FLUKE

TROUBLESHOOTER

NO. 2, MAY 1983 FOR MICRO-SYSTEM TROUBLESHOOTER USERS

Training

Since the last issue of the *TROUBLESHOOTER*, the Advanced Training schedule has been completed through the first week of May, 1984, to give everyone plenty of time for planning to attend a seminar. The new seminar dates are listed below. Contact your local sales office or Representative for prices and registration details.

Remember, we recommend that anyone attending the Advanced Training seminar first attend a free Introductory Training class. Contact your local Fluke Sales Office or Representative for information on the Introductory Training seminars.

Advanced Training Schedule

Minneapolis, MN	Jun 21-22 '83
Watford, England	Jul 5-6 '83
Ottawa, Canada	Jul 12-13 '83
Denver, CO	Jul 26-27 '83
Sunnyvale, CA	Aug 23-24 '83
Watford, England	Sep 6-7 '83
Burbank, CA	Sep 13-14 '83
San Diego, CA	Sep 27-28 '83
Austin, TX	Oct 11-12 '83
Atlanta, GA	Nov 15-16 '83
Orlando, FL	Dec 6-7 '83
Baltimore, MD	Jan 17-18 '84
Paramus, NJ	Feb 14-15 '84
Burlington, MA	Feb 28-29 '84
Canada	Mar 13-14 '84
Call (416) 678-1500 for more info.	
Detroit, MI	Apr 3-4 '84
Rolling Meadows, IL	Apr 17-18 '84
Minneapolis, MN	May 1-2 '84

Application support

The following individuals and organizations have indicated their capability of, and interest in, providing independent 9010A support. The services offered are shown with each name.

Mr. Ali Mosleh
Computer Sciences Corp.
8300 Merrifield Ave.
Fairfax, VA 22031
(703) 560-5051 (office)
(703) 560-1316 (home)
Contract programming and troubleshooting.

Mr. Gary Aiken, Eng. Manager
Diversified Data Corp.
6551 Loisdale Court
Springfield, VA 22150
(703) 922-9444
Contract programming, engineering support, integrated logistics support, documentation, and training.

Mr. Dick Thomas
General Electric Co.
Instrument & Computer Equipment Repair Service
5096 Peachtree Rd.
Chamblee, GA 30341
(404) 452-4905
Contract programming, testing, and troubleshooting.

Mr. Quint Pierson
General Electric Co.
Schenectady Instrument Service
Bldg. 28, Rm. 503
Schenectady, NY 12345
(518) 385-5107
Contract programming, testing, and troubleshooting.

Mr. John Schira
Quinton Instruments
2121 Terry Ave.
Seattle, WA 98121
(206) 223-7373
Contract programming.

These individuals or organizations have contacted us and asked that we pass their names along to our 9010A customers. If you would like your name added to the list, please let us know.

RAM SHORT or RAM LONG... which, when, and why?

To meet the needs of a wide variety of users, the 9010A contains two Random Access Memory (RAM) testing programs. RAM SHORT provides a fast test for finding shorted or open bus lines and gross failure of RAM chips. RAM LONG provides a complete checkout of all RAM circuits and components.

The RAM tests on the 9010A do not need specific knowledge about what type of RAM is in the Unit Under Test (UUT). The tests work equally well with static or dynamic RAMs from all manufacturers. Since RAM accesses occur at the speed of the microprocessor in the UUT, the dynamic characteristics of the RAM are exercised, revealing faults that might otherwise go unnoticed.

RAM SHORT performs these tests: (1) a read/write test (to check the basic functioning of the RAM), (2) a tied-data-lines test (to locate shorted or open data lines), and (3) a fast decoder test (to locate shorted or open address lines).

RAM LONG provides (1) a read/write test, (2) a tied data lines test, (3) a thorough decoder test, and (4) a pattern sensitivity test (to find subtle, intermittent RAM cell failures.

The tests are performed in order. If any test fails no further tests are performed, because subsequent test results cannot be trusted if previous errors have not been corrected.

The read/write test is performed by both RAM SHORT and RAM LONG. This test insures that each bit at every RAM address can store both 1 and 0. At each address in the tested range, a zero is written to the RAM, and then the location is read to check that all bits are zeroes. A word of all 1's is then written to the location and read back to check that all bits are 1's. If there are any bits that cannot be driven to both states, a diagnostic message is printed.

