1 01*
424 1A8 04 01 80 77 C0 C5 00*
425 1A9 80 00 00 73 40 44 0B*
426 1AA 9C 70 08 1F C0 40 00*
427 1AB 00 00 80 77 C0 C7 50*
428 1AC 9B 40 00 8F C0 40 00*
429 1AD 80 00 00 77 C0 C4 E4*
430 1AE 00 01 80 77 C0 C7 40*
431 1AF 9B D0 00 8F C0 40 00*
432 1B0 80 00 00 77 C0 C4 D4*
433 1B1 00 01 80 77 C0 C7 40*
434 1B2 9B 00 00 8F C0 40 00*
* . .
* . .
435 1B3 9F E0 00 9F C0 C4 C4*
* .
436 1B4 00 01 60 77 C0 C7 40*DIV:
* .
437 1B5 80 00 00 73 50 C9 69*DSUB:
438 1B6 80 00 00 73 60 C9 7A*
439 1B7 80 00 00 73 60 C9 8B*
* .
440 1B8 80 01 00 72 81 44 11*
441 1B9 80 00 84 74 90 D9 15*
* .
442 1BA 80 00 00 70 A0 C1 11*FROLB:
443 1BB 80 00 00 73 60 C1 44*
444 1BC 80 00 01 57 C0 C4 14*
* .
445 1BD 80 00 00 73 40 C1 66*REENT:
446 1BE 80 00 00 73 60 C1 77*
447 1BF 80 00 00 73 60 C1 88*
* .
448 1C0 80 01 00 72 81 44 11*
449 1C1 9B 50 01 1F C0 C4 51*
* .
450 1C2 80 00 00 73 40 C1 69*
451 1C3 80 00 00 73 60 C1 7A*
452 1C4 80 00 00 73 60 C1 88*
453 1C5 80 01 00 72 91 44 11*
454 1C6 9B A0 00 9F C0 E1 15*
* .
455 1C7 80 00 80 73 40 E5 00*DIVBYZ:
456 1C8 00 09 80 77 C0 C5 33*
457 1C9 A1 C0 84 9C 80 40 00*
* .
458 1CA A2 30 00 8F C0 40 00*FSUBCON:
459 1CB 80 00 80 77 C0 F5 00*
460 1CC 9C E0 00 9F C0 40 00*
* .
461 1CD A2 30 00 8F C0 40 00*FADDCON:
* .
462 1CE 80 00 00 77 C0 C4 C9*FADD1:
463 1CF 80 00 00 77 C0 C4 DA*
* .

```

464	1D0	07	FF	80	77	C0	E5	40
465	1D1	07	FF	80	77	C0	E5	51
466	1D2	80	00	00	73	90	C9	45
467	1D3	9E	C0	08	1F	C0	C4	E8
468	1D4	90	E0	02	9F	C0	04	01
469	1D5	80	00	00	77	C0	C4	10
470	1D6	80	00	00	77	C0	C2	00
471	1D7	80	00	00	77	C0	C4	C6
472	1D8	80	00	00	77	C0	C4	69
473	1D9	80	00	00	77	C0	C4	D7
474	1DA	80	00	00	77	C0	C4	7A
475	1DB	80	00	00	77	C0	C4	E8
476	1DC	80	00	00	77	C0	C4	88
477	1D0	80	00	00	77	D0	04	44
478	1DE	00	30	80	73	90	40	04
479	1DF	9F	E0	01	1F	C0	40	00
480	1E0	0F	F0	80	73	40	65	04
481	1E1	9E	60	08	1F	C0	43	00
482	1E2	00	10	80	77	D0	C0	44
483	1E3	80	00	00	77	C0	C4	67
484	1E4	80	00	00	77	C0	C4	78
485	1E5	9E	00	00	9F	D0	01	84
486	1E6	80	00	00	73	40	44	04
487	1E7	9E	C0	08	1F	C0	40	00
488	1E8	80	04	00	73	01	44	88
489	1E9	80	05	00	72	81	44	77
490	1EA	80	05	00	72	81	44	66
491	1EB	9E	70	00	9H	40	CC	44
492	1EC	80	00	00	73	40	71	10
493	1ED	9F	60	02	1F	C0	40	00
494	1EE	80	00	00	73	40	C1	C6
495	1EF	80	00	00	73	60	C1	D7
496	1F0	80	00	00	73	60	C1	E8
497	1F1	9F	E0	01	9F	C0	40	00
498	1F2	80	05	00	72	81	44	EE
499	1F3	80	05	00	72	81	44	00
500	1F4	80	05	00	72	81	44	CC
501	1F5	9F	E0	00	9F	D0	C4	00
502	1F6	80	00	00	73	50	C9	C6
503	1F7	80	00	00	73	60	C9	D7
504	1F8	80	00	00	73	60	C9	E8
505	1F9	9F	E0	01	1F	C0	40	00
506	1FA	80	00	80	77	C0	F5	00

```

507 1FB 80 00 00 73 50 D4 CC*
508 1FC 80 00 00 73 60 D4 DD*
509 1FD 80 00 00 73 60 D4 EE*
      *
      *
510 1FE 00 09 80 77 C0 C5 33*NORM1:
511 1FF 00 00 80 77 C0 C7 10*
512 200 80 00 00 73 40 44 0E*NORM1:
513 201 A0 80 08 9F C0 40 00*
514 202 00 10 80 77 C0 C5 11*
      *
