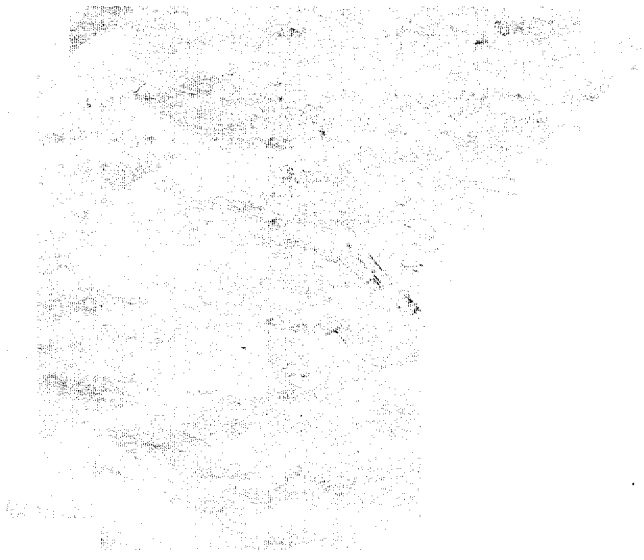
# 98032A 16-Bit Interface
# Installation and Service Manual



Hewlett-Packard Calculator Products Division
P.O. Box 301, Loveland, Colorado 80537, Tel. (303) 667-5000
(For World-wide Sales and Service Offices see back of manual.)
Copyright by Hewlett-Packard Company 1976

# Table of Contents

Figures



Tables

# Chapter 1
# General Information

## Introduction

The HP 98032A I/O Interface is a general-purpose interface which provides 16-bit data exchange between the HP 9825A Calculator and a peripheral device. The interface transfers data in a "full-duplex" mode. That is, it can have data on the output lines and be inputing data at the same time.

The hardware and software characteristics of the interface are extremely flexible. The data exchange timing and logic can be "configured", by the use of jumpers, to meet a wide variety of peripheral requirements.

The interface is shipped from the factory either as an Option or Standard Interface. The Option Interface is pre-configured for operation with a given calculator peripheral. The Standard Interface is to be configured by you, to meet your peripheral's requirements.

## Technical Specifications

| | |
|---|---|
| Data Input Lines: | 16 latched, DI0 – DI15 |
| Data Output Lines: | 16 open collector, DO0 – DO15 |
| Control Lines: | PCTL, PFLG |
| Signal Levels: | TTL and open collector |
| Temperature Range: | 0 ° to 45 °C Ambient |
| Power: | +5 Volts 300ma, obtained from the Calculator |
| Dimensions: | Approximately 16.3 × 8.9 × 3.8 cm (6.4 × 3.5 × 1.5 in) |
| Cable Length: | Standard – 4.5m (15′) |
| | Options – 2.0m (6.5′) |

# Available Options

The presently available options listed in Table 1-1, are shipped from the factory with the proper configuration jumpers and cable connector installed, as required by the listed calculator peripheral. This will allow you to connect your peripheral to the calculator without having to configure the interface. A wiring-diagram of the added connector and a diagram of the configuration jumpers are furnished with each Option Interface.

Table 1-1. Interface Options

| Interface Option | Peripheral | Factory Select Code* Setting |
|---|---|---|
| Option 062 | 9862A Plotter | 5 |
| Option 063 | 9863A Paper Tape Reader | 3 |
| Option 064 | 9864A Digitizer | 4 |
| Option 066 | 9866A/B Thermal Page Printer | 6 |
| Option 069 | 9869A Hopper Card Reader | 3 |
| Option 071 | 9871A Impact Line Printer | 6 |
| Option 083 | 9883A Paper Tape Photo Reader | 3 |
| Option 084 | 9884A Paper Tape Punch | 2 |

*See Select Code in Chapter 2.

# Hardware Description

The Standard Interface consists of three circuit boards in a case. The case plugs into any one of the calculator I/O slots. A 4.5 meter (15 feet) unterminated shielded-cable is provided for connection to your peripheral.

The logic lines available to the user are:

- Sixteen latched data input lines.
- Sixteen latched data output lines.
- Three handshake lines.
- Two peripheral control lines.
- Three peripheral status lines.
- One line for DMA (Direct Memory Access) interrupt request.
- One line to reset the peripheral.
- Shield and ground connections.

# Chapter 2
# Installation Considerations

## Introduction

The complexity of the installation procedure depends on the device to be interfaced to the calculator. If the device is one of the calculator peripherals listed in Table 1-1, and you have the correct Option Interface, the procedure is quite simple. But if the device is one of your own, you will have to "configure" the Standard Interface to meet your needs. This involves installing the correct jumpers on the Configuration Board (see Figure 2-3) and installing a connector on the end of the cable.

## Select Code

The select code should be checked for the proper setting as required by your system. The select code switch is accessible through a hole in the top of the interface case. If the interface is one of the options, the select code will be preset to the standard select code for the associated peripheral. The Standard Interface will be preset to select code 2. If it is necessary to change the setting, rotate the switch to the desired position using a small screwdriver. You should avoid using select codes reserved for the peripherals internal to the calculator. For example, the select codes shown below are used by the 9825A internal peripherals.

| Select Code | 9825A Calculator Internal Peripheral |
|---|---|
| 0 | Keyboard/Display |
| 1 | Cartridge |
| 16 | Printer |

Two interfaces should not be set to the same select code.

