CompuPro®

Technical Training Manual

FOR USE BY AUTHORIZED SYSTEMS CENTERS,
DEALERS AND OEMS ONLY.

ALL INSTALLATION AND SERVICE
MUST BE PERFORMED BY QUALIFIED PERSONNEL.

INTRODUCTION

This Training Manual is designed to aid technicians in the repair of COMPUPRO products. It is intended for use by technicians having completed the COMPUPRO TECHNICAL TRAINING. It was designed as an instructional aid and for use as an easy reference in the lab.

There are three sections to this manual. The material covers system packages, individual boards, and BIOS and operating system basics.

CONTENTS

SYSTEMS SECTION

CompuPro Systems Packages

Description of each system
Standard switch and jumper settings
Troubleshooting chart
Floppy Disk Drive Alignment procedures

BOARD PRODUCTS SECTION

Technical summary of each board
Diagnostic documentation
Troubleshooting chart
Diagnostic tests list

SOFTWARE SECTION

CP/M 2.2
CP/M 86
CP/M 8-16
MP/M 8-16
CP/M 68K

APPENDIX

Recommended equipment for System Center Labs
Logic Diagram - Jade Bus Probe

Also included in the manual is a floppy diskette containing diagnostic software useful in the repair of COMPUPRO products, as well as a list of recommended equipment for the SYSTEM CENTER labs. Technical manuals on the operating systems are contained in the software section.

TECHNICAL TRAINING MANUAL
Copyright ^1984
CompuPro
Hayward, CA 94545

Document #17200

First Printing: April 1983
Second Printing: July 1983
Third printing: July 1984

Contents of this document were proofread with the aid of SpellGuard~ a product from SORCIM Inc. Santa Clara, CA.

DISCLAIMER - CompuPro makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, CompuPro reserves the right to revise this publication and to make any changes from time to time in the content hereof without obligation of CompuPro to notify any person of such revision or changes.

Registered trademarks: CompuPro; CompuPro: CP/M; Digital Research. Trademarks: SYSTEM 8-16A, SYSTEM 8-16B, SYSTEM 8-16C, SYSTEM 8-16D, SYSTEM E/68K, RAM 16, RAM 17, RAM 21, RAM 22, SYSTEM SUPPORT 1, INTERFACER 3, INTERFACER 4, DISK 1, DISK 2, and SELECTOR CHANNEL, DISK 3, ENCLOSURE 2, M-DRIVE/H; W.J. Godbout: CP/M 80; CP/M-86; MP/M-86 and Digital Research; Digital Research.

All rights reserved. No part of this publication may be reproducted or transmitted in any form, or by any means, without the written permission of CompuPro. Printed and assembled in the United States of America.

SYSTEMS SECTION

COMPUPRO SYSTEMS PACKAGES

This section of the manual covers the COMPUPRO SYSTEMS. Contained is all material necessary to set up any system from start to finish. The make up of each system is discussed. Material is included on standard switch settings and jumpers for each system. Also included is a troubleshooting chart, covering common problems and possible remedies for the systems as a whole.

1. Description of the COMPUPRO SYSTEMS as a general product. A detailed description of each individual system is given, including the contents, features, possible applications, and advantages of each system.

2. Standard switch and jumper settings for the systems. Each board is covered individually. Required modifications for upgrading boards to run MP/M are included.

3. A troubleshcoting chart for the systems. This chart covers board level troubleshooting.

This section was designed for the purpose of getting systems up and running as quickly and easily as possible. Problems with individual boards are discussed in the next section of the manual.

2. STANDARD SWITCH SETTINGS AND CABLE CONNECTIONS

SYSTEM 8-16 A

SYSTEM 8-16 B

SYSTEM 8-16 C

SYSTEM 8-16 D

SYSTEM 8-16 E (68K)

3. TROUBLESHOOTING CHART

COMMON SYSTEM PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
Won´t boot at all	No or incorrect voltage	Check voltage on motherboard
	No clock signal on bus pin 24	Check for clock signal
	Bad or incorrectly hooked up cables	Check cables and replace as needed
	Bad DISK 1/1A	Swap DISK 1/1A
Lights on both drives lit	Cables to drives connected backwards	Turn over 50 pin cable on one end and reconnect
Runs CP/M 80 will not boot CP/M 86	Bad or missing ´Go 86´ EPROM or 6116 RAM chip	Check or replace EPROM or RAM chip
	Bad diskette	Try another diskette
	Bad DISK 1/1A	Swap DISK 1/1A
	RAM addressed incorrectly	Check switch settings
	Bad RAM board at extended address	Swap RAM boards
	Bad SYSTEM SUPPORT	Swap SYSTEM SUPPORT
	8085/88 CPU not swapping processors	Check CPU board
Head loads once then locks up	Bad memory Board	Check memory in page one
	Bad DISK 1/1A	Swap DISK 1/1A
Head loads over and over	Bad DISK 1/1A	Swap DISK 1/1A
	Bad cables	Check cables and replace as needed

SYMPTOM	PROBABLE CAUSE	REMEDY
Won´t format disk	Bad DISK 1/1A	Swap DISK 1/1A
	Bad drive	Swap drives
	Bad cable	Change cable
Nothing on the screen but sounds as if it´s booting	Bad serial I/O	Check console board
	Bad or incorrectly hooked up cable	Check cables
	Terminal set wrong	Check terminal setup
	DISK 1/1A set for wrong serial board	Check DISK 1/1A paddles
Garbage on screen	Bad serial I/O	Check console board
	Terminal not set correctly	Check switch settings on the terminal
	Bad terminal	Swap with a good one
Won´t boot MP/M	Interrupt jumpers on I/O board not correct	Check jumpers
	Bad CPU	Check X2 on CPU 8085/88
		Swap CPU
Some users won´t work under MP/M	Bad I/O board	Swap I/O board
	Interrupt jumpers wrong on I/O board	Check jumpers

SYMPTOM	PROBABLE CAUSE	REMEDY
System lock up when one user logs out	Bad I/O 3 board	Swap I/O 3 board
Panic! Trap Uninitialized Interrupt message	All terminals not turned on	Turn on all terminals or disconnect terminals not in use
	Interrupt jumpers wrong on I/O board	Check interrupt jumpers
System 816 D Won´t boot MP/M and gives message Uninitialized interrupt	Modification not done on SYSTEM SUPPORT board	Pull out pin 4 on U28
		Check for dip shunt in J8
When all else fails	Bad motherboard	Turn enclosure upside down and pound on it
		Change enclosure

COMPUPRO

FLOPPY DISK ALIGNMENT PROCEEDURE FOR QUME 842 DRIVES

7/26/84

INTRODUCTION

This document describes the equipment and proceedures required to align a QUME 842 floppy disk drive. This document is intended for the use of CompuPro System Centers, Dealers and OEMs. This is not intended for use by the end user.

REQUIREMENTS

To perform this procedure correctly, you will need the following items.

1) A working 816 letter series machine running CP/M 80.

2) A second DISK 1 or 1A addressed at A0 hex.

3) A DYMEK alignment diskette.

4) The CompuPro alignment routine called "ALIGN.COM" and its associated files.

5) An Oscilloscope with trigger input and two probes.

6) A QumeTrak 842 Maintenance manual is suggested.

7) A drive to align with cable and power supply.

PROCEDURE

1) Set the switches on the DISK 1 or 1A board so that it is addressed at A0 hex and the BOOT EPROM is inhibited. Install this in the 816 system and connect the floppy drive to be aligned to it with a 50 pin cable.

2) Connect the Oscilloscope to the drive as follows:

   A. Connect the channel input to test point "1A".
   B. Connect the external trigger input to test point "3"
   C. Connect the ground clips of both probes to ground.

3) Set the Oscilloscope as follows:

   A. Set your channel input to 200 mV./div., AC coupled, with a sweep rate of 20 mS./div.
   B. Set your trigger controls to external, DC coupled, normal triggering.

4) Power up the system and BOOT CP/M 80. Execute the program "ALIGN.COM". Insert the alignment diskette into the drive to be aligned.

5) After the routine signs on, type a <CR> and it should answer back "OK". Now type "SHOW" (All commands must be in capitol letters only!!!) and strike <CR>. The program will respond with the menu of commands. When executing a command, always follow it with a <CR>.

SHOW	- LIST COMMAND OPTIONS
INIT	- INITIALIZE DISK1 ALIGNING DRIVES
S00	- HOME TO TRACK 0
S38	- MOVE TO TRACK 38
S40	- MOVE TO TRACK 40
S76	- MOVE TO TRACK 76
TR	- PERFORM CONTINUOUS TRACK READ
S02	- SEEK BETWEEN TRACKS 0 & 2 CONTINUOUSLY (HIT ANY KEY TO TERMINATE)
DRA,B,C,D	- SELECT DRIVE FOR ALIGNMENT
BYE	- EXIT TEST ROUTINE BACK TO CP/M

Two additional commands not listed are:

H0	- SELECT HEAD 0 (THIS IS THE DEFAULT)
H1	- SELECT HEAD 1

6) Type "INIT" to initialize the controller. Then type "S00" to recalibrate the drive to track 0. Next type "S40" to seek to track 40. Type "TR" to load the heads on the drive.

RADIAL ALIGNMENT

7) At this point you should see the "cats eye" lobe pattern on the oscilloscope. See figure 43., page 49 of the QUME manual. The lobes should be of the same amplitude as shown. Now select head 1 by striking any key to unload the heads, type "H1" to select head 1, and type "TR" to load the heads again. You can go back to head 0 by typing the same procedure above by substituting "H0" for "H1".

If the lobes are not within 80% of each other, loosen the two screws on the PCB side of the main frame which secure the stepper motor. Rotate the stepper motor very slowly and carefully until the lobes are of equal amplitude.

Select the opposite head and verify that the lobes are approximately identical and equal on both heads.

Carefully tighten the stepper motor screws and re-verify the alignment.

READ/WRITE HEAD AZIMUTH CHECK

8) Type "S76" to seek to track 76. Type "H0" to select head 0, and type "TR" to load the heads.

Set the vertical deflection to 100 mV. and the time base to 0.5 mS./div. You should now see the azumith check pattern shown on figure 21. and page 27 of the QUME manual.

Compare the wave form displayed to the Azumith Check illistration in figure 21. If the displayed wave form is not within + and - 18 minutes, the head/carriage assembly needs to be replaced.

Select head 1 and check the Azumith for that head. It also should be within + or - 18 minutes.

There is no adjustment for Azumith.

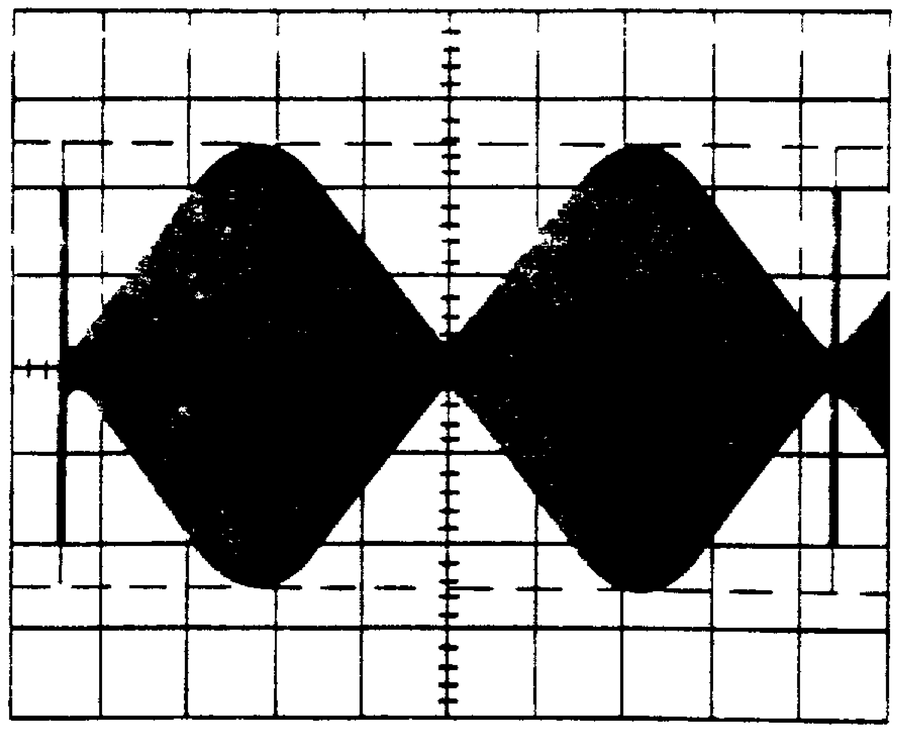
INDEX SENSOR ALIGNMENT

9) With the exact setup above, you should be able to see the index marker approximately 1/2 division in from the left edge of the screen. The Azumith pattern should be starting at 1 division from the left. Select head 0, set the time base to 50 uS. per division, and see the index marker approximately 4 divisions or 200 uS. in from the left. If it is not ~200 uS. from the left, you must adjust it.

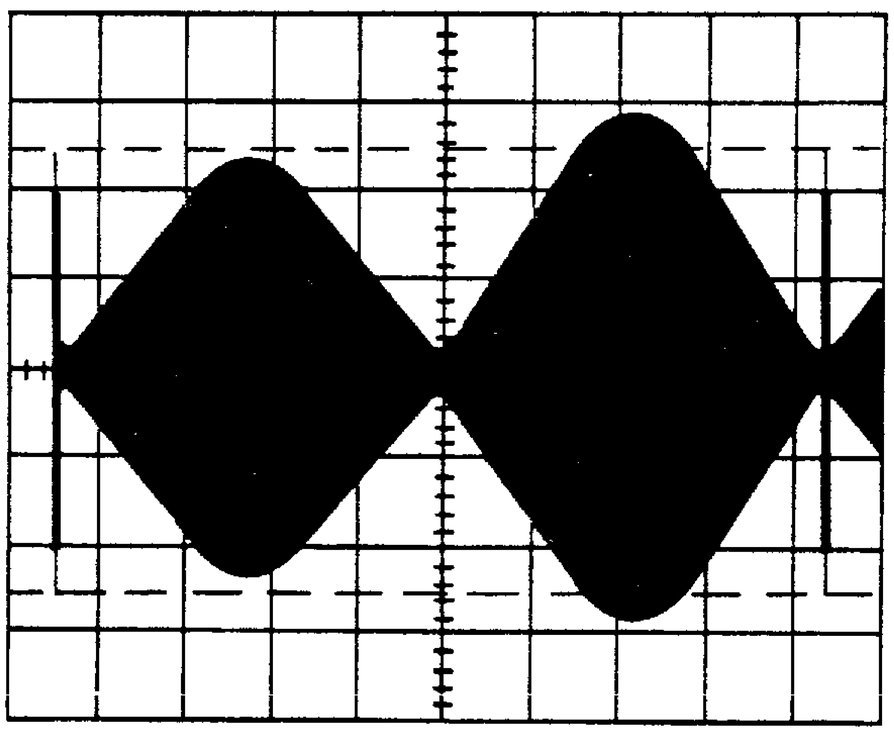
Adjust the index sensor by loosening the phillips head screw by the base of the door solenoid. There is a large washer under this screw. Carefully slide this assembly until the index marker is at 200 uS. + or - 50 uS. Carefully tighten the screw and verify that it is still in alignment.

This completes the alignment procedures for the QumeTrak 842 drive.