515 203 00 30 80 70 90 4D 01*
516 204 A0 D0 08 1F C0 40 00*
517 205 80 00 00 77 C0 C4 ED*
518 206 80 00 00 77 C0 C4 DC*
519 207 A0 00 00 9F D0 D1 CC*
      *
520 208 A1 00 02 1F C0 40 00*NORM2:
521 209 80 00 00 77 D0 C4 11*
522 20A 80 00 00 73 40 C1 CC*
523 20B 80 00 00 73 60 C1 DD*
524 20C A0 80 00 98 60 C1 EE*
      *
525 20D 00 00 80 73 40 C7 00*NORM3:
526 20E A3 30 00 8F C0 40 00NSTUFF:
527 20F 80 00 50 7F C0 40 00*
      *
528 210 80 00 00 77 D0 C9 01*NORM4:
529 211 78 00 80 73 40 65 00*
530 212 A1 50 08 9F C0 40 00*
531 213 80 00 00 73 40 44 00*
532 214 A0 E0 80 98 C0 40 00*
      *
533 215 40 00 80 73 40 65 00*NORM45:
534 216 A1 B0 08 1F C0 40 00*
535 217 00 00 80 73 40 C7 00*
536 218 80 00 84 74 40 C4 E0*
537 219 80 00 00 77 C0 C4 D0*
538 21A A2 00 00 9F C0 C4 C0*
      *
539 21B 80 00 80 73 40 E5 00*NORM5:
540 21C 07 FF 80 77 C0 DD 00*MAXANS:
541 21D FF FF 84 74 40 C7 E0*
542 21E 80 00 80 73 C0 C4 DE*
543 21F 80 00 00 77 C0 C4 CE*
544 220 A3 30 00 8F C0 40 00FFFAULT:
545 221 FE F6 80 77 C0 C7 50*
546 222 8F 60 00 9F C0 40 00*
      *
      *
547 223 00 02 C0 77 CC 45 03*FLOAD:
548 224 00 00 00 77 CA C7 00*
549 225 00 04 C0 77 CC 45 03*
      *
      *
550 226 00 00 00 77 CA C7 80*
551 227 00 06 C0 77 CC 45 03*
552 228 00 00 00 77 CA C7 A0*

```

```

553 229 00 08 C0 77 CC 45 03*
554 22A 00 00 00 77 CA C7 90*
555 22B 00 08 C0 77 CC 45 03*
556 22C 00 00 00 77 CA C7 10*
557 22D 00 00 C0 77 CC 45 03*
558 22E 00 00 00 77 CA C7 80*
559 22F 00 0F C0 77 CC 45 03*
560 230 00 00 00 77 CA C7 70*
561 231 00 11 C0 77 CC 45 03*
562 232 00 00 00 D7 CA C7 60*
*
563 233 00 02 C0 77 CE 45 03*FSTORE:
564 234 80 00 00 77 CA 44 00*
565 235 00 04 C0 77 CE 45 03*
566 236 80 00 00 77 CA 44 0E*
567 237 00 06 C0 77 CE 45 03*
568 238 80 00 00 77 CA 44 0D*
569 239 00 08 C0 77 CE 45 03*
570 23A 80 00 00 D7 CA 44 0C*
*
571 23B 00 02 C0 77 CC 45 03*FNORMCON:
572 23C 00 00 00 77 CA C7 00*
573 23D 00 04 C0 77 CC 45 03*
574 23E 00 00 00 77 CA C7 E0*
575 23F 00 06 C0 77 CC 45 03*
576 240 00 00 00 77 CA C7 D0*
577 241 00 08 C0 77 CC 45 03*
578 242 00 00 00 77 CA C7 C0*
579 243 A0 U0 00 9F D0 D1 11*
*
*
*
*
*
580 244 00 70 80 77 C0 E5 40*STRUCONT:
581 245 00 0C 80 A7 J0 47 00*
582 246 80 00 20 6B 01 44 44*
583 247 A7 E0 00 8F C0 40 00*
584 248 00 04 C0 77 DC 4D 02*
585 249 00 00 00 77 CA C5 44*
586 24A 00 04 C0 77 CE 45 02*
587 24B 80 00 00 77 CA 44 04*
588 24C 00 0F 80 77 C0 E5 40*
589 24D A7 E0 00 8F C0 40 00*
590 24E 00 02 C0 77 DC 4D 02*
591 24F 00 00 00 77 CA C5 44*
592 250 00 06 C0 77 CE 45 02*
593 251 80 00 00 77 CA 44 04*
594 252 80 00 40 77 CC 44 02*VECCONT:
595 253 28 70 00 8F CA C7 40*
596 254 80 00 00 77 C0 C4 54*
597 255 00 02 C0 77 CC 45 02*
598 256 00 00 00 77 CA C7 60*
599 257 00 04 C0 77 CC 45 02*
600 258 28 70 00 8F CA C7 40*
601 259 00 06 C0 77 CC 45 02*
602 25A 00 00 00 77 CA C7 80*
603 25B 00 02 80 77 C0 C7 80*