The tied-data-lines test also is performed by both RAM tests. This test checks that data lines to the RAM chips are not tied together. The test is performed at the *first* address in the tested range. Each data line is tested by being written to the opposite state from all other data lines. Then the data is read back. For example, on an 8-bit data bus, the following patterns would be written:

```
00000001 }
         } data bit 0 tested
11111110 }

00000010 }
         } data bit 1 tested
11111101 }

   - -     -
   - -     -
   - -     -

01000000 }
         } data bit 6 tested
10111111 }
```

The last data bit (7) does not need to be tested since it would already have been reported as tied to some other bit. If a data line is tied to the tested line, one of the lines will be pulled into a state opposite to the one in which it was written. For example:

		Data Bit 3 Test
Data Written	00001000	11110111
Data Read	00001000	10110111

bits tied

In this example, data line 6 was forced low when data line 3 went low, indicating that data lines 3 and 6 are tied together.

RAM SHORT contains a fast decoder test. A pattern of data bits is written at each address in the range to be tested. This pattern contains information about the address at which it was written. Next, each location is read back to see if the data there is what was originally written. If the data read back matches the pattern that was written, everything is fine. If the data read back differs from the written pattern, it is because two locations in RAM have become connected or "aliased" to each other. The typical cause of this is failure of the address-decoder circuitry — an address line either shorted or open, or a failure of internal RAM decoding logic. The faulty address line is identified in a diagnostic message. Its functioning can be quickly verified with the probe.

RAM LONG uses a different decoder test which tests each address location in the following way: The current location is first zeroed, and then a word of all 1's is written to each location which differs from the current location by a single address bit. After each of these locations is written, the current location is read to check that all data bits are still 0's. If any bit has changed to a 1, the current location is aliased to the test location in that bit. As above, address-aliasing is caused by an open or shorted address line or internal failure in the RAM chip.

RAM LONG also contains a test for pattern sensitivity. A RAM cell is pattern sensitive if writing a certain data pattern at one address causes a change in data at another address. Pattern sensitivity may be caused by manufacturing defects in the RAM chip, operation at temperature extremes, or marginal circuit design. Repair of a pattern-sensitivity failure generally involves replacing a RAM chip. In the pattern-sensitivity test, the RAM is written with a sequence of data patterns that guarantees to drive any two bits in the tested address range to opposite states at some time during the test. Thus, if writing to one cell causes a change in any other cell in the RAM, the pattern-sensitivity test will report this fact.

RAM SHORT detects all failures of external address-decoder circuitry and finds data lines that are stuck or tied together. It also finds the majority of RAM chip failures, because these failures typically have obvious symptoms. There are some failures that RAM SHORT is not able to detect. If RAM SHORT passes, and the technician still suspects a RAM failure, then RAM LONG should be run.

The length of time RAM SHORT takes to run is directly proportional to the number of words or bytes of memory tested. This means that running RAM SHORT on 16K bytes of RAM takes just 16 times as long as running RAM SHORT on 1K bytes of RAM. RAM LONG, on the other hand, takes time proportional to the number of words tested times the

number of patterns it writes, which increases with the size of the address range tested. This means that it takes 22 times as long to test 16K bytes of RAM as it does to test 1K bytes and over 100 times as long to test 64K bytes as it does to test 1K bytes.

It is very unlikely that a pattern-sensitivity failure will occur in two RAM locations that are on separate chips. This means that if you have a 32K-byte block of RAM that is composed of four 8K-byte banks, running RAM LONG on the four banks separately is faster and just as effective as running it on a single bank of 32K bytes.

Both RAM tests perform the tied-data-lines test only once, at the first address in the tested range. However, if the RAM being tested consists of several banks of RAM chips, tied data lines on some banks will not be diagnosed correctly. For example, consider a system with two banks of RAM, where all data lines are OK to the first bank of RAM, but one line is open to the second bank. A single RAM test over the entire address range will not correctly diagnose the stuck data bit, because the tied-data-lines test is performed only at the first address of the first bank of RAM. The problem will, instead, be incorrectly diagnosed as a read/write error in that data bit. A RAM test over the second bank alone would correctly diagnose the problem as a stuck (open) data line. If a read/write failure is reported at the beginning of a data bank, the test should be repeated for that bank alone.