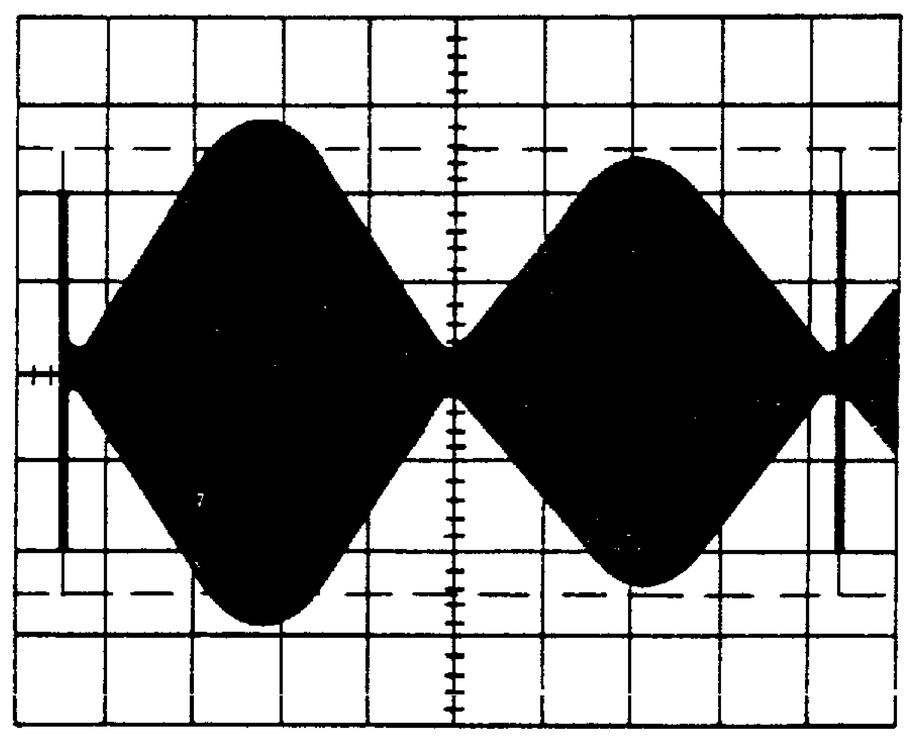
"CAT EYES" LOBE PATTERN ON TRACK 40 USING QUME CE DISK



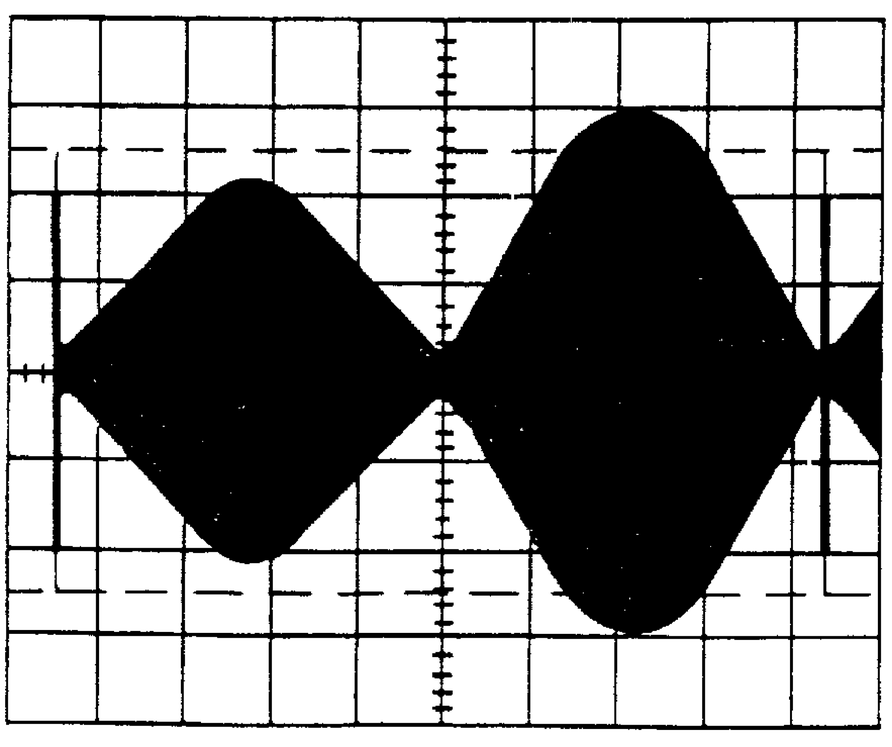
EQUAL AMPLITUDE (EXACTLY ON TRACK 40)



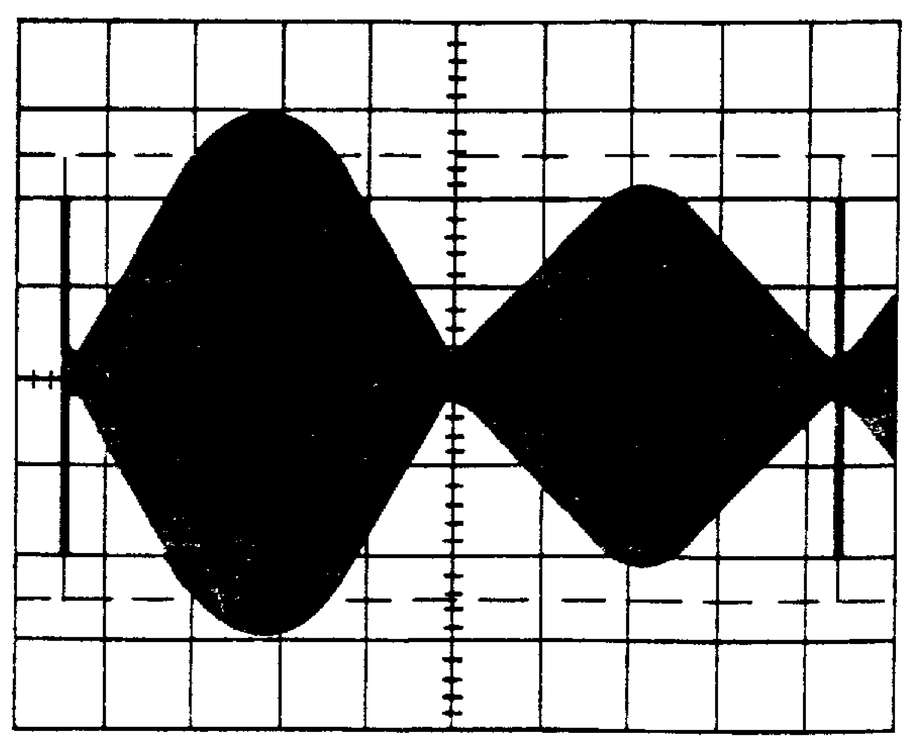
LEFT LOBE IS 80% OF RIGHT LOBE



RIGHT LOBE IS 80% OF LEFT LOBE



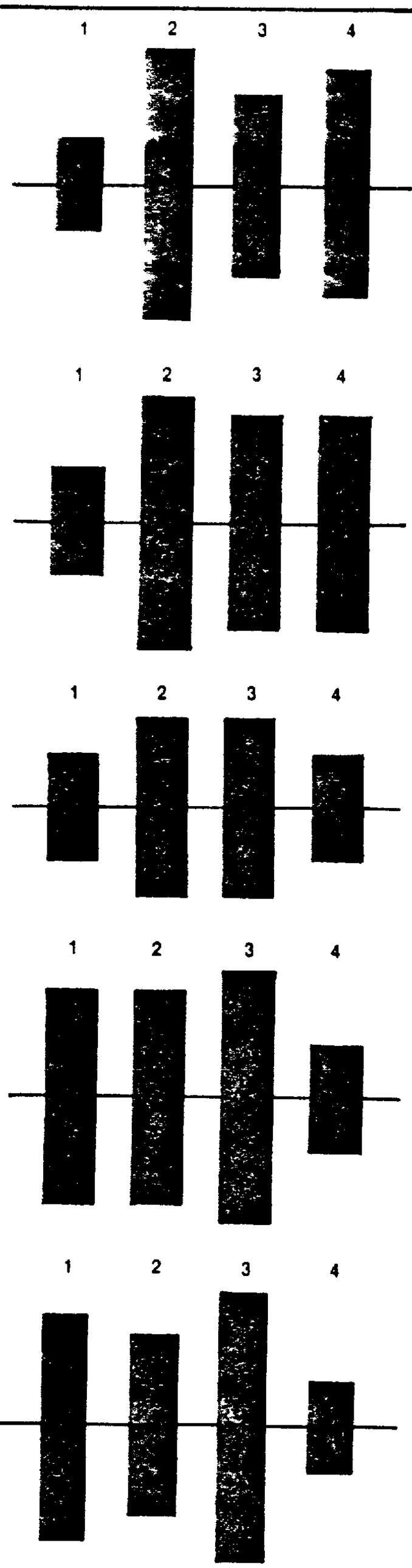
LEFT LOBE IS ABOUT 60% OF RIGHT LOBE



RIGHT LOBE IS ABOUT 60% OF LEFT LOBE

008-A 054

Figure 43. R/W Head Radial Alignment



USING A QUME CERTIFIED 12 MINUTE CE DISK (QUME P N 50235-02 OR 50236-01) THE AZIMUTH SHALL BE ACCEPTABLE IF A DRIVE SHOWS LESS THAN THE MAXIMUM ALLOWABLE 18 MINUTE AZIMUTH THE MAXIMUM ALLOWABLE AZIMUTH IS CONSI DERED TO BE ATTAINED WHEN BURST #1 IS 25% LARGER THAN BURST #2 OR IF BURST #4 IS 25% LARGER THAN BURST #3

003 A 009

Figure 21. Azimuth Check

BOARD PRODUCTS SECTION

INDIVIDUAL BOARDS

This section of the manual focuses on the individual boards that make up the system packages. First, a technical summary of the boards is given and the basic operation of the board is explained. Next, documentation for the diagnostic software used in house is described and explained. These tests include both those used in production and those used in the labs for repairs on the boards.

The final part of the material on each board is a troubleshooting chart. This chart covers problems likely to be encountered in the field. Probable causes and possible remedies for these problems are also discussed.

The purpose of this section is to give technicians the knowledge and tools necessary to repair hardware problems with COMPUPRO boards.

CONTENTS

MEMORY BOARDS
- RAM 16
- RAM 17
- RAM 21
- RAM 22

CPU BOARDS
- CPU 8085/88
- CPU 8086/87
- CPU 68K
- CPU 286

SYSTEM SUPPORT 1

INTERFACER 3
INTERFACER 4

DISK 1 AND 1A

DISK 2 and SELECTOR CHANNEL
DISK 3

COMPUTER ENCLOSURE 2

M-DRIVE/H

BOARDS DIAGNOSTIC TEST GLOSSARY

COMPUPRO MEMORY BOARDS

COMPUPRO has several memory boards, four of which are used in the system packages. All four of these boards are static RAM with at least 64K of memory on the board. Three of these boards are capable of both 8 and 16 bit operations. RAM 16, 21 and 22 will respond to SXTRQ* with SIXTN*, indicating that a 16 bit transfer can be done.

RAM 16

The RAM 16 features 64K bytes or 32K words of static RAM, using 2K byte by 8 bit, 6116 CMOS chips. This RAM is addressable on any 64K byte boundary in the 16 megabyte address space as specified by the IEEE 696 standard.

The RAM 16 dynamically switches between byte-wide and word-wide modes per the state of the sXTRQ* signal on the S-100 bus. A PAL element selects the proper memory chip or chips. The memory array is designed in such a way that when doing 8 bit operations, bytes that are odd (i.e. ending with a 1) are stored in one chip while the even bytes are stored in another. This makes it possible to do 16 bit operations using the two chips at one time. This board was designed to work with 8086/88 type processors at speeds to 10 MHz.

RAM 17

The RAM 17 features 64K bytes of static RAM, using the 6116 CMOS chip. The RAM is addressable at any 64K byte boundary in a 16 megabyte space. Four 16 kilobyte segments may be individually disabled through an on-board DIP switch. The highest 8K block is further divided into four 2K blocks which can be individually disabled. The RAM 17 is the easiest to troubleshoot since each 2 kilobyte chunk of memory directly corresponds to one RAM chip.

The windowing capabilities of the board allow its use with memory-mapped devices, or with operating systems that require a portion of memory to be global.

RAM 21

The RAM 21 features 128K bytes or 64K words of static RAM, using 16K by 1 MOS chips. The RAM 21 is addressable on any 128K boundary in a 16 megabyte address space. The board is dynamically switched between 8 and 16 bit modes by the SXTRQ* signal. A PAL is used to control the switching and to select the proper memory chips.

RAM 22

The RAM 22 features 256K bytes or 128K words of static RAM, using 8K by 8 CMOS chips. The RAM 22 is addressable on any 256K boundary in a 16 megabyte address space. The board is dynamically switched between 8 and 16 bit modes by the SXTRQ* signal. Two PALs are used to control the switching and to select the proper memory chips.

PRODUCTION TESTS - RAM BOARDS

There is one test for all of the RAM boards. This tests has several parts, and tests both the memory and the support circuitry.

MEMTEST: The first part of this test is the BIT STUCK HIGH OR LOW test. This test sends out 00Hex and FFHex to all memory locations and then reads it back. If anything other than 00 or FF comes back, an error message is given, stating the nature of the problem and the address of the error.

The ADJACENT BIT SHORTED test and the ODD EVEN BIT test are next. These tests fill the memory with 55 and AAhex to see if any data lines are shorted together. Again the error message gives the location and the nature of the problem.

Then the BIT SHIFT LEFT and the BIT SHIFT RIGHT tests send a single bit of data through one data line. The bit is then shifted to the next data line and sent through the board on that line.

The ADDRESS LINE test is similar to the BIT SHIFT tests. Each address line is tested individually by sending a bit through on each address line then shifting it to the next.

Finally, the RANDOM NUMBER test is used to detect pattern sensitivity of the board. A set of random numbers is stored in the memory and duplicated until the board is full. The data is checked when it is read back out. This test is repeated several times. As with the other portions of the test the error messages states the location and nature of the problem.

These tests work essentially as follows:

1. The board is filled with a certain byte (00,FF,AA,55)
2. One byte gets a different pattern (FF,00,55,AA)
3. The rest of the board is checked to make sure the original byte remains
4. The next byte is given the complementary pattern and the process repeats

One troubleshooting technique that is useful for all the boards but for the RAM boards in particular, is the visual inspection. Although boards shipped from COMPUPRO should have no problems like bent pins or missing or damaged parts, strange things have been known to happen in the field. Customers will often attempt to change a chip or add a feature to the board that was not designed into it. Take a very close look at all the boards before running the tests.

COMMON MEMORY BOARD PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
Board does not exist	Board not selected	Check output of extended address decoder
	Bad voltages on board	Check inputs and outputs of voltage regulators
Fails whole block of memory	Bad address buffer	Check address buffer
Fails at even 1K sectors (1000,2000)	No chip select	Check chip select on memory chip
Fails at address 00H	Problem in support circuitry	Check address buffers Check data buffers
Fails different locations	Bad ground connection on memory chip	Check ground to all chips
Intermittent failure	Dirty edge connector	Clean edge connector
	Bad ground connection on chip	Check ground on all chips

SYSTEM SUPPORT 1

The System Support 1 is a multi-function board, consisting of a real time clock with battery back-up, a RS-232 serial channel, 15 levels of interrupt control, space for 4K of RAM or EPROM with battery back-up for 2K, provision for a high speed math processor, 3 independent interval timers, and PWRFAIL generation.

The system support board takes up a block of 16 I/O ports and is addressable on any 16 port boundary. These ports are used to program the various devices; 2651 USART, clock, etc. The standard port block begins at 50 hex, and all COMPUPRO software assumes that the board is addressed there.

PRODUCTION TESTS

There are several tests to check the various parts of the board. All tests assume a 64K system with either a CPU-Z or 8085/88 CPU, J13 of the system support must be jumpered accordingly. There must be a 4K window in memory from 8000 to 8fff hex (A RAM I7 or RAM 20 is useful for windows). The switches should be set as follows: S1 paddles 4 and 5 on, the rest are off, S2 paddles 1,2,3,4 on, the rest are off, S3 paddles 5 and 7 on, the rest are off. There should be 6116 RAMs in positions U16 and U17, either a 9511, 9512, or equivalent in U13, and special headers in J7 and J8 jumpered as shown below.

```
J7   ........          J8   ........
   1 .      . 16          1 .      . 16
   2 .      . 15          2 .      . 15
   3 .      . 14          3 .      . 14
   4 .      . 13          4 .      . 13
   5 .      . 12          5 .      . 12
   6 .      . 11          6 .      . 11
   7 .      . 10          7 .      . 10
   8 .      . 9           8 .      . 9
     ........              ........
```

The first test is the serial port. If the system will boot and display on the terminal, and is able to do a DIR command then it is assumed to be reading and writing correctly.

The second test is CLOCK. This routine allows you to set the time and date of the 5832 clock chip and read it back again. You must use 24 hour format and two digits to designate hours, minutes, day, etc. (this test is self-prompting on data entry format). Once everything is set and the battery hooked up the system can be turned off. The time should be correct when CLOCK is run again and displayed. If the time is off few seconds it can be adjusted by turning C12.

The test for the math processor is either 9511 or 9512 depending on the chip. Both tests initialize the math processor, multiply two numbers, and check the result with a known answer. If "OK" appears, everything is probably working correctly. If "PERMANENT ERROR" appears it means that the routine has tried twice to get the correct respose and failed.