604 25C 80 00 00 73 50 C9 45*
605 25D A6 00 02 9F C0 C4 74*

```

```

606 25E 00 02 80 77 D0 CD BB*
607 25F 80 00 00 77 D0 D4 77*
608 260 80 00 00 73 50 C9 86*VEC2:
609 261 A6 40 02 9F C0 40 00*
610 262 00 04 80 77 C0 C5 BB*
611 263 80 00 00 77 D0 D4 88*
612 264 00 12 80 77 C0 C5 66*VEC3:
613 265 80 00 00 73 50 C1 78*
614 266 A6 90 02 9B 40 CC 48*
615 267 80 00 00 77 D0 C4 BB*
616 268 80 00 00 73 40 CC 47*
617 269 80 00 41 77 CE 4C 03*YMAJORS
618 26A A7 C0 0C 1F CA 44 04*
619 26B 00 02 80 77 DE 4D 03*
620 26C 80 00 00 77 C8 44 0B*
*
621 26D 80 00 00 73 40 19 87*VECNORM:
622 26E A7 C0 08 1F C0 40 00*
623 26F 08 00 80 73 40 66 00*
624 270 A7 30 08 9F C0 40 00*
625 271 80 00 00 77 C0 C1 77*
626 272 A6 00 00 9F C0 C1 88*
*
627 273 00 04 C0 77 DE 4D 03*VSTOP:
628 274 80 00 00 77 CA 44 07*
629 275 00 06 C0 77 DE CD 33*
630 276 A7 80 00 8F CA 8B 38*
631 277 80 00 50 7A 40 44 00*
*
632 278 80 00 40 77 CE 4C 03*VPSHI:
633 279 80 00 00 77 CA 44 05*
634 27A 00 03 C0 77 DE CD 33*
635 27B 80 00 00 D7 CA 8B 36*
*
636 27C A7 80 84 8C 40 40 00*VPOINT:
637 27D 80 00 50 7F C0 40 00*
*
638 27E 80 00 00 77 C0 C4 C4*SCALE:
639 27F 80 00 00 77 C0 C1 44*
640 280 80 00 00 77 C0 C1 44*
641 281 00 01 80 70 80 65 01*
642 282 A8 40 08 9F C0 C1 44*
643 283 80 00 00 77 C0 C1 4C*
*
644 284 00 02 80 70 80 65 01*SCALE1:
645 285 80 00 08 57 C0 C1 CC*
646 286 80 04 00 D3 01 41 4C*
*
647 287 80 00 00 77 C0 04 04*XFORM:
648 288 00 0A 80 A7 C0 47 00*
649 289 80 06 20 6B 81 44 44*
650 28A 80 00 00 D7 D0 C8 44*
*
651 28B 80 00 40 77 CC 44 04*PSHRCONT:
652 28C 00 00 00 77 CA C7 50*
653 28D 80 00 40 77 DC C4 33*
654 28E 00 00 00 77 CA C7 60*
655 28F 80 00 40 77 CE 44 05*
656 290 80 00 00 77 CA 44 06*

```

657 291 80 00 40 77 DE 83 34•  
658 292 00 02 80 77 CA 43 05•  
659 293 83 00 00 9F C0 40 00•  
•  
660 294 80 00 40 77 CC 44 04•RTRNCONT:  
661 295 00 00 00 77 CA C7 50•  
662 296 00 02 C0 77 DC CD 55•  
•  
663 297 00 00 00 77 CA C7 60•  
664 298 80 00 40 77 CE 44 04•  
665 299 80 00 00 77 CA 44 05•  
666 29A B1 60 00 9F C6 44 06•  
•  
667 29B 80 00 50 7F CA 88 32•PSHXCONT:  
•  
668 29C 00 00 00 77 CA C7 20•PULXCONT:  
669 29D 80 00 50 7F D0 C4 33•  
•  
670 29E 28 B0 09 9F C4 C7 50•BHICONT:  
671 29F 80 00 50 7F C0 40 00•  
•  
672 2A0 28 B0 09 1F C4 C7 50•BLSCONT:  
673 2A1 80 00 50 7F C0 40 00•  
•  
674 2A2 28 B0 01 9F C4 C7 50•BCCCONT:  
675 2A3 80 00 50 7F C0 40 00•  
•  
676 2A4 28 B0 01 1F C4 C7 50•BCSCONT:  
677 2A5 80 00 50 7F C0 40 00•  
•  
678 2A6 28 B0 08 9F C4 C7 50•ENECONT:  
679 2A7 80 00 50 7F C0 40 00•  
•  
680 2A8 28 B0 08 1F C4 C7 50•EQCONT:  
681 2A9 80 00 50 7F C0 40 00•  
•  
682 2AA 28 B0 03 9F C4 C7 50•BYCCONT:  
683 2AB 80 00 50 7F C0 40 00•  
•  
684 2AC 28 B0 03 1F C4 C7 50•BYSCONT:  
685 2AD 80 00 50 7F C0 40 00•  
•  
686 2AE 28 B0 02 9F C4 C7 50•BPLCONT:  
687 2AF 80 00 50 7F C0 40 00•  