RAM LONG may be the most economical use of your time if you have a board that is failing due to a suspected failure of RAM chips. RAM SHORT is more appropriate for troubleshooting completely dead boards or for production testing.

New μP/pods test large RAMs 20 times faster

The 68000, 8086, and 8088 interface pods have built-in Quick Memory Tests to allow the user to test the potentially large memory areas much faster. In addition to greatly reducing test time, the Quick Memory Tests also provide: (a) a choice of byte or word tests for the 16-bit microprocessors (μPs), (b) a choice of address increment size, (c) automatic checking for inactive data bits in the Quick ROM Test, and (d) more flexibility under program control. Each of these capabilities is discussed in detail below, followed by a short explanation of how to use the tests, and a comparison with the normal troubleshooter tests.

The speed was increased by putting more intelligence into the interface pod, eliminating the need for communication between the pod and the troubleshooter on every transaction. This reduces the RAM test time by a factor of about 20.

The combined choice of byte or word addressing and of address increment size gives the user more flexibility. For a RAM test, he can specify byte addressing with an address increment of 1, thus testing every byte individually and checking the ability of the UUT to do single-byte writes without altering the adjacent byte. This mode also causes more activity during the RAM test, increasing the chance of detecting pattern-sensitivity errors. For a ROM test, he can specify byte addressing and an address increment of 2, allowing separate checksums for individual ROMs on either the high or low byte of data. The address increment of 2 specifies every other byte, thereby testing either all odd or all even addresses, depending on the value of the starting address.

The Quick ROM Test will look to see that every data line is valid both high and low sometime during the test. If not, it will set an error flag and allow the user to see a mask of which line, or lines, were inactive. An inactive data line doesn't always indicate an error, but when an incorrect checksum is received, this flag and mask will be strong evidence of a shorted line or a bad data buffer.

The added flexibility under program control is a result of the way these tests are implemented. Because the Quick Memory Tests are contained in the interface pod and not in the mainframe, they cannot be accessed directly by the normal troubleshooter test keys. Instead they are initiated, controlled, and examined by reads and writes to "special" addresses (outside the μP's address space). When the pod receives an instruction to read or write at one of these special addresses, it is interpreted as a special function command. Thus all commands and examinations are just a series of reads or writes as far as the troubleshooter is concerned. This allows full control and decision-making while under program control.

Following this paragraph is a short explanation of these tests. For more details, look in the Operating Information section of the appropriate interface pod instruction manual, under Special Functions. If you haven't read the Operating Information section yet, we recommend that you read the entire section. It contains other worthwhile information unique to operating your particular interface pod.

Two commands are required to initiate a Quick Memory Test: the first one to specify the starting address of the block to be tested, and the second one to specify the ending address and the address increment. The format of each command is as follows:

WRITE @ t aaaaaa = x y , where

t = test code (2 for Quick RAM Test, 3 for Quick ROM Test)

aaaaaa = beginning or ending 6 digit address

x = address increment (only when specifying the *ending* address. 0 means use default increment)

y = address type (0 for starting address, 1 for ending address)

The complete address is prefixed with a test code (t) to specify whether RAM or ROM is to be tested. The write data specifies the address type (starting or ending). When specifying the ending address, the higher order digit of the write data

specifies the address increment to be used. If the write data is 01 (or just 1), the default address increment specified in the pod manual is used. The pod initiates the test upon receiving the ending address, so the starting address must be specified first.

NOTE: With some microprocessors (e.g., 8086, 68000) the address specification determines the data size (word or byte). Refer to the appropriate Interface Pod Instruction Manual for details.

For example, to perform a Quick RAM Test on an 8086 pod over the address range of 0 to FFFF (hex) using *byte* addresses and an *address increment* of 1, execute the following two instructions:

WRITE @ 2800000 = 0
WRITE @ 280FFFF = 11

The first instruction specifies the starting address and the second specifies the ending address and the increment, and begins the test.