The two ROM sockets are tested with a standard memory test called MEMTEST. It can be set to test any block, but we have already set the switches to our standard location of 8000 to 8fff hex. The routine will go through several different tests, if there are no errors listed when it starts to repeat then all the support circuitry is good.

The last test is for the interrupt controller and is called 8259A. When it is running "!!!!" will appear over the terminal screen. Make sure your terminal is set for wraparound. The program is waiting for a low on the vectored interrupt lines. Ground a jumper and touch the other side to each of the interrupt lines VI0 to VI7. The corresponding number should be displayed. If more than one pin is touched at the same time the lowest one will be displayed, as it has the highest priority. The 8259A test also tests the timer circuitry, however it can be tested separately with the TIMER routine.

This can be very handy in troubleshooting. If you use the FIRE command, one of the timers will output a low to the VI lines it`s connected to; this will be determined by J7. A RESET will output a high on the same lines. The best way to see what`s happening is with a JADE BUS PROBE.

COMMON SYSTEM SUPPORT 1 PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
No output to terminal or no input from terminal	Wrong voltages	Check inputs and outputs of voltage regulators
	Bad Serial I/O	Check chips U3 and U4
		Replace USART
Garbage on screen	Wrong baud rate	Check U45 and X1 (COMPUPRO baud is normally 9600)
	Bad serial I/O	Change USART and U3 and U4
Clock inaccurate	Frequency off	Adjust C12
Can`t set time/date	Bad voltage to U41 at chip select and Vcc	Check for 5V at D1
	Old CLOCK.CMD program	Get an update
	X3 off freq.	Replace X3
	Bad clock chip	Replace U41
Sets wrong time	Wrong data to U41	Check all data buffers
Won`t boot CP/M 86*	No output from U16	Replace GO 86 EPROM
		Run MEMTEST to check addressing
		Check chip enable on EPROM

SYMPTOM	PROBABLE CAUSE	REMEDY
Won`t boot MP/M* 8-16	See 'Won`t boot CP/M 86" above	
	Interrupts bad	Check for J8
		Check U14, U15,timers
		Check 2 MHz clock on 8253
Math processor fails	Bad 9511/9512	Replace math chip
	Bad clock	Check J5 frequency
	Chip not initialized	Check chip select

INTERFACER 3

The INTERFACER 3 board features 8 fully programmable asynchronous serial channels. Two of these ports are capable of high speed synchronous transmission. Five RS-232 handshaking lines are available on each channel as well as two distinct interrupts.

The INTERFACER 3 was designed for operation in interrupt driven/multi-user systems. A total of sixteen interrupts are brought out for jumpering by the user to the eight vectored interrupt lines on the S-100 bus.

A five bit register is used as a user select register to activate the required serial channel. This feature minimizes the number of I/O ports necessary for running a multi-user system. Up to 32 users (four boards) can use the same 8 port addresses.

PRODUCTION TESTS FOR THE INTERFACER 3

There are two tests for the INTERFACER 3 boards. The test used in production is called TI3. The test used for repairs is called I3TEST.

TEST - TI3:

This test requires construction of test hardware. A 'master' INTERFACER 3 is used along with an INTERFACER 2. Also needed is a wire wrapped board designed for communication between the INTERFACER 2 and the board under test.

The first portion of TI3 tests the serial ports on the INTERFACER 3. Special cables are used to make it possible for the master board and the board under test to 'talk' to each other using the RS-232 handshaking lines. The handshaking lines are tested to see if they are sending or receiving signals as expected. Then the data lines are tested. What actually happens is that, while one board is sending the data the other board is receiving the data. This tests the ability to send and receive data. Each port is tested separately in this manner.

The second portion of TI3 tests the interrupt circuitry on the board. Cables run from the interrupt jumper sockets on the INTERFACER 3 to sockets on the wire wrapped board, which connect the board to an INTERFACER 2 parallel port. Again the master board and the board under test talk to each other. This time the INTERFACER 2 checks the interrupt registers on the board under test to see if it is responding to data sent to it by generating a receive interrupt. It also checks to see if it generates a transmit interrupt when it sends data out.

TEST - I3TEST:

This test is less complex than TI3. Each serial port is checked for a character, one after another. One port is selected at a time by a hardware jig. This jig has eight inputs and one output. The eight inputs are connected to the eight serial channels on the INTERFACER 3. (If a jig is not available, the user can either hook up multiple terminals or move the serial cable from one channel to the next.) One channel is selected at a time and hooked up to the terminal. This makes it possible to select any user, type a character and see it echoed on the screen. It also allows for examination of the handshaking lines by means of LEDs on the jig or on an RS-232 breakout box.

Interrupts are checked manually with an oscilloscope. The scope probe is placed at the point where each interrupt is brought out for jumpering to the vectored interrupt lines on the S-100 bus. When a character is generated on a port by the keyboard, an interrupt can be seen at the interrupt jumper socket.

TI3 can be stopped and stepped through one byte at a time. It is then possible to check a port at any stage of the cycle to see if it is being selected properly and if it is actually getting the data sent to it.

COMMON INTERFACER 3 PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
Won't run at all	On board supply voltages not correct	Check voltages at the inputs and outputs of the voltage regulators
Runs CP/M*, will not run MP/M*	Interrupt jumpers not correct.	Check jumpers and replace if needed.
	Interrupts not being generated properly	Use TI3 to check for interrupts at the jumpers
Garbageonthe screen	Bad baud clock	Checkclockgenerator output
		Check clock inputs to the USART's
One channel does not work	Channel not selected	Use TI3 to check if channel is being selected
	Bad USART	Use TI3 to check handshaking lines, and to see if correct data is echoed on to the screen.
	Bad RS-232 driver or receiver	Check handshaking lines with TI3.
Printer will not work with INTERFACER 3	Bad RS-232 driver or receiver	Check handshaking lines with TI3.

INTERFACER 4

The INTERFACER 4 board features 3 fully programmable asynchronous serial channels, two of which are capable of high speed synchronous transmission and one capable of current loop operation. Each channel has five RS-232 handshaking lines, two distinct transmit and receive interrupts and bi-directional clock drivers. Also featured on the board is a pin compatible CENTRONICS parallel interface port with handshaking lines, and a universal parallel port with 16 data, and 3 handshaking lines.

The INTERFACER 4 was designed for operation in interrupt driven/multi-user systems as well as single user systems. A user select register is used to activate the required port. This feature minimizes the number of I/O ports necessary for using several INTERFACER 3 or 4 boards.

PRODUCTION TESTS FOR THE INTERFACER 4

There is one test for the INTERFACER 4 in production. This is the I4A test. This test is broken down into several different tests for the purpose of repairs. These are, I4TEST, CTEST, I4PTST, DIPTST and I4INT.

I4A: This test requires construction if test hardware. A master INTERFACER 4 is used along with an INTERFACER 2. A wire wrapped board interfaces an INTERFACER 2 to the board under test. A printer with a CENTRONICS interface is also used.

The first portion of I4A tests the serial ports on the INTERFACER 4. Special cables are used to make it possible for the master board and the board under test to talk to each other using the RS-232 lines with handshaking. The handshaking lines are tested to see if they are sending or receiving signals as expected. Then the data lines are tested. What actually happens is that while one board is sending the data the other board is receiving the data. Each serial port is tested in this manner.

Next the CENTRONICS port is tested. This part of the test simply sends data to the printer using the handshaking lines. This tests the data lines and the handshaking lines as well as the status buffer on the board.

The universal parallel port is then tested. The port is connected to a parallel port on the INTERFACER 2 through the wire wrapped board. All combinations of data are transmitted by the board under test and received by the INTERFACER 2. Then, as in the serial part of the test, the data is sent to the board under test. This tests the ability of the port to send and receive data correctly.

The next part of the test checks to see if the DIP switch S1 works correctly. This test expects to see a logical 1 come through when the switch is off and a logical 0 when the switch is on. First the program looks for all 1s one S1 3-10 then looks for all 0s on S1 3-10. If anything else is present an error message will be displayed and the test will be terminated.

Finally, the interrupt circuitry is tested. Cables run from the interrupt jumper sockets on the INTERFACER 4 to sockets on the wire wrapped board, which connect the board to an INTERFACER 2 parallel port. The master board and the board under test talk to each other. The INTERFACER 2 checks the interrupt registers on the board under test to see if it is responding to data sent to it by generating a receive interrupt. It also checks to see if it generates a transmit interrupt when it sends data out.

REPAIR TESTS

There are several different tests used in the repair of the INTERFACER 4. They are designed to test each portion of the board independently. They also allow for more control on the part of the technician.

I4TEST tests the serial ports on the board. One channel is selected at a time and hooked up to the terminal. This makes it possible to select any user, type a character and see it echoed on the screen. It also allows for examination of the handshaking lines by means of LEDs. Three inputs to a hardware jig are connected to the three serial channels on the INTERFACER 4. Each port is checked character by character. Moving the terminal cable is acceptable if the jig is not available.

Interrupts are also tested with I4TEST. Interrupts are checked manually with an oscilloscope. The scope probe is placed at the point where each interrupt is brought out for jumpering to the vectored interrupt lines on the S-100 bus. When a character is generated on a port by the keyboard, an interrupt can be seen at the interrupt jumper socket.

CTEST tests the CENTRONICS port on the INTERFACER 4. Data is simply sent to a printer through the CENTRONICS port, making use of the handshaking lines. The results are checked by examining the message printed by the printer. A <SPACE> will end the test.

I4PTST tests the parallel port on the INTERFACER 4. This test works like the test for the serial ports. All combinations of data are sent to the parallel port on the INTERFACER 4 from the INTERFACER 2 through the wire wrapped board. The data is then sent from the INTERFACER 4 to the INTERFACER 2.

DIPTST test the dip switch on the INTERFACER 4. A hardware box containing a set of switches is connected to the parallel port connector. LEDs are used to indicate the logic level of each paddle of the dip switch S1 on the INTERFACER 4. When one of the paddles is switched the logic level will change and should be indicated by the LEDs.

COMMON INTERFACER 4 PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
Won't run at all	On board voltages not correct	Check voltages at the inputs and outputs of voltage regulators
Runs CP/M, will not run MP/M	Interrupt jumpers not correct	Check jumpers and replace as needed
	Interruptsnot being generated properly	Use I4TEST to trace interrupts at jumpers
Garbage on screen	Bad baud clock	Check clock generator output Check clock inputs to the USART's
One serial channel does not work	Channel not selected	Use I4TEST to check if channel is being selected
	Bad USART or Bad RS-232 driver or receiver	Use I4TEST to check hand-shaking lines, and to see if correct data is echoed on to the screen
Printer will not work on serial channel	Bad RS-232 line driver or receiver	Check data and handshaking lines with I4TEST

SYMPTOM	PROBABLE CAUSE	REMEDY
CENTRONICS port not printing	No strobe signal	Check strobe output at U9 pin 11 using CTEST
	Bad data buffer	Check outputs and enable of data buffer U11 using CTEST
	Incorrect status	Check outputs of status buffer U12 (should be C5 Hex when SRD* is low)
CENTRONICS port printing garbage	Incorrect status	Check outputs of status buffer U12 using CTEST
	Bad data buffer	Check outputs and enable of data buffer U11 using CTEST
Printer on UNIVERSAL parallel port prints garbage	Bad data buffer	Check outputs of data buffers U6 and U7 using I4PTST

DISK 1

The DISK 1 controller board is usually responsible for booting the system and handling the transfer of all data between the floppy disk sub system and the main memory in the enclosure. In order to perform this task the CPU issues the necessary commands to the DISK 1 controller chip, the 8272. Some versions may have the equivalent 765 chip. Commands may be from one to nine bytes in length. There are 15 seperate commands for the controller chip. These commands are listed on page 32 of the DISK 1 manual.

A command is issued by the CPU one byte at a time. After the 8272 receives a command byte, the CPU must poll the status register of the 8272 to see if the 8272 is ready to receive the next command byte. After the complete command has been received the 8272 may request a DMA cycle. If the system is functioning properly, and no other board is requesting the bus, the CPU will grant control of the bus to the DISK 1. At this point the DISK 1 will generate all control and status strobes for the system. It will also have access to the system memory and will either read from or write to the floppy disk drives. The DISK 1 transfers one byte of data per DMA request rather than multiple byte hog or burst mode.

PRODUCTION TESTS

DSKTST: This is a combination of individual tests that check the DISK 1's ability to seek, read, write, and select drives. DSKTST treats all diskettes as single-sided. Any screening operation should also use a program that knows about double-sided diskettes, such as FORMAT. The following is the menu of commands available with DSKTST:

DSKTST

```
0 - CONFIDENCE TEST
1 - SEEK TEST
2 - READ/DISPLAY TRACK,SECTOR
3 - READ TEST
4 - WRITE VERIFY READ TEST
5 - RANDOM SEEK, READ/WRITE

* - DISPLAY ALL CONTROLLED STATUS BYTES
D - SELECT DRIVE
E - DISPLAY AND CLEAR ERROR COUNT
F - FORMAT A DISKETTE
H - HELP ME
R - RESTART TEST
S - SET MODE OF DISK (SINGLE,DOUBLE)
X - EXIT BACK TO CP/M*

SPECIFY DRIVE (A:-D:)
```

0 - THE CONFIDENCE TEST: This test is a combination of tests 1,3,4, and and 5.

1 - SEEK TEST: This test has the drive seek between tracks. There are a variety of patterns, e.g. trk 0, trk 1, trk 0, trk 2, trk 0, trk 3, etc.

2 - READ/DISPLAY TRACK SECTOR: This test dumps out data from the diskette. It gives the operator the choice of which track and sector he wishes to view. This is useful for dumping a specific sector's data from the diskette.

3 - READ TEST: This test reads data from the diskette starting at track 0 and sequentially reads each sector and track. Each sector is checked for errors.

4 - WRITE VERIFY, READ TEST: This test writes sectors of F0 hex out to the diskette and reads them back again. F0 is thought to be the most difficult byte to successfully write and read.

5 - RANDOM SEEK READ/WRITE: This test selects a random track, goes out and reads its headers, and confirms that it is on the proper track. It writes data to a sector and confirms that the data is correct. Then it steps to a new track and repeats the process.

* - DISPLAY ALL CONTROLLER STATUS BYTES: If this key is pressed and any of the DSKTST selections are run, the first byte will be displayed each time a command is given to the 8272.

D - SELECT DRIVE: this gives the operator a choice of drives A - D.

E - DISPLAY AND CLEAR ERROR COUNT: There are four memory locations that hold the error count for,

READ ERRORS:
WRITE ERRORS:
SEEK ERRORS:
SOFT ERRORS:

This command reads these memory locations and then clears them. When a hard error occurs, DSKTST displays what command was given and the resulting status bytes. This is a read error:

ERROR, Func=46 Status bytes=40 01 00 08 00 01 03

Read 2d command Status 0,1,2,trk,head,sector,density

An explanation of the 8272's commands and status bytes begins on page 34 of the DISK 1 manual.

F - FORMAT A DISKETTE: This command calls up the format routine. It formats only one side of the diskette.

H - HELP ME: This or any other undefined letter calls up the DSKTST menu.

R - RESTART TEST: This command also calls the DSKTST menu.

S - SET MODE OF DISK: This command allows the user the ability to inform the program at what density the diskette is formatted.

X - EXIT BACK TO CP/M: This command returns the user back to the operating system.

In house testing of the DISK 1 consists of booting up the system, adjusting the data separator (phase lock loop), formatting single and double sided diskettes at both single (128 bytes per sector) and double (1024 bps) density. Then the TMA arbitration circuit is tested while DSKTEST option 0 is running.

BOOTING THE SYSTEM: The EPROM on board the DISK 1 contains the necessary code to boot the system. 8080 code is provided for 8085 and Z-80 processors. 8086 code is provided for the 8088, 8086, and the 286 CPUs. 68000 code is provided for the 68000 processor.

This EPROM contains the instructions the 8272 needs to begin loading the CP/M operating system off the diskette into the system memory. Requirements for booting are a CPU, enough static RAM for your CP/M (usually 64K), a DISK 1, at least one floppy disk drive and a COMPUPRO CP/M diskette matching the processor.