•  
688 2B0 28 B0 02 1F C4 C7 50•BMICONT:  
689 2B1 80 00 50 7F C0 40 00•  
•  
690 2B2 28 B0 04 9F C4 C7 50•BGECONT:  
691 2B3 80 00 50 7F C0 40 00•  
•  
692 2B4 28 B0 04 1F C4 C7 50•BLTCONT:  
693 2B5 80 00 50 7F C0 40 00•  
•  
694 2B6 28 B0 0C 9F C4 C7 50•BGTCONT:  
695 2B7 80 00 50 7F C0 40 00•  
•

```

696 2B8 2B B0 0C 1F C4 C7 50 *BLECONT:
697 2B9 00 00 50 7F C0 40 00*
698 2BA 00 00 00 77 C4 C7 50 *BRACONT:
699 2BB 00 00 50 7F C6 41 54 *DOBRANCH:
*
*
*
700 2BC 80 00 40 77 DC 44 03 *MVGET:
701 2BD 00 00 00 77 CA C7 60*
702 2BE 00 03 C0 77 CC 45 03*
703 2BF 00 00 00 77 CA C7 50*
704 2C0 00 05 C0 77 CC 45 03*
705 2C1 00 00 00 D3 4A C7 70*
*
706 2C2 AB C0 00 8F C0 40 00 *MVLRCONT:
707 2C3 AC E0 08 1B 40 4C 07 *MVLRI:
708 2C4 AC A0 08 1F C0 40 00*
709 2C5 00 02 C0 77 CC 85 66*
710 2C6 00 00 00 77 CA C7 40*
711 2C7 00 02 C0 77 CE 85 55*
712 2C8 00 00 00 77 CA 8B 74*
713 2C9 AC 30 00 98 40 CC 77*
*
714 2CA 80 00 00 77 CC 44 06 *MVODD:
715 2CB 00 00 00 77 C8 C7 40*
716 2CC 80 00 00 77 CE 44 05*
717 2CD 80 00 00 77 C8 44 04*
718 2CE 00 06 00 7F C0 C5 33 *MVDONE:
*
719 2CF AB C0 00 8F C0 40 00 *MVRLLCONT:
720 2D0 AC E0 08 1B 40 4C 07 *MVRLLI:
721 2D1 AC A0 08 1F C0 40 00*
722 2D2 80 00 40 77 CC CC 66*
723 2D3 00 00 00 77 CA C7 40*
724 2D4 80 00 00 77 C0 CC 66*
725 2D5 80 00 40 77 CE CC 55*
726 2D6 80 00 00 77 CA 8B 54*
727 2D7 00 02 80 73 50 CD 77*
728 2D8 AD 00 00 9F C0 40 00*
*
729 2D9 80 00 00 77 C0 C1 44 *PATCHCONS:
730 2DA 80 00 00 77 C0 C1 44*
731 2DB 44 00 00 7F C6 45 04*
*
732 2DC 80 00 00 77 C0 C1 55 *CPATCHCNS:
733 2DD 22 00 80 77 C0 C5 55*
734 2DE 92 20 00 9F C0 C1 55*
*
*
735 2DF 80 00 40 77 CC 44 04 *MADACONS:
736 2E0 00 00 50 7B 4A C5 22*
*
*
*+      Kern
*+      syst      If 'oops' follow. Executable through
*+                  texture.
*

```

737 2F6 2F 70 00 9F C4 C7 40\*LDUP16:  
    \*;+ Transfer the PC, which was loaded with the  
    \*;+ desired address to test, to a work register in R9R1  
    \*  
738 2F7 80 00 40 77 CE 44 04\*LP16CONT:  
    \*;+ Transfer work register to memory write address.  
739 2F8 80 00 00 77 CA 44 05\*  
    \*;+ Write R5 (previously loaded with desired data) to memory  
740 2F9 80 00 40 77 CC 44 04\*  
    \*;+ Transfer work register to memory read address.  
741 2FA 2F 70 00 9F CA C7 60\*  
    \*;+ Read memory data to R6 and go to LP16CONT.  
    \*  
    \*  
    \*;+ The following is an 8-bit version of the same test.  
    \*  
742 2FB 2F C0 00 9F C4 C7 40\*LCUP8:  
    \*;+ Transfer PC to work register.  
    \*  
743 2FC 80 00 00 77 CE 44 04\*LP8CONT:  
    \*;+ Transfer work register to memory write address.  
744 2FD 80 00 00 77 CA 44 05\*  
    \*;+ Write R5 to memory.  
745 2FE 80 00 00 77 CC 44 04\*  
    \*;+ Transfer work register to read address.  
746 2FF 2F C0 00 9F C8 C7 60\*  
    \*;+ Read memory into R6 and goto LP8CONT.  