To see the progress or results of the test during or after execution, press READ, then ENTER. In general, Ax means test Aborted, Bx means Busy (test not finished), Cx means test Complete (no errors), and Fx means test Failed. These codes are explained in more detail in the manual. If the test failed, or you're just curious, related data is available at other special addresses as described in the manual. This data includes the specified starting and ending addresses and the status code. The ROM test also makes available the checksum of the last block tested and a hex mask of any inactive bits. The RAM Test data includes the failure address, the expected data at that address, the actual data read, and a hex mask of bad data bits.

The Quick RAM Test will find any errors that RAM SHORT will find. It will also allow testing the RAM space of a 16-bit μP one byte at a time, which RAM SHORT/LONG won't. It is a two-pass RAM test. The first pass (pass 0) tests the read/writability of every bit, both high and low. A status code of F0 means the test has failed the first pass and has found a RAM location which has a read/write problem. The second pass (pass 1) reads every location again, looking for the unique data last written into that location. A failure of this pass usually means an address-aliasing problem, but could also be caused by noise, refresh, or pattern-sensitivity problems. This pass will find *all* hard address-aliasing problems, but it is not as good as RAM LONG for detecting or identifying soft errors (i.e., intermittent, pattern-sensitive, etc.).

The Quick ROM Test provides a checksum of the block tested. Though not as complete a test as the CRC signature of the troubleshooter, it is sufficient to detect most ROM errors.

Because of the extra address encoding and having to read at special addresses to see the results, the Quick Memory Tests are a little harder to use in Immediate Mode than the standard troubleshooter tests. But programs can be written to make execution of these tests as easy as executing standard tests. When large blocks are tested, the greatly enhanced speed will justify the small increase in operator difficulty.

Now Available

Pod Adapter Packaging Kit

For builders of adapters for 9000-Series interface pods, Fluke is now offering the Pod Adapter Packaging Kit. The kit is helpful in overcoming the mechanical difficulties of constructing a pod adapter and provides all of the parts necessary for (1) housing the adapter circuitry, (2) connecting the pod to the adapter, and (3) connecting the adapter to the UUT. The user must provide his own adapter design and the necessary electronic parts. The kit contains the following parts:

— blank printed circuit board for mounting adapter components
— 40-pin ZIF socket
— 40-pin ribbon cable for connecting the adapter to the UUT (for 40-pin sockets only)
— "small pod" case (includes the hardware necessary for mounting the circuit board)
— blank decal for labeling the adapter
— technical note on Pod Adapters (Fluke Technical Publication B0156)

The U.S. price of the Pod Adapter Packaging Kit is $195. Order model number 9000A-200. Order one kit for each adapter.

Special Offer of New Interface Pod UUT Cables

Fluke is now using a twisted-pair transmission-line cable for the link between the 9000-Series Interface Pod and the Unit Under Test (UUT). The new high-performance cables offer greater shielding and reduce the possible effects of system noise in the μP system. Also, the new cables are up to 12 inches long.

Although all of the Fluke 9000-Series Interface Pods are currently being shipped with the new shielded cables, we realize that many of you have only the older UUT cables. To give everyone the benefits of the new cables, we are offering the new cables at a reduced cost for a 5-month period. From May 1 to September 30, you can order replacement UUT cables for the Interface Pods listed below at a U.S. cost of only $65 (a 40% savings).

Contact your local Fluke Sales Office or Representative to place your order.

Pod	Part No.	L	Pod	Part No.	L
6502	685479	12"	Z80	685461	12"
6800	685479	12"	8080	685487	12"
6802	685479	12"	8085	685495	7"

Utility Programs Introduction

The Utility Programs tape contains seven programs to further expand the capabilities of your Troubleshooter. The programs are:

Merge Tape — to selectively move and renumber programs
Register Add — to do unsigned 32-bit addition of 9010A register contents
Register Subtract — to do unsigned 32-bit subtraction of 9010A register contents
Frequency Counter — to take frequency readings with the 9010A probe (10 Hz to 6 MHz)
Setup — to change *some* of the parameters in the Setup Menu while under program control
Probe Pulser — to change the probe pulser mode while under program control

The U.S. price of the 9000A-910 Utility Programs is $95. Contact your local Fluke Sales Office or Representative to order your copy.