ADJUSTING THE DATA SEPARATOR: Place a scope probe on the right side of R9, adjust your scope to approximately .5 volts/division and 1 ms per division. Run DSKTST and select the appropriate drive and select test option #3 (read data).The 8272 will begin to read data off the drive. The signal on the scope will appear as a wide band, with a narrower band in the center. The narrow band will arch up or down as the trimpot R35 is adjusted with a screw driver. The desired result is for the narrow band to have no arch at all and to be centered in the broad band. With the signal in this position, the phase lock loop is properly adjusted.

FORMATTING A DISKETTE: Place a double sided diskette in drive B: and call up the FORMAT routine. Format the disk twice, once at 128 bytes per sector (single density) and once at 1024 byte per sector (double density).

On the screen an 'F' will appear as each track is formatted. Then the format will be verified. If a 'V' appears for each sector, all is well. If an 'S' appears for any track a soft error has been detected. These errors indicate that the phase lock loop had some trouble in reading the data during verification. If an 'E' appears a hard error has been detected. These errors indicate that the DISK 1 could not read the data and the data was not received by the system. The severity of the problem is shown by the number of errors present when the verification is complete.

TESTING THE T.M.A. ARBITRATION CIRCUIT:

DMATST: This test is designed to test the DMA assertion of the DISK 1. The test simply utilizes the MPX to assert its priority, which is lower than the DISK 1, and make sure the DISK 1 still gains control of the bus.

To run this test two additions must be made to the standard system: a MPX-1 board and a 64K RAM board. The Ram board is necessary because the mode in which the MPX-1 asserts its priority is by accessing system memory, at extended page 1. If DMATST is run and there is no MPX, the message will state 'loading MPX' and the system will crash. This is because the test received no information back form the MPX-1. If there is an MPX 1, but no RAM, the message will state that the MPX-1 was loaded with a program, but when it went to search for system memory, it found none.

To terminate the MPX, the program DMAEND is invoked. This will tell the MPX to stop searching, and also print out whether there have been any memory errors while it was running.

Parameters: Standard COMPUPRO serial configurations
MPX DMA priority is 0EH
RAM is from 10000H to 1FFFFH
MPX attention port is 0FH

Switch Settings:

MPX	S1 off	S1 on	S2 off	S2 on
		1 x	x	1
	x	2	x	2
	x	3	x	3
	x	4	x	4
	x	5		5 x
		6 x		6 x
		7 x		7 x
	x	8	x	8

RAM 17	S1 off	S1 on	S2 off	S2 on
	x	1	x	1
		2 x		2 x
		3 x		3 x
		4 x		4 x
		5 x		5 x
		6 x		6 x
		7 x		7 x
		8 x		8 x
		9 x	x	9
		10 x		10 x

SINGLE STEPPING TECHNIQUE: A useful technique for troubleshooting the DISK 1 is single stepping through the EPROM boot program. A listing of the PROM is available on the master CP/M 2.2 diskette in the file named GBROM.ASM.

This method of troubleshooting is particularly helpful when the system will not boot up. The address and data lines can be examined easily

as well as control signals. The DATA INT signal can be examined to see if it is being received by the CPU. DATA INT is fed through U9 (port C2) to become data bit 7. It is sent to the CPU on the data bus. It indicates that the 8272 completed the previous instruction.

There are two locations in the boot PROM where data bit 7 is polled by the CPU. The first is RCAL2: this is located after the recalibrate drive command. The second is located after the read command. It is labeled READ2.

A word of caution- the single stepping device e.g. the Jade bus probe or the front panel, may show that data bit 7 is being generated when in reality it is floating. This should be checked with a scope or logic probe.

Single stepping through the DMA cycle can also be helpful. In order to do this XRDY and RDY should be shorted together and held low to stop the system. Then open the doors on the drives and lift pin 1 of U8. Using a logic probe short the lifted pin 1 to its respective socket pin. The probe should show a low. While holding the probe in place reset the system. Wait a moment then stop the system and remove the logic probe. Then single step once. The address bus should show 00 01 00 hex, the TMA and disable lines should be asserted. The purpose of this procedure is to make sure the DISK 1 is initializing the DMA address properly.

COMMON DISK 1 PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
System won't boot	Bad voltages on board	Check voltages at inputs and output of voltage regulators
	Bad clock to controller chip	Check clock on U14
	No board select	Check BDSEL* on U40
	Bad address buffer	Check U41 and U19
	Bad disk drive or diskette	Swap with known good ones
Head loads once then crashes	Incorrect DMA address	Check data buffers U44 and U5 DMA counters U12 U13,U25,U26,U29,&U30 Address buffers U43, U42, U39
	Bad or wrong diskette	Try another disk
Clicking sound over and over	EPROM not disabled	Check U24a for SER OUT STB* to produceBOOT DIS* that negates BOOT
Head loads over and over	Phase lock loop out of adjustment	Adjust R35
	P.L.L. defective	Check P.L.L circuit shown on pg. 50 of DISK 1 manual
Lights on drives do not blink	DISK 1 is not scanning drives	Check U31 inputs and outputs

SYMPTOM	PROBABLE CAUSE	REMEDY
Soft errors on format	Phase lock loop out of adj.	Adjust R35 Swap U5 and/or U19
Lights on both drives brightly lit	50 pin cable connected backwards	Turn cable around on one end
Hard errors on format between tracks 41 and 76	Bad resistor at R27 Bad pulse on U34 pin 6	Check R27 for correct value Check U34 for COMPWRITE*, should change pulse width at track 41 from 1.6us to 1.3us, 1.5 on double density
System boots and puts banner on the screen but does not give prompt	P.L.L. circuitry is not going to double density mode DATA pulse from U19a not dropping from lus to .5us	Swap controller chip Check to see MFM line is grounded Check diode matrix

DISK 1A

The DISK 1A controller board is essentially identical to the DISK 1 board except it has the ability to run both 8" and 5 1/4" drives at the same time. Additional features include the ability to have a larger EPROM with more routines, digital instead of analog data separation and write pulse generation, and alternate drive select and head load capability with motor timeout. The only feature missing from the DISK 1A is the software serial port. Since the DISK 1A uses the same controller chip and has all same port addressing that the DISK 1 had, it is software compatible. All the new features of the DISK 1A were added to the unused ports and bit positions of the DISK 1.

PRODUCTION TESTS

The production tests are almost identical to the DISK 1 production tests, and we refer you at this time to the DISK 1 section for an explanation of the first tests for the DISK 1A.

DSKTST: See DISK 1 section.

In house testing of the DISK 1A consists of booting up the system, formatting double sided diskettes at both single (128 bytes per sector) and double (1024 bps) density in both 8" and 5 1/4" drives. Then the TMA arbitration circuit is tested while DSKTEST option 0 is running. Finally a PIP test is run to verify that data can be passed between the 8" and 5 1/4" floppies.

BOOTING THE SYSTEM: The EPROM on board the DISK 1A contains the necessary code to boot the system. 8080 code is provided for 8085 and Z-80 processors. 8086 code is provided for the 8088, 8086, and the 286 CPUs. 68000 code is provided for the 68000 processor.

This EPROM contains the instructions the 8272 needs to begin loading the CP/M operating system off the diskette into the system memory. Requirements for booting are a CPU, enough static RAM for your CP/M (usually 64K), a DISK 1A, two floppy disk drives and a COMPUPRO CP/M diskette matching the processor.

ADJUSTING THE DATA SEPARATOR: Since the data separator is entirely digital and contained in the one 8 pin IC (U 10), no adjustment is neccesary.

FORMATTING A DISKETTE: Place a double sided diskette in drive B: and call up the FORMAT routine. Format the disk twice, once at 128 bytes per sector (single density) and once at 1024 byte per sector (double density).

On the screen an 'F' will appear as each track is formatted. Then the format will be verified. If a 'V' appears for each sector, all is

well. If an 'S' appears for any track a soft error has been detected. These errors indicate that the phase lock loop had some trouble in reading the data during verification. If an 'E' appears a hard error has been detected. These errors indicate that the DISK 1A could not read the data and the data was not received by the system. The severity of the problem is shown by the number of errors present when the verification is complete.

TESTING THE T.M.A. ARBITRATION CIRCUIT:

DMATST: This test is identical to that of the DISK 1, and we refer you to this section under DISK 1.

SINGLE STEPPING TECHNIQUE: A useful technique for troubleshooting the DISK 1A is single stepping through the EPROM boot program. A listing of the PROM is available on the master CP/M 2.2 diskette in the file named GBROM.ASM.

This method of troubleshooting is particularly helpful when the system will not boot up. The address and data lines can be examined easily as well as control signals. The CINT signal can be examined to see if it is being received by the CPU. CINT is fed through U9 (port C2) to become data bit 7. It is sent to the CPU on the data bus. It indicates that the 8272 completed the previous instruction.

There are two locations in the boot PROM where data bit 7 is polled by the CPU. The first is RCAL2: this is located after the recalibrate drive command. The second is located after the read command. It is labeled READ2.

A word of caution- the single stepping device e.g. the Jade bus probe or the front panel, may show that data bit 7 is being generated when in reality it is floating. This should be checked with a scope or logic probe.

Single stepping through the DMA cycle can also be helpful. In order to do this XRDY and RDY should be shorted together and held low to stop the system (this allows the bus probe to stop the DMA cycle). Then open the doors on the drives and lift pin 2 of U2. Using a logic probe short the lifted pin 2 to its respective socket pin. The probe should show a low. While holding the probe in place reset the system. Wait a moment then stop the system and remove the logic probe. Then single step once. The address bus should show 00 01 00 hex, the TMA and disable lines should be asserted. The purpose of this procedure is to make sure the DISK 1A is initializing the DMA address properly.

COMMON DISK 1A PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
System won't boot	Bad voltages on board	Check voltages at inputs and output of voltage regulators
	Bad clock to controller chip	Check clock on U14 pins 19 and 21
	No board select	Check BDSEL* on U45
	Bad address buffer	Check U46 and U33
	Bad disk drive or diskette	Swap with known good ones
Head loads once then crashes	Incorrect DMA address	Check data buffers U50 and U51 DMA counters U22-24, U35-37 Address buffers U47-49
	Bad or wrong diskette	Try another disk
Clicking sound over and over	EPROM not disabled	Check U56a for MTR STB* to produce <NOT> BOOT that negates PROM EN*
Head loads over and over	Data on diskette bad	Use good diskette
	P.L.L. defective	Check P.L.L circuit for required signals
Lights on drives do not blink	DISK 1A is not scanning drives	Check U5 inputs and outputs, and J1-4 for proper drive size

SYMPTOM	PROBABLE CAUSE	REMEDY
Soft errors on format, or read	Drive bad or out of alignment Data separator bad	Replace with known good drive Swap U10
Lights on both drives brightly lit	50 pin cable connected backwards	Turn cable around on one end
System boots and puts banner on the screen but does not give prompt	P.L.L. circuitry is not going to double density mode	Check U10 pin 5 for MFM* signal Check to see MFM line is grounded

DISK2 and SELECTOR CHANNEL

The DISK2 and SELECTOR CHANNEL are a two-board set designed to control Winchester hard disk drives of the Shugart SA4000 type interface. Such drives include the Shugart SA4000 series as well as the Fujitsu 2300 series. Software drivers which support one drive (not multiple drives) are currently available for the Fujitsu 2301 (10 Mbyte) and 2302 (20 Mbyte) drives. Shugart drives are not currently supported in software. Software drivers for other hard disk drives may be available from CompuPro or CompuPro system centers in the future. It should be noted that Memorex 101 series drives of 1982 vintage or later are actually relabelled Fujitsu 2300 series drives and are supported in CompuPro software.

The DISK2 board has the hardware to perform reads, writes and seeks on the disk drives, and to request, arbitrate and accept control of the system bus. The SELECTOR CHANNEL hardware drives the address, control and status lines on the system bus during DMA cycles. The DISK2 cannot function without a companion SELECTOR CHANNEL. The SELECTOR CHANNEL could serve as companion to other DMA devices that are designed to use it, though no such devices have been produced.

The heart of the DISK2 is a finite state machine; essentially a CPU implemented in a prom sequencer. Address bits serve as instruction input and data output serves as control signals. Three of the address bit inputs to the finite state machine are driven by a command register, while the other address input bits are fed back from the machine's latched outputs. Macro-instructions are loaded in from the command register and clocked through their micro-steps by a counter. The micro-instructions can either hang waiting for a condition to come true or continue until the macro-instruction is completed. Such a design technique was chosen because it affords high-speed, low-cost controller circuitry.

The rest of the DISK2 circuitry is summarized as follows:

1. disk drive interface circuitry including:
   a) a universal shift register and related circuitry for conversion from parallel to serial (write) and serial to parallel (read)
   b) cyclic redundancy check (CRC) circuitry for confirming that data transfer was performed without error for both header fields and data fields (the CRC circuitry actually compares header fields as they are read off the disk to header fields stored in buffers and notifies control circuitry when a desired header has been found).

2. S100 bus interface circuitry including:
   a) DMA arbitration circuitry
   b) registers which hold command and status bytes, and track, head and sector header information

The SELECTOR CHANNEL hardware can be divided into three blocks:
   a) system address, control and status line buffer/drivers and timing circuitry for DMA transfers
   b) registers which hold command and status bytes
   c) hard disk boot prom circuitry

PRODUCTION TESTS

The production tests for the DISK1 and SELECTOR CHANNEL are the same except that there is an additional test for the SELECTOR CHANNEL which confirms proper operation of its boot circuitry.

Part of the test simply involves accessing the hard disk under the CP/M operating system. The more demanding part of the test is a program written in the Forth programming language which exercises the read, write and format circuitry of ther DISK 2. (Some component routines of the FORTH language test program have changed since the last release of this training manual.)

Here are the steps of the production test.

1. Type "DIR X:". This test simply uses the CP/M command DIR X:, where X = a logical drive on the hard disk which has files on it. Successful execution of this command confirms that the controller set can actually talk to the disk under control of CP/M. Error messages such as "BDOS ERROR on DRIVE X" indicate a failure of the test.

2. Type "PD2". This loads an exhaustive Forth language test that can manipulate the controller hardware with a series of very simple commands.

3. Type "PROT". This runs a series of routines which accomplish the following sequence of actions:

   a) read a given sector to synchronize the Johnson counter,
   b) format the first 16 cylinders,
   c) write data to sectors just formatted,
   d) write data to sectors at the outer, middle and inner cylinders of the disk,
   e) read and check data just written,
   f) repeat steps d & e with a complimentary data patter,
   g) and finally, destroy some of the newly formatted headers to make sure that the next controller tested will have to format its own headers in order to pass the test

The above test assures that the controller can read from, write to, and format the disk drive without error. When the test

detects a failure, it suspends operation and displays the DISK2 status byte and sector number for the error detected. Also, while the test is running, a transistor which inhibits writing to the hard disk on power down is tested. The transistor emitter node voltage is examined with a scope probe ensuring that it dips below 0.5 volts on writes and rises above +2.0 volts when not writing, thus ensuring that the write gate is turning on and off.

As mentioned above, the SELECTOR CHANNEL boot prom circuitry is tested in production. To accomplish this, the hard disk is formatted and an appropriate CP/M system is generated on the hard disk (these appropriate CP/M files can be found on a CP/M 2.2 floppy), then the DISK1 boot prom is disabled, the DISK2 boot prom is enabled and the system booted from the hard disk. After repeated successful booting, a directory is accessed from both the hard and floppy drives to ensure proper operation of CP/M when it is booted from the hard disk.

REPAIR TEST

There are no additional programs for testing the controller in the repair lab environment, but PD2 can be used more extensively than it is in production. Specifically, the simple FORTH words that make up the production test can be used individually to exercise basic blocks of the controller hardware, allowing the technician to isolate the cause for failures discovered in the production test. Generally the production test is first rerun by the technician so that he may examine the status of the controller, then more simple commands are issued to the controller accordingly.