    \*  
    \*  
    \*  
747 300 80 00 00 9F C0 40 00\*TEST:  
748 301 80 00 50 7F C0 40 00\*NOP:  
749 302 80 00 50 7F C0 40 00\*NOP2:  
750 303 80 C0 50 7E 4C 10 00\*SFA:  
751 304 8C E0 00 9F C0 40 00\*DUMP:  
752 305 8C E0 00 9F C0 40 00\*  
753 306 12 70 81 9F C0 C7 40\*TAP:  
754 307 12 60 81 9F C0 C7 00\*TPA:  
755 308 80 00 50 79 00 C4 22\*INX:  
756 309 80 00 50 79 C0 CC 22\*DEX:  
757 30A 80 00 D0 7C 40 40 00\*CLV:  
758 30B 80 00 D4 7C 40 40 00\*SEV:  
759 30C 80 00 D0 7C 80 40 00\*CLC:  
760 30D 80 00 D4 7C 80 40 00\*SEC:  
761 30E 80 00 D0 7C C0 40 00\*CLI:  
762 30F 80 00 D4 7C C0 40 00\*SEI:  
763 310 80 00 50 79 50 C9 01\*SBA:  
764 311 80 00 50 79 50 51 10\*CBA:  
765 312 80 00 50 7E 80 44 00\*TAPX:  
766 313 00 00 D1 7F C0 C7 00\*TPAX:  
767 314 80 00 50 7F C0 C5 22\*ADXI:  
768 315 80 00 50 7F C0 C5 33\*ASPI:  
769 316 80 00 50 7A 40 C4 10\*TAB:  
770 317 80 00 50 7A 40 C4 01\*TBA:  
771 318 80 00 50 7F 00 40 00\*SDA:  
772 319 8C E0 00 9F C0 40 00\*DAA:  
773 31A 94 80 00 9F C0 40 00\*NLDXX:  
774 31B 80 00 50 78 40 C1 01\*ABA:

775 31C 94 80 00 9F C0 40 00 \*NLDAX:  
 776 31D 94 E0 00 9F C0 40 00 \*NLDBX:  
 777 31E 95 10 00 9F C0 40 00 \*NSTAX:  
 778 31F 95 40 00 9F C0 40 00 \*JMPAX:  
 779 320 AB A0 00 9F C0 C7 40 \*BRA:  
 780 321 80 00 50 7E 00 40 00 \*SDB:  
 781 322 A9 E0 00 9F C0 C7 40 \*BMI:  
 782 323 AA 00 00 9F C0 C7 40 \*BLS:  
 783 324 AA 20 00 9F C0 C7 40 \*BCC:  
 784 325 AA 40 00 9F C0 C7 40 \*BCS:  
 785 326 AA 60 00 9F C0 C7 40 \*ANE:  
 786 327 AA 80 00 9F C0 C7 40 \*REU:  
 787 328 AA A0 00 9F C0 C7 40 \*RVC:  
 788 329 AA C0 00 9F C0 C7 40 \*EVG:  
 789 32A AA F0 00 9F C0 C7 40 \*BPL:  
 790 32B AB 00 00 9F C0 C7 40 \*BMI:  
 791 32C AB 20 00 9F C0 C7 40 \*EGE:  
 792 32D AB 40 00 9F C0 C7 40 \*BLT:  
 793 32E AB 60 00 9F C0 C7 40 \*BGT:  
 794 32F AB 80 00 9F C0 C7 40 \*BLE:  
 795 330 80 00 50 7F D0 C4 23 \*TSX:  
 796 331 80 00 50 7F D0 C4 33 \*INS:  
 797 332 8C F0 00 9F DC C4 33 \*PULA:  
 798 333 80 00 00 9F DC C4 33 \*PULB:  
 799 334 80 00 50 7F C0 CC 33 \*DES:  
 800 335 80 00 50 7F C0 CC 32 \*TXS:  
 801 336 80 10 00 9F CE AC 33 \*PSHA:  
 802 337 80 20 00 9F CE BC 33 \*PSHB:  
 803 338 90 00 50 7F C6 47 00 \*JMPIN:  
 804 339 80 30 40 4F DC C4 33 \*HTS:  
 805 33A 95 60 00 9F C0 C7 40 \*FPSHD:  
 806 33B 90 F0 00 9F DC C4 33 \*RTI:  
 807 33C 95 60 00 9F C0 C5 42 \*FPSHX:  
 808 33D 95 60 00 9F C0 C7 40 \*FPMH:  
 809 33E 8C E0 00 9F C0 40 00 \*wAI:  
 810 33F 10 00 A1 9F C0 C7 40 \*SwI:  
 811 340 80 00 50 78 90 D4 00 \*NEGA:  
 812 341 16 C0 81 9F C0 C7 90 \*FPHI:  
 813 342 97 30 00 9F C0 C7 40 \*FPULD:  
 814 343 80 00 50 79 00 FC 00 \*CUMA:  
 815 344 8E F0 00 9H 01 44 00 \*LSRA:  
 816 345 97 30 00 9F C0 C5 42 \*FPULX:  
 817 346 8E F1 00 9A 81 44 00 \*RUHA:  
 818 347 8E E0 00 98 80 44 00 \*ASRA:  
 819 348 80 00 50 78 80 C1 00 \*ASLA:  
 820 349 80 00 50 78 80 C1 00 \*ROLA:  
 821 34A 80 00 50 78 C0 CC 00 \*DECA:  
 822 34B 97 30 00 9F C0 C7 40 \*FPUL:  
 823 34C 80 00 50 78 D0 C4 00 \*INCA:  
 824 34D 91 00 50 78 80 44 00 \*ISTA:  
 825 34E 97 B0 00 9F C0 40 00 \*FDUP:  
 826 34F 00 00 D0 78 80 C7 00 \*CLRA:  
 827 350 80 00 50 78 90 D4 11 \*NEGB:  
 828 351 98 20 00 9F C0 40 00 \*FSWP:  