A pod adapter for the NSC800

Although Fluke does not manufacture an NSC800 Interface Pod, 9000-Series products can still be used to test and troubleshoot NSC800-based micro-systems. This example shows how an existing interface pod can be adapted (through the use of circuitry external to the pod) to an NSC800-based system. Even if you don't have to repair NSC800-based systems, this example might give you some general ideas about adapting existing 9000-Series Interface Pods to your particular micro-system.

The NSC800 features a bus structure and a signal set that are almost identical to those of the 8085 microprocessor (μP). The read/write cycle timing characteristics of the NSC800 are also very similar to those of the 8085. These similarities in bus structure and cycle timing make the 9000A-8085 Interface Pod the logical choice for adapting to an NSC800-based Unit Under Test (UUT).

The circuitry required for an adapter will depend on the requirements of the particular UUT involved. For example, if the NSC800 UUT does not use a certain signal line, the circuitry used to adapt that line to the 8085 pod will not be necessary. As you read through the example, you should refer to the timing diagram of Figure 1. Refer also to the schematic diagrams of Figures 2, 3, and 4.

NOTE: The NSC800 is a CMOS μP, which allows it to be operated at power supply voltages other than +5V. The 8085 pod (and the troubleshooter probe) operate at TTL signal levels. If the NSC800 UUT uses a power supply voltage other than +5V, level-shifting circuitry will have to be included for every line (except ground) connected between the UUT and the 8085 pod. This circuitry is not shown in the schematic diagrams of this example. If buffer circuitry is necessary, the ability of the pod to check for drivability errors may be affected.

The Clock Circuitry. The clock requirements for the NSC800 and the 8085 pod are basically the same: both μPs can be driven directly by a crystal or by a UUT-generated clock signal. The minimum and maximum operating frequencies of the NSC800 fall within those of the 8085 pod, so no clock division or multiplication circuitry is required.

Both μPs generate a single-phase clock output signal. A quick comparison of the clock signals for the pod (shown in Figure 1) and the NSC800 shows that the 8085 pod CLK output signal must be inverted for use on the NSC800 UUT. This will allow the 8085 pod bus signals to be synchronized with those on the UUT. If the CLK signal is not used on the UUT, no inverter is necessary.

NOTE: If a crystal is used on the NSC800 UUT for generating clock signals, the length of the 8085 pod cable lines (and the adapter cable lines) may make it necessary to include a clock oscillator in the adapter circuitry. For more information regarding the generation of 8085 Interface Pod clock signals, refer to the Fluke Technical Data bulletin titled "Guide to 8085 Microprocessor-Based System Testing." Contact your local Fluke Sales Office or Representative for a copy. Ask for Application Information bulletin B0151.

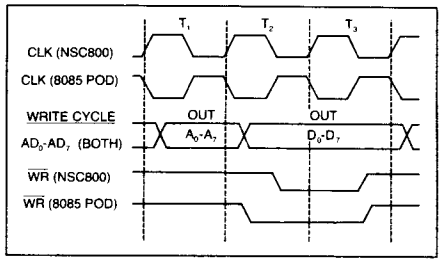

Figure 1. Timing differences between NSC800 and 8085 signals.

The Address and Data Buses. The address and data bus structures of the 8085 pod and the NSC800 are basically identical. Both multiplex the low-order address bits with the data bits. The timing of the multiplexing with relation to the clock signal is the same. The Address Latch Enable (ALE) signals are also identical. Therefore, no adapter circuitry is required for the address/data buses or the ALE signal.

Read ($\overline{RD}$) and Write ($\overline{WR}$) Control Signals. The $\overline{RD}$ control signal of the NSC800 is nearly identical to that of the 8085 pod. However, Figure 1 shows that the $\overline{WR}$ control signals *are* different. The $\overline{WR}$ pulse of the NSC800 begins one-half clock period (T-state) *later* than the $\overline{WR}$ pulse of the 8085 pod. The trailing edge of each write pulse occurs at the same point in the cycle.

The leading edge of the NSC800 $\overline{WR}$ pulse occurs when the data on the bus is guaranteed to be valid. This is *not* true of the 8085 pod $\overline{WR}$ pulse. If the NSC800 UUT only latches data on the trailing edge of the $\overline{WR}$ pulse, the above mentioned timing difference is not important. In that case simply connect the $\overline{WR}$ pin of the 8085 pod to the $\overline{WR}$ line of the UUT. However, if the UUT performs some special action on the leading edge of the $\overline{WR}$ pulse, the 8085 pod $\overline{WR}$ pulse may have to be modified such that the leading edge occurs when valid data is on the bus.