The status of the controller is read from a port on the DISK2; the port is implemented in hardware by an octal latch (U37). The individual status bits can be interpreted as follows:

BIT #	Signal name	Function
7	ATTENTION*	Tells whether status is latched or transparent (low is latched)
6	TIME OUT	Indicates drive could not find selected header
5	CRC ERROR	Indicates CRC error on data read
4	OVER RUN	Indicates that last data read over-wrote previous data before it could be transferred into memory
3	READY*	Indicates disk drive is ready
2	SEEK COMPLETE*	Indicates disk drive has reached the track it was sent to
1	WRITE FAULT*	Indicates disk drive has detected a write error
0	TRACK 0*	Indicates drive head at track 0

* indicates active low

After the status of a faulty controller is read (by running the PROT program), the nature of the problem is determined by interpreting the individual bits of the status byte; the most frequently set bits are TIME OUT, CRC ERROR, and OVER RUN. When the particular problem is determined, the following Forth words are useful in isolating the source of the problem:

LIMIT not a command, but a constant which marks the beginning of a buffer in memory which is used in conjunction with other words

DUMP a command word which displays memory contents at the terminal for inspection
syntax: beginning adrs DUMP e.g. LIMIT DUMP

FILL a command word which fills a block of memory with bytes specified in the command line
syntax: beginning adrs blocklength data byte FILL
e.g. LIMIT 400 E5 FILL

BUILD a command word which creates an image in memory to be used for formatting the headers of a single track; the track and head data used are taken from variables which reflect the head's current postion on the disk

FMTTRK a command word which formats a single track using a memory image placed at LIMIT usually by the BUILD command

HR/W "hard read/write", a command word which either reads data from a specified area on the disk to a memory buffer or writes data from a memory buffer to a specified area on the disk
syntax: memory adrs logical sector 0 or 1 HR/W
note: a zero in the third position causes a write, a one causes a read; the logical sector is a unique number assigned to each sector according to the following formula:
logical sector = (cylinder# * 88) + (head# * 11) + sector#

CHKFMT a command word which reads the headers from the track at which the head is positioned, without checking header CRC data, and writes them into memory at LIMIT; this allows examination of possibly erroneous headers

SEEK a command word which moves the head to a specified track
syntax: track# SEEK

The use of the above commands is dictated by the nature of the problem. For example, in the event of a TIME OUT error, it is desirable to inspect the actual headers of the track in question as they come off the disk. A command sequence of

CHKFMT LIMIT DUMP

will read the headers of the current track into memory and then display that memory image at the terminal. The headers can now be examined to see if they are actually trashed (a condition which may indicate faulty read/write circuitry), or if they appear to be in order (which may indicate that the CRC circuitry is incorrectly identifying them as bad).

If the headers appear trashed, the next step is determining whether they are being written or read improperly. One way to do this is to repeat the above command sequence and observe whether the headers seem to be trashed in the same way each time (which could indicate that they were formatted incorrectly to begin with and that the read circuitry is reading them without error), or if they show a different image each time they are read (probably indicating bad read circuitry). Alternately, a known good controller board can be swapped into the system to read the headers using the command sequence

track# SEEK

to move the head to the suitable track, andagain

CHKFMT LIMIT DUMP

to actually read the headers. If the headers read by the known good board still appear bad, then the write circuitry of the board under repair is probably bad.

When it is necessary to format a single track the command sequence

BUILD FMTTRK

will format the current track from an image built in memory.

The source of data CRC errors can be tracked down in much the same way as TIME OUT errors, except that the HR/W command is used in place of the CHKFMT command when writing or reading data fields. For example, to read the data from logical sector 100h the command sequence

LIMIT 100 1 HR/W

will read from that sector and write its image into memory starting at LIMIT; the DUMP command can then be used to display the image at the terminal. Data can be written to sector 100 by first running the command sequence

LIMIT 400 (data byte) FILL

to fill 400 bytes of memory starting at LIMIT with the specified byte and then

```
LIMIT   100   0   HR/W
```

to actually write it to disk.

This completes discussion of the more important command words in the D2 software.

TROUBLESHOOTING TIPS TABLE

SYMPTOM	PROBABLE CAUSE	REMEDY
Hangs up on DIR X: and/or D2 test	Drive not selected	Check drive select
	No clock received from drive	Check clock on drive
	finite state machine locked-up	check PROMs
BDOS error on DIR X:	Problem with CRC circuitry	Check CRC circuitry
DIR X: fails to display entire directory	Problem with byte/sector counting circuitry	check U27/28 counters, U29 multiplexor, and related gates
failure to boot off f hard disk	Bad EPROM on SELECTOR CHANNEL	Change EPROM
fails D2, status byte 23 or 43	data CRC error or TIME OUT error	check clock, CRC, shift register, and header information buffers
fails D2, status byte 13	data OVER RUN	check DMA arbitration and assertion circuitry, and selector channel strobes and address/data drivers
fails D2, status byte 11	data OVER RUN and WRITE FAULT	check finite state machine and related circuitry

BH,MM,RK &FB 4/25/83

REVISEDBY BH ON 15AUG84

31 JULY 84

DISK 3 TRAINING MANUAL

The CompuPro DISK 3 is a DMA hard disk controller for the ST506 interface to 5-1/4" Winchester hard disk drives. This interface is compatible with the ST412 interface. The controller card can be jumpered at the factory for use with drives of the SA1000 type interface; special order is required.

SOFTWARE SUPPORT

Since these interfaces include a wide variety of drives from different manufacturers, the format program supplied with CompuPro operating systems has been written to accommodate many different drives. When you enter the format program, the screen displays the default settings of various drive parameters such as number of cylinders, number of heads, and step rate. You are then given the opportunity to change any of these parameters to accommodate your drive.

However, CompuPro operating systems only accommodate certain hard disks in their hardware interfaces:

1. Quantum 5-1/4" Q540 (40 Mbyte) and Q520 (20 Mbyte)
2. Quantum 8" Q2080 (80 Mbyte, requires special order SA1000 DISK 3)
3. Seagate ST-506 (5 Mbyte)

Other hard disks that are close equivalents to the above drives may or may not work with CompuPro operating systems.

Drives that are not supported by CompuPro operating systems can be supported in two ways: modification of source code for CP/M, MP/M and Concurrent CP/M; or creation of resident system processes (RSPs) for inclusion in the gensys process for MP/M and Concurrent CP/M. Such support should only be attempted by serious programmers experienced with assembly language and CP/M and MP/M system level programming.

CURRENT HARDWARE

There are two versions of the DISK 3 that are considered current. One is the "F" revision board with ECO 30 (and PROM 224C); the other is the "D" revision board with ECO 31 (and PROM 223C). Both versions are free of any known bugs. The only difference is that the "F" revision boards are able to use one of their signals (reduced write current line) as an extra head-select line to accommodate drives with sixteen heads (e.g. Maxtor large capacity drives). This difference should be insignificant to most customers.

Boards of the following revisions can be upgraded in the field:

1. revision "D" ECO 19 (qualified technicians only)
2. revision "F" without ECO (qualified technicians only)

3. revision "D" ECO 29 (new PROM needed)
4. revision "F" ECO 28 (new PROM needed)

CompuPro prefers that boards of the first two categories be returned to the factory for modification and thorough test. Contact Customer Service for details on the modification procedure.

CURRENT FORMAT PROGRAM

The format program, DISK 3, has been modified since its first release and should be upgraded to the current level. The current version is 2.6, and drives that were formatted with earlier versions should be reformatted with the 2.6 version and current hardware.

To determine the level of your DISK 3 program, type "DISK3 ?". The screen should display a list of optional parameters for the DISK 3 program, and at the top of that list should be the version number.

HARDWARE FUNCTION

The DISK 3 hardware is very complex and CompuPro expects that any intensive troubleshooting of it will be done by factory technicians. However, some background is presented here in hopes that it will help the system center technician to identify problems in the field.

The DISK 3 is an intelligent DMA controller. When its I/O port is "poked", it fetches a command block from system memory and executes that command. Access of the disk drive is performed at disk speed (5 Megabits/sec.) and data is transferred directly to or from memory.

The DISK 3 circuitry can be divided into two main parts: those circuits related to the system interface, and those circuits related to the disk drive interface.

SYSTEM INTERFACE

An 8085 microprocessor controls the system interface; it runs a low level program contained in a PROM (U23). It responds to a "poke" of the DISK 3 port by requesting the bus from the system master. When the bus is granted, the 8085 loads a command block (known as the I/O parameter block or IOPB) from system memory into its RAM (U33), and releases control of the bus. The 8085 then interprets the command block and programs the disk interface controller to perform any disk related activity.

When the disk interface controller has completed an operation it tells the 8085. If there are any steps left to the command sequence, the 8085 programs the disk interface controller to continue. If the command sequence is complete, the 8085 reads the disk status and writes it the command block in memory. The system master can now read status that indicates the results of the operation.

DISK DRIVE INTERFACE

The main parts of the disk drive interface are the controller and data separator.

The disk drive interface is controlled by a finite state machine (FSM). An FSM is a circuit made up of discrete logic and memory which can accomplish the tasks of a microprocessor. The heart of the FSM is its memory; the memory's address inputs function as instructions, and its data outputs function as control and data signals. Each memory cell of the PROM contains a different step in an instruction sequence.

The FSM on the DISK 3 is based on a bipolar PROM (U34) and uses PALs to process its outputs into useful control signals.

The DISK 3 FSM is programmed and started by the 8085. The FSM then takes control and carries out the instruction. If any disk data is to be transferred to or from system memory, the FSM requests the bus from the system master. When the bus is granted, the FSM directs the timing of the S100 bus cycle.

When the FSM has completed its instruction, it signals to the 8085.

The data coming from the hard disk is a complex signal composed of clock and data components. The actual serial data is separated from the complex signal by a circuit aptly called the data separator.

The DISK 3 data separator is composed of exact value analog components and high-speed digital components. ECOs 7 and 19 greatly modified this circuit.

Thorough description of the circuit is beyond the scope of this manual. Problems with its operation should be directed to the Technical Support Department at CompuPro.

CIRCUIT LOCATION

Here is a guide to the location of circuits on the schematic in the DISK 3 technical manual.

PAGE 1: 8085, RAM and EPROM
S100 data and address buffers
internal board strobes
vectored interrupt jumper posts
I/O port decode and reset

PAGE 2: DMA signal and priority interface
S100 status and control
FSM
overrun detect circuit
wait state sync. circuit

PAGE 3: data separator
disk data write clock
hf data detector
drive status and control interface

r/w multiplexer
Vcc fail detect
drive select buffer

PAGE 4: differential receivers and transmitters
r/w mux/demux and data compensator

TEST SOFTWARE

PRODUCTION TEST

The production test for the DISK 3 begins with an exercise of the board with a FORTH language program. Then, the board is used to format, read and write a hard disk under CP/M 8-16 and MP/M 8-16.

The FORTH test is a series of routines or FORTH words that test the various commands that the controller must execute. It requires a simple CP/M 80 system with a DISK 3 and hard disk. The test writes garbage to the hard disk, so use a hard disk with expendable data on it.

Failure in any one of the routines should cause that routine to halt and an error message to be displayed at the screen. Succesful completion of a routine should cause an "OK" to be displayed at the terminal.

The procedure for running the FORTH test:

1. Boot the FORTH test diskette; CP/M 80 should sign on.

2. Type "dv driver.blk". This causes the FORTH program to sign on with the DISK 3 applications loaded.

3. Type "fst". This is the first actual step of the test. It reads 500 sectors in the area of the disk that has precompensated data.

4. Type "ocat". This reads 500 sectors from the disk in an area that does not have precompensated data.

5. Type "tst". This is a combination of routines that read, write, and format various areas on the disk.

6. For extended running of this test, type "tst many". This causes the "tst" routine to be repeated until a key is struck at the terminal.

The FORTH test proves that all basic operations of the controller can be accomplished without error.

The CP/M and MP/M portion of the test assures that the DISK 3 can access the hard disk under control of the operating system. This test requires an 85/88 based system that can boot CP/M 8-16 and MP/M 8-16. Since it includes formatting the hard disk, use a hard disk which has expendable data on it.

The CP/M and MP/M test procedure is very simple:

1. Boot a floppy-only version of CP/M 8-16.

2. Type "disk3 format verify", and answer "y" to the questions. This does a simple format and verify of all of the header and data fields of the hard disk. It takes about 20 minutes.

3. Boot a hard disk version of CP/M 8-16 (CPMH40.SYS).

4. PIP the contents of the floppy diskette to the hard disk (drive A:).

5. Type "dir a:" to read the directory of drive A:.

6. Boot a hard disk version of MP/M 8-16 (MPM340.SYS).

7. Read the directory of drive A:.

8. PIP the contents of drive A: to drive B:.

9. Succesful completion of all the above steps without error messages completes the production test.

LAB TEST

The lab test for troubleshooting is the same as the production test. But the lab technician is expected to better understand the error message(s) generated by a failure.

Generally, any errors in the CP/M and MP/M testing should show up in the FORTH testing. All error messages in the FORTH test should give an error number. This number is really the status byte returned by the DISK 3 after running the instruction that failed. The table on page six of the DISK 3 technical manual shows the meaning of the status byte.

The FORTH test software contains routines, or words, which are not used in normal testing, but which are useful when troubleshooting a failed board. They are useful because they allow you to perform simple low level operations on the disk like read a single sector or format a single cylinder. To use these words, boot the DISK 3 FORTH test diskette and type "dv driver.blk". An explanation of the more useful words follows.

1. SPECIFY - "Q540 specify" - sets up the specify block of drive relative parameters. (See page 10 of the Technical Manual for an explanation of the specify block.)

2. INIT - resets DISK 3 circuitry, selects drive 0, and sets up IOPB.

3. HOME - causes the drive to move its heads, or seek, to sector 0.

4. ST - does a "Q540 specify init home" all in one command for convenience. Since this command does essential initialization of the drive under test, it should be run at the beginning of every session with the FORTH words.

5. NOOP - inserts a NO OPERATION command into the IOPB and pokes the DISK 3. A return to FORTH (signalled by an "OK" at the screen) after this word means that the DISK 3 system interface probably correctly accessed the IOPB from system memory, interpreted and executed the command correctly, and returned status to the IOPB indicating completion. This is a good word to start at when troubleshooting a DISK 3 that hangs upon any attempt to poke it.

6. SEEK - "100 seek" - seeks, or moves the heads, to the cylinder specified.

7. FMTCYL - "100 FMTCYL" - formats the cylinder specified.

8. BUF - a constant which points to a buffer in system memory used for read and write operations.

9. FILBUF - "E5 FILBUF" - fills the system buffer memory with the hex byte specified.

10. DU - "buf du" - dumps, or displays at the terminal, the contents of 64 bytes of memory beginning at the address specified (here it is the value BUF).

11. RD - "10 rd" - reads the sector specified into the buffer.

12. RDS - "20 10 rds" - read multiple sectors into memory starting at the second number and ending at the first number minus one. After running this word, the buffer will contain data from the last sector read.

13. WR - "10 wr" - writes data from the buffer into the specified sector.

14. MANY - "noop many" - repeats the preceding command until a key is struck. This is useful for running a periodic event in the DISK 3 circuitry while observing nodes with the scope probe.

15. HEX - changes the number base of the terminal I/O processor to hexadecimal. FORTH comes up in decimal.

16. DECIMAL - changes the number base to decimal.

COMPUTER ENCLOSURE 2

There are two enclosure styles available, a desk top and a rack mount. Other than the mounting hardware they`re identical. The ENCLOSURE 2 comes with 20 slot fully terminated and motherboards available. All the motherboards are also identical except for the number of board slots. The constant voltage power supply has a circuit breaker on the A.C. line and is fully fused on the outputs. It is fused at 25 Amps on the 10 volt output, and 3 Amps each on the 16 volt outputs. It will operate well with a varying line voltage, but must have 60 Hz. No provision has been made for 50 Hz operation though a different transformer could be installed for this purpose.

PRODUCTION TESTS

There is only one check made on the ENCLOSURE 2. This is a check of the the bus voltages. Facing front of the empty enclosure, the two farthest left bus pins in any slot (pins 1 and 51) should be 10 volts. The second from the left in front (pin 2) should be +16 volts. The second from the left in back should be -16 volts. The two farthest right pins (pins 50 and 100) should be ground, as are pins 20 and 70. All the rest should be at a termination voltage of 2.7 volts. This can be adjusted with the trimpot at the back of the motherboard. If this termination voltage is off it may produce erratic operation of the computer. All pins should be checked for proper voltages. The system boards can be installed at this time and the system booted.

COMMON ENCLOSURE 2 PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
No boot	No voltage on bus	Check fuses
	No voltage from supply	Check rectifier and supply circuit
	Two bus pins shorted	Check each connector, a in may be touching the one opposite
Erratic operation	Bad termination voltage	Adjust R6, check U2 on motherboard
Blows fuses	Pins 2 and 52 sorting	Check all connectors
	Board not inserted correctly	Check all boards

WH 4/1/83
Board Products Section

M-DRIVE/H

The M-Drive/H consists of 512K of dynamic RAM, and an LSI refresh controller. An 8203 refreshes the 64K x 1 DRAM and arbitrates access to them.