 829 352 9C D0 00 9F C0 40 00 \*FADD:  
 830 353 80 00 50 79 00 FC 11 \*CUMB:  
 831 354 8F 10 00 9B 01 44 11 \*LSRB:  
 832 355 9C A0 00 9F C0 40 00 \*FSUB:  
 833 356 8F 11 00 9A 81 44 11 \*RORB:  
 834 357 8F 00 00 98 80 44 01 \*ASRB:

835	358	80 00 50 78 80 C1 11	ASLB1
836	359	80 00 50 78 A0 C1 11	ROLB1
837	35A	80 00 50 78 C0 CC 11	DEC81
838	35B	98 E0 00 9F C0 40 00	FMULT1
839	35C	80 00 50 78 00 C4 11	INC81
840	35D	80 00 50 78 80 44 01	TSTB1
841	35E	9A 60 00 9F C0 40 00	FDIV1
842	35F	00 00 00 78 80 C7 10	CLR81
843	360	8D 50 00 9F C0 C3 42	NEG1
844	361	A3 80 00 9F C0 40 00	FNORM1
845	362	A8 80 00 9F C0 C7 40	PSHRET1
846	363	8D 90 00 9F C0 C3 42	COM1
847	364	8D F0 00 9F C0 C3 42	LSR1
848	365	A9 40 00 9F C0 C7 40	RTRM1
849	366	8E 10 00 9F C0 C3 42	ROR1
850	367	8E 30 00 9F C0 C3 42	ASR1
851	368	8E 60 00 9F C0 C3 42	ASLB1
852	369	8E 90 00 9F C0 C3 42	FOL1
853	36A	8D 80 00 9F C0 C3 42	DEC1
854	36B	A9 80 40 9F C2 CC 33	PSHM1
855	36C	8D 00 00 9F C0 C3 42	INC1
856	36D	8E C0 00 9F C2 C3 43 02	TST1
857	36E	80 00 50 7F C6 43 02	JMP1
858	36F	8E D0 00 9F C1 C3 43 02	CLR1
859	370	8D 50 00 9F C0 C7 40	NEG1
860	371	A4 40 00 9F C0 40 00	STRUK1
861	372	A5 20 00 9F C0 40 00	VEC1
862	373	8D 90 00 9F C0 C7 40	COM1
863	374	8D F0 00 9F C0 C7 40	LSR1
864	375	A9 C0 40 9F DC C4 33	PUL1
865	376	8E 10 00 9F C0 C7 40	RCR1
866	377	8E 30 00 9F C0 C7 40	ASR1
867	378	8E 60 00 9F C0 C7 40	ASLB1
868	379	8E 90 00 9F C0 C7 40	FOL1
869	37A	8D 80 00 9F C0 C7 40	DEC1
870	37B	8C E0 00 9F C0 40 00	
871	37C	8D D0 00 9F C0 C7 40	INC1
872	37D	80 00 50 78 80 47 00	TST1
873	37E	80 00 50 7F C6 47 00	JMP1
874	37F	80 00 50 79 50 51 44	CLR1
875	380	80 00 50 79 50 CD 00	SUBA1
876	381	80 00 50 79 50 4D 00	CMPA1
877	382	80 00 50 79 70 CD 00	SBCA1
878	383	8C E0 00 9F C0 40 00	
879	384	8D 00 50 7A 40 E3 00	ADCA1
880	385	80 00 50 7A 40 63 00	BITA1
881	386	80 00 50 7A 40 C7 00	LDAA1
882	387	8C E0 00 9F C0 40 00	
883	388	80 00 50 7A 40 F3 00	ECPA1
884	389	80 00 50 78 60 C3 00	ADCA1
885	38A	80 00 50 7A 40 DD 00	CBAA1
886	38B	80 00 50 78 40 C3 00	ACDA1
887	38C	93 60 00 9B 50 4D 02	CPII1
888	38D	92 10 00 9F C0 C7 40	BSR1
889	38E	80 00 50 7A 00 C7 30	LDS1
890	38F	8C E0 00 9F C0 40 00	
891	390	80 00 50 79 50 CD 00	SUBAD1
892	391	80 00 50 79 50 4D 00	CNPAD1
893	392	80 00 50 79 70 CD 00	SBCAD1
894	393	8C E0 00 9F C0 40 00	

995	394	80 00 50 7A 40 E5 00	ANDAD:
996	395	80 00 50 7A 40 65 00	BITAD:
997	396	80 00 50 7A 40 C7 00	LDAAD:
998	397	80 00 50 7A 40 44 00	STAAD:
999	398	80 00 50 7A 40 F5 00	EORAD:
900	399	80 00 50 7A 60 C3 00	ADCAD:
901	39A	80 00 50 7A 40 DD 00	DPAAD:
902	39B	80 00 50 7B 40 C3 00	ADDAD:
903	39C	93 60 00 9B 50 4D 02	CPXD:
904	39D	8C E0 00 9F C0 40 00	00
905	39E	80 00 50 7A 00 C7 30	LDSJ:
906	39F	80 00 50 7A 00 44 03	STSD:
907	3A0	92 B9 00 9F CC 45 02	SUBAK:
908	3A1	92 C0 00 9F CC 45 02	CMPAK:
909	3A2	92 D0 00 9F CC 45 02	SBCAK:
910	3A3	8C E9 00 9F C0 40 00	00
911	3A4	92 E0 00 9F CC 45 02	INDAK:
912	3A5	92 F0 00 9F C0 45 02	BITAK:
913	3A6	93 00 00 9F CC 45 02	LDAAK:
914	3A7	94 70 00 9F CC 45 02	STAAK:
915	3A8	93 10 00 9F CC 45 02	EOPAK:
916	3A9	93 20 00 9F CC 45 02	ADCAK:
917	3AA	93 30 00 9F CC 45 02	DPAAK:
918	3AB	93 40 00 9F CC 45 02	ADDAK:
919	3AC	93 50 40 9F CC 45 02	CPXK:
920	3AD	92 20 00 9F C0 C9 52	JSPK:
921	3AE	93 30 40 9F CC 45 02	LDSK:
922	3AF	94 40 40 9F C0 45 02	STSK:
923	3B0	80 00 50 79 90 C0 00	00
924	3B1	80 00 50 79 90 40 00	CMPK:
925	3B2	80 00 50 79 70 C0 00	SBCAK:
926	3B3	8C E0 00 9F C0 40 00	00
927	3B4	80 00 50 7A 40 C5 00	ANDAK:
928	3B5	80 00 50 7A 40 65 00	BITAK:
929	3B6	80 00 50 7A 40 C7 00	LDAAK:
930	3B7	80 00 50 7A 40 44 00	STAAK:
931	3B8	80 00 50 7A 40 F5 00	EORAK:
932	3B9	80 00 50 7B 40 C5 00	ADCAK:
933	3BA	80 00 50 7A 40 DD 00	DRAAK:
934	3BB	80 00 50 7B 80 C5 00	ADDAS:
935	3BC	93 60 00 9B 50 4D 02	CPK:
936	3BD	92 20 00 9F C0 C7 50	JERK:
937	3BE	80 00 50 7A 00 C7 30	LUSK:
938	3BF	80 00 50 7A 00 44 03	STS:
939	3C0	80 00 50 79 90 C0 11	SUBBK:
940	3C1	80 00 50 79 90 40 01	CMPBK:
941	3C2	80 00 50 79 70 C0 11	SBCBK:
942	3C3	8C E0 00 9F C0 40 00	00
943	3C4	80 00 50 7A 40 C5 11	ANDBK:
944	3C5	80 00 50 7A 40 65 01	BITBK:
945	3C6	80 00 50 7A 40 C7 11	LDBBK:
946	3C7	8C E0 00 9F C0 40 00	00
947	3C8	80 00 50 7A 40 F5 11	EORBK:
948	3C9	80 00 50 7B 60 C5 11	ADCBK:
949	3CA	80 00 50 7A 40 DD 11	DRABK:
950	3CB	80 00 50 7B 40 C5 11	ADDBK:
951	3CC	00 FF 00 77 C0 E3 00	ADAK:
952	3CD	80 00 50 7A 40 C1 20	HADAK:
953	3CE	80 00 50 7A 00 C7 20	LOXK:
954	3CF	8C E0 00 9F C0 40 00	00

955 3D0 80 00 50 79 50 CD 11\*SUBBD:  
956 3D1 80 00 50 79 50 4D 01\*CMPBD:  
957 3D2 80 00 50 79 70 CD 11\*SBCBD:  
958 3D3 8C E0 00 9F C0 40 00\*  
959 3D4 80 00 50 7A 40 E5 11\*ANDBD:  
960 3D5 80 00 50 7A 40 65 01\*BITBD:  
961 3D6 80 00 50 7A 40 C7 10\*LDABD:  
962 3D7 80 00 50 7A 40 44 01\*STABD:  
963 3D8 80 00 50 7A 40 F5 11\*EORBD:  
964 3D9 80 00 50 78 60 C5 11\*ADCBD:  
965 3DA 80 00 50 7A 40 DD 11\*ORABD:  
966 3DB 80 00 50 78 40 C5 11\*ADDAD:  
967 3DC FF FF D0 7F C0 C7 F0\*SBUG:  
968 3DD 00 00 D0 7F C0 C7 F0\*CBUG:  
969 3DE A0 00 50 7A 00 C7 20\*LDXD:  
970 3DF 80 00 50 7A 00 44 02\*STXD:  
971 3E0 93 90 00 9F CC 45 02\*SUBBX:  
972 3E1 93 A0 00 9F CC 45 02\*CMPBX:  
973 3E2 93 B0 00 9F CC 45 02\*SBCBX:  
974 3E3 AC 20 00 9F C0 40 00\*MVRL:  
975 3E4 93 C0 00 9F CC 45 02\*ANDBX:  
976 3E5 93 D0 00 9F CC 45 02\*BITBX:  
977 3E6 93 E0 00 9F CC 45 02\*LDABX:  
978 3E7 94 60 00 9F CE 45 02\*STABX:  
979 3E8 93 F0 00 9F CC 45 02\*EURBX:  
980 3E9 94 00 00 9F CC 45 02\*ADC BX:  
981 3EA 94 10 00 9F CC 45 02\*ORABX:  
982 3EB 94 20 00 9F CC 45 02\*ADD BX:  
983 3EC AC F0 00 9F C0 40 00\*MVRL:  
984 3ED AD F0 00 9F C0 C7 40\*WADX:  
985 3EE 94 30 40 9F CC 45 02\*LDXX:  
986 3EF 94 50 40 9F CE 45 02\*STXX:  
987 3F0 80 00 50 79 50 CD 11\*SUBB:  
988 3F1 80 00 50 79 50 4D 01\*CMPB:  
989 3F2 R0 00 50 79 70 CD 11\*SBCB:  
990 3F3 AD C0 00 9F : C7 50\*CPATCH:  
991 3F4 80 00 50 7A 40 E5 11\*ANDB:  
992 3F5 80 00 50 7A 40 65 01\*BITB:  
993 3F6 80 00 50 7A 40 C7 10\*LDAB:  
994 3F7 80 00 50 7A 40 44 01\*STAB:  
995 3F8 80 00 50 7A 40 F5 11\*EORB:  
996 3F9 80 00 50 78 60 C5 11\*ADC B:  
997 3FA 80 00 50 7A 40 DD 11\*ORAB:  
998 3FB 80 00 50 78 40 C5 11\*ADD B:  
999 3FC 8C E0 00 9F C0 40 00\*  
1000 3FD AD 90 00 9F C0 C7 40\*PATCH:  
1001 3FE 80 00 50 7A 00 C7 20\*LDX:  
1002 3FF 80 00 50 7A 00 44 02\*STX:

## 067-0902-00/067-0942-99 MEMORY MAP

The ROM on the 067-0942-99 Personality Board defines some memory space in the 067-0902-99 fixture. Figure 1 shows the resulting memory map.