The $\overline{WR}$ pulse of the 8085 pod can be adjusted to begin at almost the same point as the $\overline{WR}$ pulse of the NSC800. Note from the NSC800 timing diagram that the NSC800 $\overline{WR}$ pulse has a duration of about one T-state. Thus a rising-edge-triggered D flip-flop can be used to delay the start of the 8085 pod $\overline{WR}$ pulse by one-half T-state. The CLK output of the 8085 pod is used to clock the flip-flop. The circuit necessary for this function is shown in Figure 2.

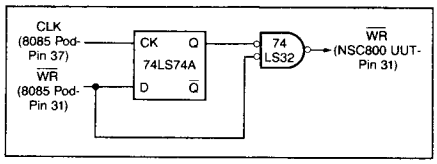

Figure 2. Circuit for adapting the 8085 Pod $\overline{WR}$ Signal to an NSC800 UUT.

Note that connecting the flip-flop between the 8085 pod and the NSC800 UUT defeats the pod's ability to check for drivability errors on the $\overline{WR}$ control line. For this reason, the circuitry of Figure 2 should be used only if the UUT requires it.

I/O Reads and Writes. The 8085 pod and the NSC800 feature identical input/output (I/O) bus structures. However, there is a difference in the timing of I/O operations. The NSC800 automatically extends the I/O read or write cycle by inserting a wait state. The 8085 does not have this characteristic; wait states must be generated by external circuitry. If the NSC800 UUT requires this extra wait state, the adapter will need to include the circuitry necessary for generation of an 8085 wait state. Figure 3 shows an example of a wait state generation circuit for the 8085 pod. The wait state will only be generated during an I/O operation. Note that the circuit of Figure 3 also allows wait states to be generated by the UUT through the use of the NSC800 $\overline{WAIT}$ line.

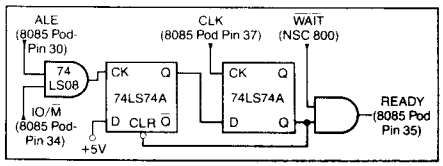

Figure 3. Wait state generation circuit for an 8085 Pod/NSC800 adapter.

If the UUT does not require the extended I/O cycle of the NSC800, simply connect the $\overline{WAIT}$ line of the NSC800 directly to the READY pin of the 8085 pod. (Refer to Figure 4.)

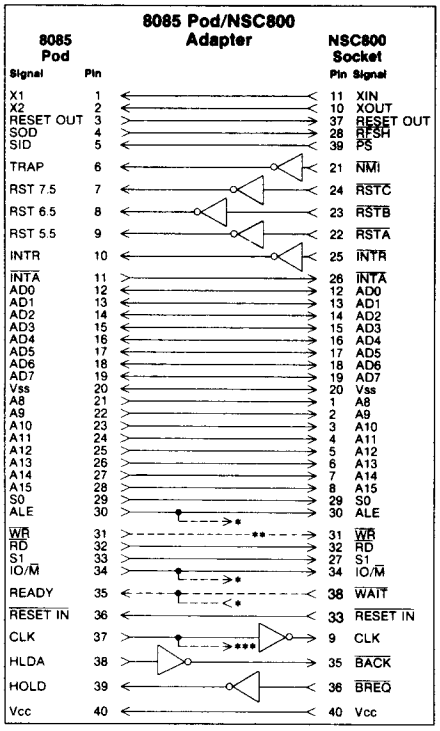

Figure 4. Adapter Wiring.

*See text and figure 3
**See text and figure 2
***See text and figure 2 and 3

Dynamic RAM Refresh Signals. The NSC800 μP features on-chip dynamic-RAM refresh control circuitry, but the 8085 pod does not support dynamic-RAM refresh. Few NSC800 UUTs use this capability, but if yours does, you might have to design refresh control circuitry into the adapter. In that case, the 8202 Dynamic RAM Controller should be considered for use in the adapter, as it is designed to be compatible with 8085 μP characteristics.