The starting address is loaded into a series of counters on the board. Data is then transferred in sequence without the need to send a new address for each byte. This speeds up transfers considerably.

The M-Drive/H looks like two I/O ports to the system. It takes up no memory space. One I/O port is used to load the address into the counters. The other is used to read and write data.

PRODUCTION TESTS

There is one test of the M-Drive/H in production. There are several portions to this test, which test both the RAM array and the support circuitry.

MDR: This test requires no special hardware . The first part of the test is the BIT STUCK HIGH test. This sends an 00 Hex to each address on the board, then reads it back out . If anything but 00 comes back an error message is given, saying the address, which bit was wrong, and the exact location of the bad chip.

Next is the BIT STUCK LOW test. This sends out an FF Hex to each address on the board, and reads it back. The error messages is the same, giving the location of the problem and the nature of the error.

The CHECKERBOARD test sends a 55 Hex out to each location and reads it back. Again the message gives location and nature of the problem. This test is helpful in determining if two data lines are shorted together.

The ADDRESS LINE TEST tests each address line individually. A 55 Hex is sent to address 0, and verified. 55 is then sent to address 2 then to 4,8,16 Hex on to address 800000 Hex. The address is simply incremented one address line at a time. The contents of the rest of the board are checked each time the address is incremented to see if an address line is shorted to any other line.

Finally there is the RANDOM WRITE/READ TEST. This test takes the code of the test and duplicates it over and over until the board is full. This tests pattern sensitivity of the board.

REPAIR TESTS

There are two very short tests used in the repair of the M-Drive/H. These are MDRV and MDRVIN. These tests simply allow access to any given address with any byte of data. With the use of DDT86, the desired address and data can be loaded into the routine and executed. MDRV writes the data to the address and MDRVIN reads the data back from that location. These can be stopped and single stepped through for ease in tracing the problem.

An example of how DDT could be used to step through a short routine follows (underline indicates user entry):

```
B>DDT86 MDRV.CMD
 DDT86 1.1

   START          END
CS 1C00:0000  1C00:011F
  -S100
  1C00:0100 B1
  1C00:0101 00 05       (Extended address)
  1C00:0102 BB
  1C00:0103 00 AA       (Low order address)
  1C00:0104 00 55       (High order address)
  1C00:0105 B4
  1C00:0106 55 AA
  1C00:0107 .
  G100
```

First the value of the extended address is changed, then the low order, then the high order are changed. The data can also be changed to whatever data you wish. This is a continuing loop that loads the same address over and over with the same data.

Stepping through this routine enables you to see the address being loaded in to the counters one byte at a time and shifted to higher levels until the entire address is loaded. It also enables you to see the data loaded in. This makes it very easy to detect a problem with either data or address lines. When using MDRVIN the address is inserted as before. The data does not need to be reloaded into the routine. The data that was loaded into memory with MDRV should still be there when read back out using MDRVIN.

COMMON M-DRIVE/H PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
Board is not recognized in boot M-Drive/H memory size message	Bad memory chip	Check board with MDR test
Board does not exist in MDR	Board is not being selected	Check board select U6 pin 3 (high active)
More M-Drive exists than expected	Board is being selected for more than one board location	Check for extra board select for all eight board locations
Fails row 0 column 0	Bad control signal	Check outputs from U6 PAL chip
Fails whole row	No RAS	Check output from U10
	Bad address line	Use MDRV and MDRVIN to check address lines
Fails different locations	Bad data buffer	Use MDRV and MDRVIN to check data lines

CPU 8085/88

The CPU 8085/88 is a system processor board with two microprocessors, 8085 and 8088. Both processors share much of the same S100 bus interface circuitry (i.e. control, status, address, and data drivers) and only one processor can be on-line at a time. The 8085 currently runs at a speed of 6 MHz (using the 8085AH-1 chip) and can be switched for 2 MHz operation; the 8088 currently runs at 8 MHz only (using the 8088-2 chip). The 8085 is an 8-bit CPU and the 8088 is a 16-bit CPU (essentially an 8086) with an 8-bit data bus; because of its 8-bit data bus, the 8088 can easily share the data bus with the 8085.

The CPU 8085/88 is supported at CompPro by the CP/M-80, CP/M-86, CPM 8-16, and MP/M 8-16 operating systems.

The support hardware for the two CPUs can be divided into separate functional blocks for discussion, the first of which is the SWAP CIRCUITRY. This collection of flip-flops and gates carries out an orderly transfer of control of the bus from one CPU to the other. The sequence of steps is as follows:

1) processor A is on-line, processor B is off-line (because its hold signal is activated)

2) processor A executes the assembly language command "IN xx" where "xx" is a hex byte which corresponds to a pre-arranged address (the standard address for CompuPro software is FD)

3) an octal comparator compares the "xx" to a switch setting and, if they match, issues a pulse which, when "ANDed" with control and status signals that identify an "IN" instruction, causes a flip-flop configured as a one-shot to pulse

4) this momentary pulse initiates circuitry which deactivates the HOLD line on processor B and activates the HOLD line on processor A

The circuitry which implements step four must respond to a PHOLD* signal from DMA devices on the S100 bus, delaying it during a processor swap until the swap is over, and guiding the PHOLD* to the on-line processor.

The PROCESSOR SWAP circuitry can optionally reset the newly on-line processor by the setting of switches 1-4 and 1-5. A one-shot controlled by an RC circuit responds to state of the processor hold acknowledge signals and drives the reset to the appropriate processor if the switch settings allow.

Failures in the PROCESSOR SWAP circuitry can usually be traced to one of the processors not responding to a hold request quickly enough, so

swapping out the processors is usually a good first step in trouble-shooting such a problem. After ruling out the processors as cause for a swap failure, make sure that the "portpulse" signal is being generated at U2 pin 5 while running the SWAP5 program (see production tests). If there is no "portpulse", check the decoder/comparator circuitry; if there is a "portpulse" check the flip-flops (U9 and U10) and associated gates which generate the actual hold signals.

A second block of the CPU support hardware is the MEMORY MANAGEMENT circuitry which is used to selectively assert extended address bits A16 to A23 under program control. To do this, an octal latch, driven by the CPU data lines, drives its output onto the extended address lines; the latch is clocked by gates and a comparator which recognize an "OUT xx" comand, where "xx" is a hex byte matching the setting of switch 3. (Switch 3 and the comparator are also used by the SWAP circuitry). The octal latch drives the contents of the accumulator onto the extended address lines. Since the 8088 can already drive A0-A19, provision is made through a multiplexor (U29) to only "manage" address bits A20-A23 when the 8088 is on-line.

Troubleshooting this circuitry is best done while running the BASIC pogram, T5 (see production tests below). If the test shows that the extended address lines are not being asserted at all, check out the decoder/comparator circuitry; if the extended address lines are being asserted, but they are being asserted incorrectly, check the registers that actually drive the extended address lines.

Another separate block of support circuitry is the POWER ON JUMP feature; this allows the processor to begin fetching instructions from any address on a 256 byte boundary in the first 64K page of memory when power is turned on and, optionally, when a reset occurs. To accomplish this, a flip-flop which detects the power-on-clear or reset signal enables a POWER ON JUMP latch and disables the regular processor data-in latch. The POWER ON JUMP latch forces a three-byte jump instruction (according to setting of switch 2) onto the processor data lines and then is disabled. The processor jumps to the specified address, the regular data-in latch is enabled, and the processor begins fetching instructions from the address to which it has just jumped.

Failures in this circuit can be traced out by simply enabling a POWER ON JUMP and single-stepping through the first few bus-cycles after a power on, examining the state of the data latches (U38 and U26) and power on detect flip-flop (U5) after each step. If the POWER ON JUMP fails but the POWER ON RESET works well, check whether the power on capacitor (C9) and resistors (R13, 14 and 15) are intact.

The next block of circuitry to discuss includes the clock circuitry for the processors and S100 bus. The 8085 is driven by a simple 12 MHz crystal or by a 4 MHz signal produced by a TTL-gated RC tank circuit; switch four selects which of these signals is passed through to the clock inputs of the processor. The 8088 requires a clock

signal which maintains 66/33 % duty cycle. An Intel 8284 IC, driven by a 24 MHz crystal, is used to provide this. Since the PHI signal for the S100 bus and the system clock for the board must be driven by the same clock used for the on-line processor, a flip-flop which "remembers" which processor is on-line is used to enable either the 8085 clock or the 8088 clock onto the PHI and system clock lines. Finally the 2 MHz S100 CLOCK signal is driven by a flip-flop which divides the output of the 4 MHz tank circuit mentioned above.

Failures in the 8085 clock can usually be traced to bad crystals or a faulty switch 4; the failure of an 8088 clock is usually traceable to a bad 8284 IC.

The last block of circuitry to be discussed can be referred to as the BUS STATUS MACHINE. This is a PROM (U30) whose address lines are driven by the status lines of the 8085 or the 8088 (depending on which one is on-line), and whose output drives (via an octal latch) the S100 status lines. The PROM essentially "decodes" the status of the on-line processor and drives the S100 status lines accordingly. Failure of this circuit can usually be traced to the PROM itself or to the tri-state buffer (U40) that gates the 8085 status lines to the PROM address lines.

Most circuitry of the 8085/88 board not discussed above falls under the category of S100 BUS INTERFACER. This circuitry includes gates to guide an interrupt to the on-line processor, jumpers to connect S100 vectored interrupts to one of the three restart inputs of the 8085, and gates to assert a wait signal at the on-line processor depending on the state of S100 signals XRDY and PRDY, and switch 1-3. In addition, a socket is provided which allows connecting the data lines of an IMSAI Front Panel to data input lines of the on-line processor. Failure of this circuit can usually be traced to the PROM itself or to the tri-state buffer (U40) which gates the 8085 status lines to the PROM inputs.

PRODUCTION TEST

The production test for the CPU 8085/88 involves running programs under two operating systems. First, CP/M80 is booted and a test of the memory manager, a test of the swap circuitry, and a BASIC program are run.

The memory manager test, NEWT5, is an assembly language program which asserts the extended address lines one-by-one and then reads in the actual state of those lines using an Interfacer II parallel port and a modified cable connected directly to the S100 bus. Running of this test requires that the test system's memory be set to global.

The swap circuitry test, SWAP5, is also an assembly language program. It uses the 8085 to initialize two counters (one for the 8085 and one for the 8088), it then swaps to the 8088, uses the 8088 to decrement its own counter, swaps back to the 8085, uses the 8085 to decrement

its respective counter, and compares the two counters (assuring that the 8088 did come on-line and successfully accomplish its instructions). This sequence is repeated 256 times, thus requiring the CPU to perform numerous rapid swaps without error.

The BASIC test (run under the MBASIC interpreter), BASTST, is a simple program that uses a loop to decrement a counter and outputs the value of that counter during each iteration of the loop. This test serves as an intensive processor exercise.

Upon completion of the above tests, CP/M86 is booted and a BASIC86 test, TST, is run. TST does essentially what the above BASTST does: repeatedly outputting to the terminal under control of a loop structure; again, this is an intensive exrecise of the processor.

LAB TESTS

The lab tests used for troubleshooting CPU's which have failed in production are almost the same as the production tests; the main difference is that the memory manager test used is not NEWT5, but T5, a program written in BASIC. T5 asserts the extended address lines just as NEWT5 does, but instead of using a parallel port and special cable to check the results, a JADE probe, which shows the state of the S100 bus with LED's, is viewed by the technician to assure that the proper sequence of extended address bits has been asserted.

Production tests can be single stepped under software control (using DDT commands such as trace) or hardware control (using IMSAI Front Panel or a JADE probe modified to single-step) in the technician's lab, allowing isolation of the cause of failures to specific portions of programs.

COMMON CPU 8085/88 PROBLEMS

SYMPTOM	PROBABLE CAUSE	REMEDY
Will not boot up	Bad clock	Check Xtals, 8284 (U19), and switch 4
	Bad voltages	Check inputs and outputs of voltage regulators
	Bad reset capacitor (C9)	Check JADE probe to see if board is being continually reset, bus is locked up
	Processor held in wait state	Check U17 and pull-up resistors on its inputs
	Processor interrupt asserted	Check U6, U32, and U13 interrupt gates
	Address latch not enabled	Check ALE and SYSALE at U23 (pins 1,2,3)
Boots CP/M 80, but won't boot CP/M 86	Switch circuitry is faulty	Check the HOLD signal on both CPUs, make sure the correct one is held, and that only one is held; trace back through flip-flops (U9 & U10)
	Bad Xtal X2	Check X2 and associated circuitry
	Bad CPU 8088	Change CPU chip
	Bad swap decoder circuitry	Check switch 3 setting and condition, (U34) comparator, and "Portpulse" flip-flop and gates

SYMPTOM	PROBABLE CAUSE	REMEDY
Will not boot MP/M*	Switch circuitry is faulty (does it boot CP/M 86 ?)	Check HOLD signal on both CPUs, trace back through flip-flops and "Portpulse" circuitry
	Bad Xtal X2	Check X2 and associated circuitry
	Bad processor	Replace 8085 or 8088
Does a partial boot	Bad Prom G-165 U30	Change Prom G-165
Works at 2MHz will not work at 5MHz	Bad paddle switch S4	Check switch
	Bad Xtal X3	Check X3

27 JULY 84

TRAINING MANUAL FOR CPU 8086/87

The CPU 8086/87 is a 16-bit processor card based on the Intel 8086 microprocessor, the 80130 interrupt timer/controller, and, optionally, the 8087 floating point math processor. The basic CPU is available in an 8 Mhz (8086-1) or 10 Mhz (8086-2) version; however, installation of the optional 8087 requires that the processor speed be cut down to 5 Mhz because both the 8086 and the 8087 share the same clock and, at this time, the fastest 8087 available from Intel is the 5 Mhz version.

The 8086 chip can be set to run in one of two modes, MIN or MAX, depending on the system environment in which the processor will be used. Setting the processor to MIN facilitates its use in a simple, single processor environment; setting the processor to MAX facilitates its use in a multiprocessor environment. The 8086 on the CompuPro CPU 8086/87 is set to the MAX mode.

80130

The 86/87 comes equipped with the INTEL 80130, an LSI chip which processes interrupts according to a programmable priority and timing scheme. It is available from Intel with 16K-bytes of internal ROM, but the one made available from the CompuPro factory is a ROMless version. However, provision is made on the 86/87 CPU card to decode addressing for the ROM space if a ROM-version 80130 is acquired.

The 80130 is programmed through a 16-byte block of ports, and decoding of the address space for these ports is also provided for on the 86/87 card. Both the 80130 ROM and I/O space decoding circuits use digital comparators which compare dip switch settings to address lines driven by the processor. The 86/87 card also provides buffered input to the interrupt pins of the 80130 from the S100 vectored interrupt lines as well as jumper posts for simple connection to the vectored interrupts from the appropriate outputs of the 80130.

CompuPro software does not make use of the 80130. The interrupt driven MP/M-86 relies solely on the SYSTEM SUPPORT I to process interrupts, and CompuPro recommends that the 80130 be removed from the board when running MP/M-86. However, the 80130 is tested in production at the factory (see PRODUCTION TESTS below).

If your system is using the 80130 but it fails to respond to vectored interrupts, it should first be determined whether the 80130 itself is operational and that the switch settings enabling and defining its I/O port space are correct. If the 80130 and its switch settings are in order, a good place to start looking for problems is at the I/O chip select (IOCS*) signal at U24 pin 2; if a program that attempts to access the 16-byte block of ports does not pulse this chip select, then the octal comparator (U9)

or the exclusive-or gates (U8 and 10) that comprise the port address decoder circuit should be examined. If there is an IOCS* pulse, then assure that the vectored interrupts are actually being brought to the 80130 from the bus.

BYTE SERIAL TRANSFER CIRCUIT

The 86/87 card has a BYTE SERIAL TRANSFER CIRCUIT. This circuit fulfills the S100 requirement that 16-bit CPUs be able to perform word transfers in one bus cycle (with 16-bit memory) or two cycles (with 8-bit memory). Since the 8086 processor only knows about word transfers, when a word is to be transferred to or from an 8-bit slave the 8086 will generate the strobes for the even byte ("thinking" that it has actually created the strobes for a word transfer) and then it will wait while the TRANSFER SEQUENCER circuit creates the strobe for the second, odd byte. In this discussion the TRANSFER CIRCUIT circuit is understood to include the actual buffers that drive the S100 data bus as well as their timing and control components.