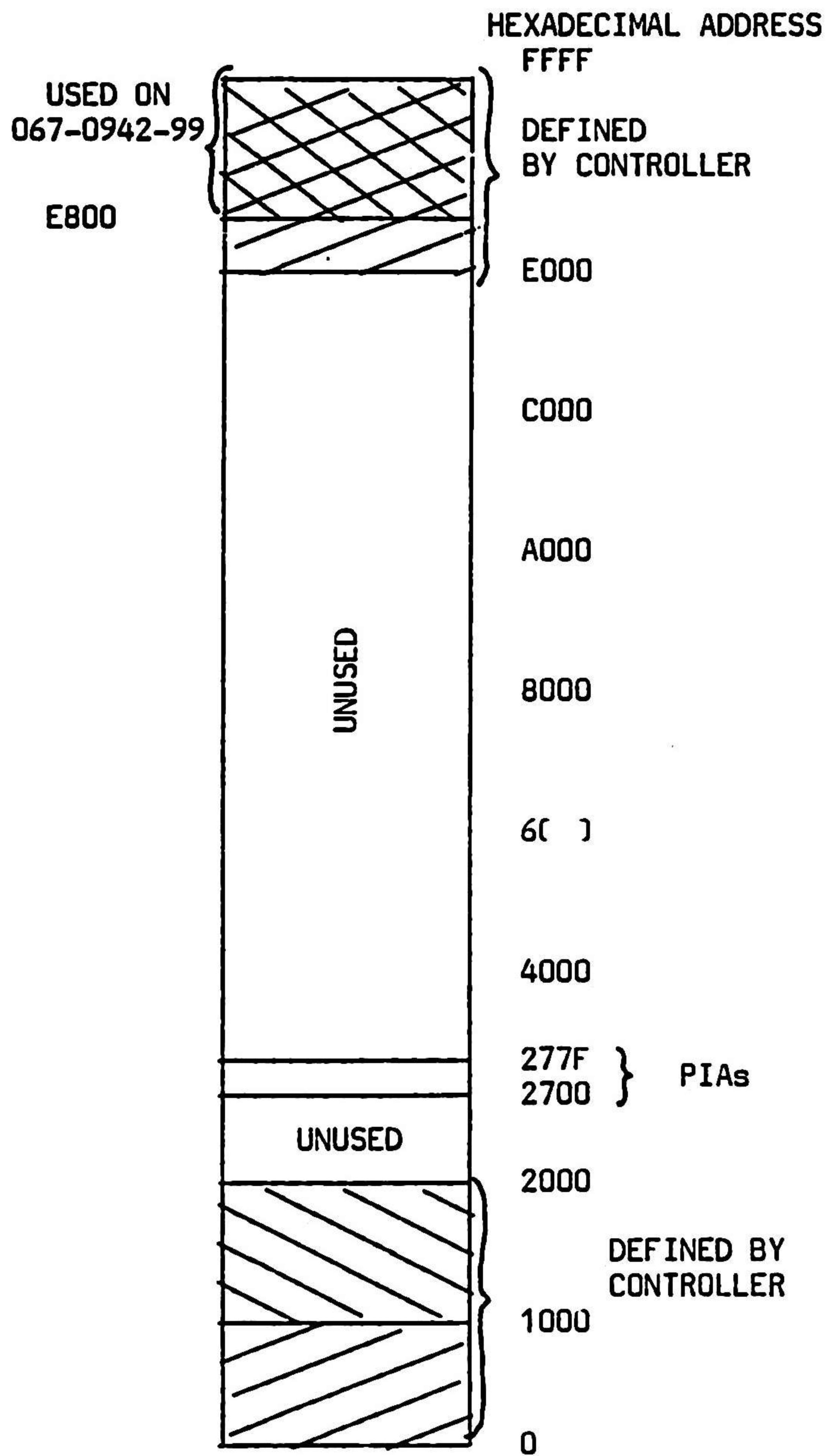


Figure 1. 067-0942-99  
Memory Map

**FIRMWARE FOR 067-0942-99**

If the 067-0942-99 fails to operate correctly, the following firmware listing may be useful in finding circuit faults.