The Refresh ($\overline{RFSH}$) control signal of the NSC800 is used to alert circuits on the UUT that a refresh operation is taking place. If your NSC800 UUT uses the $\overline{RFSH}$ line, you may wish to use the 8085 pod to activate this line for testing purposes. To do this, connect the Serial Output Data (SOD) pin on the 8085 pod to the $\overline{RFSH}$ line on the UUT.

Connecting the $\overline{RFSH}$ line to the SOD line is useful only for testing the drivability of the $\overline{RFSH}$ line, using the WRITE CTL and BUS TEST functions of the troubleshooter. During normal troubleshooter operation, however, the $\overline{RFSH}$ line should be tied to Vcc, if it is to be held active.

The Power Save Line (NSC800). The Power Save ($\overline{PS}$) line of the NSC800 has no counterpart on the 8085 pod. However, many NSC800-based UUTs activate this pin to force the NSC800 into the "Power Save" mode. To use the 8085 pod to detect an active level on this line, connect $\overline{PS}$ line on the UUT to the SID (Serial Input Data) pin on the 8085 pod. A READ STS command from the troubleshooter keyboard can then be used to detect activity on this line.

Adapting Interrupt Lines. All of the interrupt lines on the NSC800 UUT can easily be adapted to the 8085 pod. Figure 4 shows that only inverters are used to adapt these lines. If the UUT does not use an interrupt signal of the NSC800, tie the corresponding pin on the 8085 pod to its inactive (low) level.

Adapting DMA lines. There is a slight timing difference between the DMA signals of the 8085 (BACK and BREQ) and those of the NSC800 (HLDA and HOLD), but this timing difference is not critical. These lines can be adapted through the use of inverters (see Figure 4). If your UUT doesn't use the $\overline{BREQ}$ and $\overline{BACK}$ lines, tie the HOLD line of the 8085 pod to its inactive (low) level.

Constructing the Adapter. To construct the adapter, you'll want to use a Fluke Pod Adapter Packaging Kit. The kit is described in the "Now Available" section of this newsletter.

Using the Adapter with the 8085 Pod and the Troubleshooter. This adapter allows the use of virtually all of the functions of the 9000-Series Troubleshooters. Refer to the Operator Manual for instructions regarding the troubleshooter. Refer to the 8085 Interface Pod Instruction Manual for instructions about using the pod. Some points of caution are described in the paragraphs below.

The RUN UUT Mode. The RUN UUT mode of the troubleshooter cannot be used in the usual way. This is because the instruction set of the NSC800 μP is a superset of the instruction set of the 8085. Therefore the 8085 pod cannot execute NSC800 code.

In many cases, however, the RUN UUT mode of the troubleshooter can be used. 8085 instructions can be loaded into UUT RAM using the troubleshooter WRITE function. The RUN UUT mode can then be used to execute the 8085 code, starting with the initial RAM address of the code.

Status and Control Lines. The $\overline{RFSH}$, $\overline{INTA}$, RESET OUT, and $\overline{BACK}$ lines of the NSC800 UUT can all be "written to" using the WRITE CTL function of the troubleshooter. The $\overline{RSTC}$, $\overline{RSTB}$, $\overline{RSTA}$, $\overline{PS}$, $\overline{RESET\ IN}$, $\overline{INTR}$, $\overline{NMI}$ $\overline{BREQ}$, and $\overline{WAIT}$ lines of the UUT can all be "read" using the READ STS function.

Remember that those NSC800 μP input or output lines with inverters between the UUT and the pod will have *opposite* Status or Control bit values.

The Probe. The troubleshooter probe can be used normally on the NSC800 UUT — *if* the UUT signal levels are compatible. CMOS signal levels are probe-compatible if the CMOS circuitry is operated with Vcc equal to +5 volts.

For more information about pod adapters, there is a bulletin available on the subject. Contact your local Fluke Sales Office or Representative for a copy. Ask for Application Information bulletin B0156.

John Fluke Mfg. Co., Inc.
P.O. Box C9090, Everett, WA 98206
800-426-0361 (toll free) in most of U.S.A.
206-356-5400 from AK, HI, WA
206-356-5500 from other countries

Fluke (Holland) B.V.
P.O. Box 5053, 5004 EB, Tilburg, The Netherlands
Tel. (013) 673973, TELEX 52237

Printed in U.S.A. P0001B-01U8304/SE EN