Most of the TRANSFER CIRCUIT circuit can be found on pages two and three of the schematic. Page two shows the bidirectional buffers and registers that allow the upper byte of a word to be transferred to or from memory either by ganging together the S100 DATA OUT and DATA IN lines in the case of a 16-bit slave, or passing both the high and low bytes through the appropriate IN or OUT lines in the case of an 8-bit slave. This array of buffers is immediately controlled by the outputs of the PROM G193 (U50); the inputs to this PROM are driven by both the TRANSFER circuitry and other control circuitry on the card (including the POWER ON JUMP and STATUS DECODER circuits).

Page three shows both the circuitry which checks the 16-bit acknowledge line (SIXTN*) generated by the slave, and the sequencer which generates the control strobes to accomplish the transfer of the second byte of a word while the processor waits.

Past problems with this circuitry have been noted when a 1.5 K ohm pull-up resistor was used to pull-up the SIXTN* signal; that value of resistance was not sourcing enough current to ensure that the SXTN* signal would rise back up quickly enough after the slave device had pulled it down, resulting in an unreliable SXTN* detector. The problem was corrected by replacing the 1.5K ohm resistor (R8) with a 560 ohm resistor, but there are likely to be some older boards in the field with the original value resistor.

Another problem related to this circuit was found when the 86/87 was run with early RAM21 memory cards: the rise time on the SXTN* line generated by the early RAM21s was again not quick enough. The problem was solved by modifying the RAM21 circuitry (see notes in the RAM21 training manual) so problems with the 86/87 BYTE SERIAL TRANSFER CIRCUIT should be investigated only after assuring that the problem is not due to an unmodified, old RAM21 in the system.

The first thing to investigate when having problems with this circuitry is PROM G193, it can be quickly swapped out for a

known good part and it directly controls the actual data buffers.

Next, determine whether the signals ONECYCLE (U28 pin 3) and TWOCYCLE (U15 pin 6) are being generated correctly according to the state of the SXTN* signal; the presence of SXTN* during a transfer (as when running with RAM 21s or RAM 22s) should generate a ONECYCLE, while the lack of SXTN* during a transfer (as when running with RAM 17s) should generate a TWOCYCLE.

If those signals are correct, check outputs of the flip-flops (U12,U13, and U14) and their clock inputs. These flip-flops act as a shift register and time the sequence of a two byte transfer.

CONTROL STROBE GENERATOR

The next block to discuss can be called the CONTROL STROBE GENERATOR. This circuitry is built around the INTEL 8288 bus controller chip (U36) which monitors status signals (S0,S1, and S2) and the processor clock to produce the appropriate read, write, address latch, and interrupt acknowledge signals for the internal bus cycles. These internal signals are then translated to S100 bus control signals by other circuitry.

A subset of the circuitry which accomplishes this translation to S100 signals is the READ/WRITE STROBE GENERATOR. This circuit, found mostly on page six of the schematic, monitors the read and write outputs of the 8288 chip, the outputs of the WORD/BYTE TRANSFER CIRCUIT, the CPU card's internal clock, and the signal ALE (address latch enable). From these signals, it derives the unbuffered S100 signals pDBIN, pWR*, pSYNC and pSTVAL*. These signals are then buffered onto the S100 bus by U37 (page four of the schematic). A problem was noted in this circuitry on boards of revision "B" and earlier: the flip-flop (U15) that clocked the ALE signal through to help create the strobes was sometimes found not to respond quickly enough. The solution was implemented in the "B1" revision by adding two inverters (U27) in line between ALE at U28 pin 8 and the D-input of flip-flop U15b; the gates add sufficient propagation delay to relieve the close timing demands made on flip-flop U15a.

In case of problems with control strobes, an easy first step is to swap out the 8288 chip because so much of the other control circuitry depends on its proper operation. After that, tracing back from the unbuffered control signals (at the top of page six) is fairly straight-forward, though keep in mind that the WORD/BYTE TRANSFER circuit must be operating correctly and the internal clock must be reliable.

HOLD REQUEST TRANSLATOR

The next block to discuss is the HOLD REQUEST TRANSLATOR. As was mentioned above, the 8086 on this CPU card is set to the MAX mode; consequently, bus request and bus grant signals which are employed to transfer control of the bus to a requesting DMA device are communicated to the processor via a single bi-directional pin, the RQ/GT0 pin. The MAX-mode 8086 recognizes a DMA bus request by a low-going pulse on the RQ/GT0 pin, grants

control of the bus by a subsequent low-going pulse on the same pin, and resumes control of the bus only after a third and final low-going pulse on the same pin.

On the S100 bus, however, a DMA bus request is made by asserting the HOLD* signal, the bus is granted by asserting the pHLDA (processor hold acknowledge) signal, and the bus request is terminated by dropping of the HOLD* signal. So the CPU circuitry must translate the S100 HOLD* signal to a momentary pulse on the 8086 RQ/GT0 line, translate the resultant grant pulse from the processor to the S100 HLDA signal, and translate the dropping of HOLD* to a third, terminating pulse to the 8086.

The circuitry that accomplishes this is found on page four of the schematic; the signal RQGT* is the key point at which the 8086 RQ/GT0 pin connects to the translation circuitry. Note that when an 8087 math processor is installed in the system, all requests to, and grants from, the processor are passed through the 8087 via its RQ/GT1 pin; when there is no 8087, jumper J8 (page one of schematic) must be installed to connect the RQGT* signal directly to the 8086.

The most commonly noted problem with this circuitry is when a HOLD* signal is translated to the processor as a hold request, but the subsequent grant pulse from the processor is not translated to a HLDA signal. The system will usually do a single head-load (a blinking of the drive light and a click from the disk) and hang without booting; examination of the bus with a JADE probe usually reveals that the HOLD* signal and all DMA arbitration lines are asserted (for the Fh priority of the DISK1) but that HLDA is inactive. Usually, such a problem can be traced to the failures of the gates on chips U5, U11, and U44.

Note that examination of the RQGT* line with a logic probe can cause enough capacitive loading to alter the circuit's performance; so after determining that the open-collector inverter (U44) is reliable, avoid probing this circuit directly at the RQGT* signal.

STATUS GENERATOR

The STATUS GENERATOR monitors processor status lines, address latch timing, and wait state select switches. It decodes them with a PROM, and creates the appropriate S100 status lines.

The heart of this circuit (and the part most prone to failure) is PROM G192 (U53); when the S100 status signals are incorrect, an easy first step is to exchange this PROM for a known good one. Other problems have been traced to faults in octal buffer U52 or quad latch U51.

MEMORY MANAGER

The MEMORY MANAGER expands the addressing capability of the 8086 from 20 bits (1 Mbyte) to 24 bits (16 Mbytes). Much like the memory management schemes on the CompuPro CPU 85/88 and CPU Z, it allows the CPU card to latch any four bits to address lines A20 to A23 under program control using an I/O instruction.

An I/O write to a 16 bit address whose lower byte is FD and

whose upper byte matches the setting of SWITCH 4 paddles 3-10 will latch the four most significant bits of the word written to the outputs of quad latch U46. Upon the transition of the signal Address Latch Enable during the next bus cycle, the four bits latched at U46 will be latched to the S100 address lines A20 through A23 by part of octal latch U45.

Note that this memory manager requires the use of the 8086's 16-bit I/O instruction: OUT DX,rr, not its 8 bit I/O instruction. (For important points about the actual implementation of such a memory management scheme, see page nine of the 8086 technical manual.)

Failures of this circuit fall under two categories: failure to properly decode the I/O instruction that latches the data, or failure to latch the proper data. The latter is more common and almost always traces to bad latches or bus lines at U44 or U45. The failure to decode the actual instruction can often be traced to bad contacts at SWITCH 4 (switching them on and off repeatedly should remedy this), or a faulty octal comparator (U22).

To troubleshoot the circuit, run the MEMMNGR program (see PRODUCTION TESTS below) and observe the upper address lines on the JADE probe. If those address lines are being activated, but in the wrong sequence or pattern, then examine U45 and U46 and related data lines. If there is no action at all on those upper address lines (and if you are running the program under the guidelines explained below in the section on PRODUCTION TESTS), rerun the program and see if you are getting a pulse at U23 pin 6. If no pulse, check for low going pulses at IOWC* (U23 pin 5), the EO output of the comparator (U22 pin 19) and the EI input of the comparator (U22 pin 1). The source of these pulses can be easily traced on page five of the schematic to either the 8288 bus controller or the lower address byte latches.

POWER-ON-JUMP CIRCUIT

The POWER-ON-JUMP circuit, is much like its counterparts in the CompuPro 85/88 CPU and Z80 CPU. When enabled, it causes an assembly language jump instruction with a jump address to be forced onto the CPU's data bus as the CPU is fetching its first op code and operands. The jump address will be XX000 hex where "XX" is a hex byte corresponding to the setting of SWITCH 5. The CPU can optionally be caused to jump on reset by setting SWITCH 5-9, and the whole circuit can be disabled by setting SWITCH 5-10.

This circuitry is found at the bottom half of page six of the schematic: inverting octal buffers (U17 and U31) drive the CPU data bus, while their inputs are driven either by inverters or pull-up resistors depending on the setting of SWITCH 5 and the condition of the address lines. Gates (U23 and U38) and a flip-flop (U54a) are used to detect a power-on-clear signal and the processor read strobe and to enable the octal buffers during the first five bus cycles after power on or reset.

The most common problem with this circuit, as with the

POWER-ON-JUMP circuits of the 85/88 or Z80 CPUs, is when the power-on-clear capacitor (C8, page 7 of schematic) is faulty or the wrong value. A value less than the specified 18 uF may not create a long enough power-on-clear pulse.

If the capacitor is the correct value and power-on-clear remains low briefly after power on, then check at pins 1 and 19 of U17 and U31 to see if the octal buffers are being enabled at power up when power-on-jump is enabled. If not, then determine whether power-on-clear is causing U38 pin 8 to go low and clear flip-flop U54a. If the octal buffers are being enabled, but the actual jump is not being accomplished, single step through the first six bus cycles of a power-on-jump attempt and examine the CPU data bus to see that the proper sequence of words are being placed on it. If the sequence is incorrect, assure that the inputs to the buffers are correctly driven by the inverters and gates that monitor the address lines, and also assure that the power on jump signal (U54 pin 5) is reaching PROM G193 (U50 pin 1) and causing the regular data bus drivers to be disabled during a power-on-jump.

POWER-ON-CLEAR AND RESET CIRCUIT

The POWER-ON-CLEAR AND RESET CIRCUIT creates internal and external clear signals when detecting a power-on or a reset signal. The circuit can be found on the top of page seven of the schematic.

Problems with either the passive components or inverter/buffers can cause power-on-clear to rise too quickly (or to not rise at all) or can cause the reset signal to be inactive (or always active). Troubleshooting is straightforward: observe the bus with a JADE probe to determine that POC* (power-on-clear), SLAVE CLR*, and RESET* are being generated when power is first turned ON, and that SLAVE CLR* and RESET* are asserted when the system reset button is pushed. Problems with these signals usually trace back to a faulty buffer or inverter (U27,39,42, or 55).

CLOCK GENERATORS

The final block of circuitry includes the CLOCK GENERATORS for the internal clock and the S100 signals PHI and CLK. The internal clock (which drives the CPU and other CPU components) and PHI are generated by the 8284A integrated circuit (U40) from INTEL, which is in turn driven by a tank circuit built around crystal X1.

Also incorporated into this circuit are the signals which cause wait-states (S100 signals RDY and XRDY and internal signal WAIT*). The 8284A IC monitors the gated result of these signals and, if needed, requests wait states from the processor by outputting a low on its READY signal.

The CLK signal generator is a tank circuit built around a 4 MHz crystal X1 whose output is divided by two and then buffered.

When there are problems with the internal clock or PHI,

first investigate the tank circuit around X1 for a clean and correct frequency at U56 pin 12; problems here usually trace to capacitors C9 and C10 or the crystal itself. If the tank circuit looks O.K. and the PHI or internal clock is still bad, replace the 8284A with a known good chip.

If the processor is being held in a wait state (READY from the 8284A is always low), replace the 8284A; if still in a constant wait state, determine whether AND gates U28 and U38 are operating correctly.

PRODUCTION TESTS

The production tests for the 8086/87 CPU include a memory manager test, called MEMMNGR; a test of the 80130, called 80130; a simple test of the 8087, called 8087; a more demanding test of the 8087, called BENCH87, and a BASIC program, called TST.

MEMMNGR

The MEMMNGR test, a program written in the "C" language, asserts the extended address bits via the memory manager in a prearranged pattern. Then it reads and checks the address lines through a modified cable and I/O2 parallel port, and outputs a message to the terminal indicating success or failure to set the extended address bits in the correct pattern. The test requires global memory so that the setting of the extended address lines will not actually cause the processor to leave the page that contains the running program. As the program is now written, it can only be run on CP/M-86 operating systems of revision 1.0-J or earlier.

80130

The 80130 test is an assembly language program which initializes the 80130 IC, sets up vectored routines, and then asserts vectored interrupts on the S100 bus. The vectored routines output to the terminal the number of the vectored interrupt asserted.

The actual assertion of interrupts is done by outputting to an I/O2 parallel port which is connected by a modified cable

directly to the S100 vectored interrupt lines. The operator of the test observes the terminal screen to assure that the correct pattern of numbers (0 through 7) are being displayed, indicating that the 80130 is responding correctly to program control. The 80130 test expects an I/01 or I/02 serial port addressed at 00H, and, of course, the proper switch settings on the processor board to enable I/O ports of the 80130 IC for programming (specifically SWITCH 3-2 on). Also, the program will not operate correctly with a SYSTEM SUPPORT in the bus because the interrupt processing circuitry on the SYSTEM SUPPORT interferes with the 80130.

8087

The 8087 test is also an assembly language program. It simply "talks" to the control registers of the 8087, and then checks if they have responded at all. If there is an 8087 IC installed on the CPU, the program will output to the screen this message: "THE 8087 IS ALIVE"; if there is no 8087 installed, the program will output the message: "NO 8087 HERE".

BENCH87

A more thorough test of the 8087 is another assembly language program BENCH87 which creates a specified number of differential equations at random and then solves them using floating point instructions which take advantage of the 8087. If successful, this program will ouput the resulting answers of the equations, if not successful the program will hang.

TST

The last test is a BASIC program called TST which outputs a number to the screen, increments it, and outputs it again, continuing to a preset limit. TST gives the whole CPU a thorough workout, requiring it to complete fairly intensive disk and memory operations.

LAB TESTS

The lab tests for the 8086/87 are essentially the same as those used in production except for the following differences.

In the MEMMNGR test, the extended address bits are visually observed with a JADE probe in the lab as opposed to being read off the bus via a modified cable and parallel port; this allows the technician to actually observe where the test fails (e.g. which bits are not getting set or whether any bits are being set at all).

In the 80130 test, the vectored interupts are asserted manually by touching a grounded lead to the S100 vectored interrupt pins instead of writing out to a parallel port whose cable connects to those interrupts; this allows the technician to assert any pattern of interrupts at any pace.

TROUBLESHOOTING CHART

SYMPTOM	PROBABLE CAUSE	REMEDY
On reset, system boots, signs on, but won't input from terminal	I/O wait-state machine failure	check SW 3-3 and 3-4 U53,6,28,38, & 40
On reset, does one head load at disk and then hangs in HOLD state	HOLD translation circuitry not responding correctly	check U4,5,6,11 & 44 (page 4 of schematic
On reset, one head-load at disk, then caught in loop	bad data buffer(s) or control circuit	check U47,48,49,16,5(29 & 35 (page 2 of schematic
Boots with RAM17, but fails boot after one head-load with RAM16 or RAM21	two cycle sequencer caught in two-cycle mode	check U24,28,35 & 15
Fails power-on-jump	bad power-on-clear signal	check power-on-clear circuit (page 7 of schematic)

31 JULY 84

CPU 68K TRAINING MANUAL

The CompuPro CPU 68K is based on the Motorola 68000, a microprocessor with 32 bit internal registers and a 16 bit data bus. Its architecture is similar to that of some minicomputers, with eight data registers, eight address registers, an asynchronous bus, and a mostly orthogonal instruction set. It has 24 address bits and can directly address sixteen megabytes of memory. An optional memory management unit (MMU), the 68451, can be used to increase that address space to 32 megabytes.

HARDWARE CONFIGURATIONS

The CPU 68K is available from CompuPro in the System 816 E package. It is also available separately in a variety of configurations:

1. A&T 8 MHz processor
2. A&T 8 MHz processor with MMU
3. CSC 10 MHz processor
4. CSC 10 MHz processor with MMU

The clock speed of a CPU 68K card depends on three things: the microprocessor chip, the oscillator crystal, and the oscillator capacitors. The table below identifies the correct values for those components in relation to clock speed.

CPU 68K CLOCK SPEED RELATIVE COMPONENTS

COMPONENT	8 MHz value	10 MHz value
CRYSTAL	16 MHz	20 MHz
CAPACITORS	150 pF	120pF
mPROCESSOR	68000L8	68000L10

Note that the CPU 68K can be jumpered for operation at half speed (jumper J1 - see CPU 68K technical manual).

Future CPU 68K cards may have provisions for running with the Motorola 68010 microprocessor - essentially an improved

version of the 68000.

SOFTWARE

The main CPU 68K software package available from CompuPro is the operating system CP/M 68K from Digital Research. CP/M 68K is much like CP/M 86 in its user interface. It comes with an assembler and a C language compiler.

The C compiler is of special importance because of the relative lack of applications software for CP/M 68K. C is considered one of the most portable high level languages, and there are already many application programs written in C for other computers.

Included with CP/M 68K in the CompuPro system package is the MAPFORTH operating system and FORTH compiler. MAPFORTH is a complete operating system that runs independently of CP/M 68K. It includes a complete assembler. (Note that the MAPFORTH assembler expects reverse Polish code and, as such, does not deal with conventional assembly language source code.

Some Digital research programming languages are becoming available now, including a BASIC compiler. Pascal and FORTRAN complilers are expected from Digital Research in the future. They will probably be available through CompuPro as they are received and approved.

Note that no software available from CompuPro supports the MMU. Any development work done with the MMU will receive minimal technical support from CompuPro.

HARDWARE

The CPU 68K circuitry is explained in detail in the Technical manual supplied with it. This overview of the hardware concentrates on parts of the board that are most important when troubleshooting or installing a CPU 68K system.

MEMORY MANAGEMENT UNIT

Memory management is a broad term encompassing many different concepts relating to organization of memory resources. A simple form is the segregation of memory into separate areas for different functions: program intructions in one area and program data in another, user (application) programs in one area and supervisor (operating system) programs in another.

More complex memory management involves the protection of certain areas of memory from unauthorized access.

Even more complex memory management involves virtual memory. This is the use of hardware, software and auxiliary storage to make a relatively small amount of main memory seem to a programmer to be much larger than it really is.

The 68451 chip is a memory management device manufactured by

Motorola for use with the 68000. With a considerable amount of software development, a 68000 computer equipped with the 68451 could accomplish the less complex tasks of memory management well, and the more complex tasks (virtual memory in particular) satisfactorily. (Replacing the 68000 with a 68010 would lend a better hardware environment for virtual memory.)

CompuPro offers no software support for the MMU, so an MMU-equipped board is useful only to a software developer. The MMU in the CPU 68K as shipped from CompuPro is reset to a passive state and has no effect on system memory organization.

The actual function of the MMU board is to translate logical address lines from the processor, to physical address lines for actual memory. While translating the address lines, it delays the processor address strobe for two clock states, whether it is in its passive mode or not. So an MMU equipped board takes two extra clock states to complete every bus cycle.

Installation of the MMU requires a special PAL chip, instead of PAL 184P-3B, PAL 184P-3C must be used. The installation should be performed at the factory to ensure proper test.

ROM

The CPU 68K has two sockets for onboard ROM which are mapped into 64K memory page FDh, program space only. The "program space only" requirement means that only instructions and immediate data can be fetched from the ROM. Attempts to fetch indirectly addressed data (such as constants in a table) are not allowed by the decoding scheme in the circuitry. Contact the Technical Support department at CompuPro for details on the availability of a custom PAL which allows accessing all types of data from the ROM.

The ROM sockets are organized in a pair to accommodate the use of commonly available 8-bit memory parts: 2716s for 2K words of ROM, 2732s for 4K words, and 2764s for 8K words.

INTERRUPTS

Interrupts for the 68000 fall into two basic categories: hardware controlled (autovector), or software controlled. In the autovector mode, the 68000 responds to an interrupt by interpreting its priority and vectoring under hardware control to a location predetermined by the designers of the chip. The autovector mode did not work correctly in early revisions of the CPU 68K (revision "D" and earlier). The current "F" revision of the board operates correctly in the autovector mode. Revision "D" boards can be upgraded, revision "C" boards cannot. Contact the Technical Support department for details.

In the software controlled interrupt mode, the 68000 expects an intelligent interrupt controller to provide a vector byte on the data bus which, when multiplied by four, points to the

appropriate jump address for handling that interrupt. This type of interrupt scheme has been successfully implemented for the CPU 68K using the 8259s on the SYSTEM SUPPORT I. Although CompuPro at this time has no software support for developers trying to write code to support such a scheme, some examples of succesful code techniques can be obtained by contacting the Technical Support department.

The revision "D" CPU 68K circuitry correctly supports software controlled interrupts, but the artwork of the "F" revision introduced a change in the circuit that may cause problems in processing such interrupts. Again, contact the Technical Support department for details on a modification for the "F" revision to accommodate software controlled interrupts.

PROGRAMMABLE ARRAY LOGIC

The CPU 68K makes great use of programmable array logic (PAL) chips to decrease the overall chip count for the board and to make its design more flexible. There are four PALs in all, and each one serves a distinct purpose. PAL 184P-1B (U15) synchronizes the timing of byte serial transfers or word transfers for 8- or 16-bit memory.

PAL 184P-2A (U31) decodes addressing for devices or functions with reserved places in the 16 Mbyte memory map: the ROM in page FDh, the MMU in page FEh, and I/O space in page FFh.

PAL 184P-3B directly controls the bidirectional data buffers under timing control from PAL 184P-1B. It also synchronizes the S100 wait state signals RDY and XRDY with the system clock. This PAL must be replaced with PAL 184P-3C when an MMU is installed.

PAL 184P-4A translates CPU 68K strobes to S100 strobes.

SCHEMATIC GUIDE

The schematic in the CPU 68K technical manual can be broken into different circuits according to their function:

1. CPU and MMU - page 1
2. ROM sockets - page 2
3. S100 status generator - page 3, top
4. power-on-jump - page 3, bottom
5. ROM address switching - page 3, center left
6. wait state/DTACK generator - page 4, top
7. HOLD to BUS REQUEST translator - page 4, center
8. vectored interrupts circuit - page 4, bottom
9. data bus buffers - page 5, top
10. clock generators - page 6
11. power-on-clear circuit - page 6, center
12. address buffers - page 2

TEST SOFTWARE

PRODUCTION TEST

The production test for the CPU 68K is very simple. It confirms that the CPU can run CP/M 68K and work with floppy disk and M DRIVE. The procedure is as follows.

1. Boot CP/M 68K

2. Type "TST".

"TST" is a submit file which does a directory of floppy drive A:, PIPs two C language programs to drive M:, and runs those programs from drive M:. The two C programs are ALPHA and FPRIME. ALPHA outputs the upper and lower case alphabet to the screen, and FPRIME finds and displays all prime numbers from one to 1000.

A failure in TST should result in a CP/M error message such as "BDOS ERROR", "VERIFY ERROR", OR "EXCEPTION ## AT ADDRESS XXXX".

The hardware required to run the production test is a basic CP/M 68K system with M DRIVE.

LAB TEST

The lab test begins with a run of the production test software. It also includes tests for proper autovector operation and, when needed, proper MMU operation.

The autovector test, 68KINT, is an assembly language program which initializes the autovector table, enables interrupts, and enters a loop which repeatedly outputs the exclamation point character (!) to the terminal. When an interrupt is asserted, the CPU should vector to a routine which outputs the number of vectored interrupt.

To run the test, the technician boots CP/M 68K and types "68KINT". Then he asserts interrupts by manually grounding the vectored interrupt lines on an extender card with an alligator clip lead. CPU 68K switch 1-1 must be on for autovector operation, and with standard jumpering the CPU will only process vectored interrupts 0 through 5 and NMI.

To test an MMU-equipped CPU, you need a basic MAPFORTH system (basic CPU 68K system with I/O console port at address 10H -Interface 1,2,3, or 4). To run the test:

1. Boot MAPFORTH

2. Type "235 LOAD". This loads the MMU test program. Successful load should result in the message "OK" being displayed at the screen.

3. Type "MMU-TST". This causes an MMU data construct called

a descriptor to be loaded into the 8-byte MMU accumulator and displayed at the screen. The screen should show a left hand column labeled "AC0" through "AC7" and a right hand column of hexadecimal bytes. The bytes can be any value except for all 00s or all FFs.

4. Strike another key. This causes the next decriptor to be loaded and displayed. Check again for hex bytes that are not all 00s or FFs, and also that the bytes are different from the bytes in the last descriptor.

5. Repeat step 4 until the last descriptor (31) has been loaded and displayed.

This test, while not a sophisticated workout of the MMU, ensures that it is alive and responding correctly to attempts to access its registers.

DIAGNOSTIC TEST GLOSSARY

CLOCK.COM: Tests clock circuitry on the SYSTEM SUPPORT 1

CTEST.COM: Tests the CENTRONICS port on the INTERFACER 4

DIPTST.COM: Test dip switch S1 on INTERFACER 3

DISK2.COM: Formats the hard disk and prints bad sector map

DISK2.DOC: Explains the features of DISK2.COM

DMAEND.COM: Terminates the DMATST

DMATST.COM: Test the DMA circuitry on the DISK 1

D2.COM: Forth program used to test the SELECTOR CHANNEL and DISK 2

FORTH.BLK: Block program associated with D2 and MALIGN

I3TEST.COM: Tests the serial ports and interrupts on the INTERFACER 3

I4TEST.COM: Tests the serial ports and the interrupt on the INTERFACER 4

MDR.CMD: Test the M-DRIVE/H

MDRV.CMD: Writes to one address on the M-DRIVE/H

MDRVIN.CMD: Reads from one address on the M-DRIVE/H

MEMMNGR.CMD: Tests the memory manager on the CPU 8086

MEMTEST.COM: Tests memory boards and memory on CPU/Z and SYSTEM SUPPORT 1

SWAP5.COM: Tests ability of CPU 8085/88 to swap processors

TIMER.COM Tests timers on the SYSTEM SUPPORT 1

T5.BAS: Test the extended address lines with the CPU/Z

ZINT1.COM: Tests the interrupt lines with the CPU/Z

68KINT.68K: The interrupt circuitry on the CPU 68K

8087: Test for the presence of an 8087 on the 8086

8259A: Tests the interrupt controller and timers on the SYSTEM SUPPORT 1

9511: Tests the math processor on the SYSTEM SUPPORT 1

9512: Tests the 9512 math chip on the SYSTEM SUPPORT 1

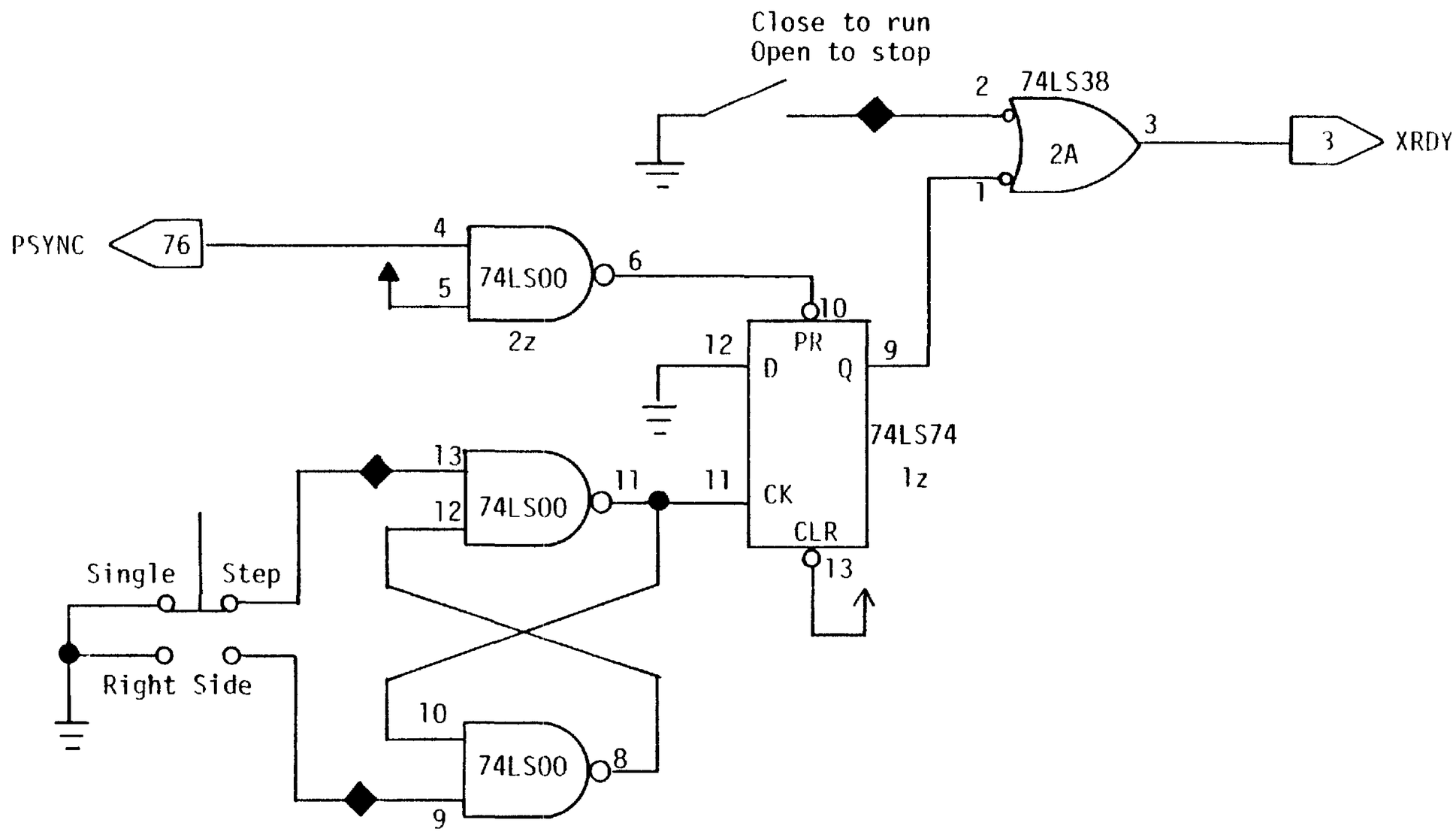
80130: Tests the interrupt controller on the CPU 8086

RECOMMENDED EQUIPMENT FOR SYSTEM CENTER LABS

The following list represents the minimum set of equipment needed by a Full Service System Center's technical service labs personal to perform the necessary assembly and field testing of COMPUPRO Systems.

1. Dual trace oscilloscope. Minimum bandwidth 30 MHz. (We use Hitachi V-1050F 100 MHz).
2. Digital volt ohm meter.
3. Jade bus probe (with modification for single stepping).
4. Mullen Extender Board with logic probe (with 60 ohm 15 watt power resistors on power lines to limit current).
5. RS-232 breakout box.
6. COMPUPRO system with terminal.
7. Soldering iron.
8. Desoldering tool.
9. Assorted small hand tools.
10. Small gauge wire such as wire wrap wire.
11. Wire wrap pencil with stripper.
12. 6" by 16" piece of Plexiglass to cover enclosure power supply.
13. High grade alcohol for cleaning edge connector (without lanolin).
14. Freeze spray.
15. Compressed air.
16. Good lighting.

SINGLE STEPPER FOR JADE BUS PROBE



This is a list of the I/O ports used by all COMPUPRO products.

PORT	PRODUCT
00 through 03	INTERFACER 1 and 2
10 through 17	INTERFACER 3 and 4
50 through 5f	SYSTEM SUPPORT
90	DISK 3
C0 through C3	DISK 1
C4 and C5	RESERVED
C6 and C7	MDRIVE/H
C8	DISK 2
F0	SELECTOR CHANNEL
F1	MPX
F2 through F5	RESERVED
FD	MEMORY MANAGER and SWAP PORT ON DUAL PROCESSOR
FFF0 through FFFF	80130 on CPU 8086/87