# Select Code Interrupt Considerations*

In the interrupt mode, the 98032A operates on one of two priority levels. Interfaces set to Select codes 0 through 7 are on the low priority interrupt level and select codes 8 through 15 are on the high priority level. Devices requiring fast interrupt service should be set to the high level. Priority within a level is in order of the select code, with 7 and 15 having the highest priority.

# Checking Interface Operations

When an interface is installed it should be checked for proper operation. To do this, read the status from the appropriate select code. For correctly installed interfaces, the returned value will be equal to or greater than 288. If the interface is not inserted correctly or if the select code setting is incorrect a zero will be returned.

# Option Interface Installation

Each Option Interface is shipped from the factory with the proper configuration jumpers and cable connector for a given calculator peripheral (see Table 1-1). To connect one of the listed peripherals to the calculator follow the instructions supplied with the peripheral. Those instructions describe how to check the interface select code and make the necessary cable connections.

# Standard Interface Installation

To install the Standard Interface with any other peripheral device it will be necessary to configure the interface to meet your needs. Configuring the interface is done by installing (soldering) the proper jumpers and if necessary a timing capacitor onto the Configuration Board (A-2). An appropriate cable connector should also be installed on the end of the cable.

In some cases, even though the configuration of the interface is very flexible, you may need to alter, or add to, the I/O logic of your device in order to make it compatible with the calculator I/O system.

*An I/O ROM with interrupt capability is required to use the interrupt mode.

# Basic Installation Steps

The following basic installation steps are given, to aid you in connecting your peripheral to the calculator via the Standard 98032A Interface. The information necessary to implement these steps follows in the next sections of this chapter.

1. Determine the type of handshake that your peripheral will require.
2. Select, if necessary, a timing capacitor.
3. Determine the logic lines that are to be used by your peripheral.
4. Install the proper configuration jumpers to obtain the proper logic (positive or negative true) and I/O handshake.
5. Connect the cable to your device or install the proper cable connector on the cable.
6. Set the select code switch to the proper position.
7. Connect the system together and check from the calculator keyboard all intended I/O operations (refer to Checking Interface Operations).
8. If there is a failure in step 7, install the optional test connector (refer to Interface Operational Test) and check similar I/O operations. If this fails refer to the Service section of this manual. If the I/O operations on the test connector operate properly, the problem is probably in the cable wiring, choice of jumpers, or your peripheral.

# Data Handshake

Synchronization of data exchanged between the calculator and a peripheral is referred to as the "handshake". The handshake is accomplished via the PCTL, PFLG and I/O lines. The peripheral receives information about the data exchange on the PCTL and I/O lines and then responds on the PFLG line. The I/O line tell the peripheral which direction the data is to be transferred, low indicates a calculator output operation, high indicates a calculator input operation.

If your peripheral does not require a handshake, connect the PFLG line to the PCTL line and then isolate these wires. Install either jumper 3 or jumper 4, but NOT both. Refer to Configuring the Logic (page 12) and to Figure 2-3.

The handshake lines, their states and intended meanings are listed in Table 2-1.

Table 2-1. Handshake Lines

| LINE | STATE OR MNEMONIC | MEANING |
|---|---|---|
| **Output** | | |
| I/O | LOW | Calculator output operation. |
| PCTL (From Interface) | CONTROL CLEAR | No new output data available. |
| | CONTROL SET | New output data is available on output line. |
| PFLG (From Peripheral) | READY | Peripheral is ready for next data transfer. |
| | BUSY | Peripheral is not ready for next data transfer. |
| **Input** | | |
| I/O | HIGH | Calculator input operation. |
| PCTL (From Interface) | CONTROL CLEAR | Calculator is not requesting new data. |
| | CONTROL SET | Calculator is requesting new data. |
| PFLG (From Peripheral) | READY | Peripheral is ready for next transfer. |
| | BUSY | Peripheral is not ready for next transfer. |

The reason that the state of the PCTL and PFLG lines are not referred to as being either "high" or "low" is that the logic level of these lines can be complemented by installing jumpers on the Configuration Board. The use of mnemonics for the state of these lines allows discussion of the handshake logic without refering to the specific jumper configuration chosen for your peripheral.

## The Two Modes of Handshake

The two modes of handshake are referred to as "full" and "pulse". The pulse mode is selected by installing jumper 6 on the configuration board. Without jumper 6 the full mode is selected.

In the full mode, which can be considered the normal mode, the calculator will check the PFLG line (from the peripheral) to ensure that it is at the ready state before setting PCTL (control set) to initiate another data transfer.

In the pulse mode of handshake the calculator will not check the state of the PFLG line before setting the PCTL line to control set (refer to Figure 2-2). Applications which require this mode do not have true ready/busy levels on their PFLG line. Only a transition on the PFLG line is used to terminate the transfer.

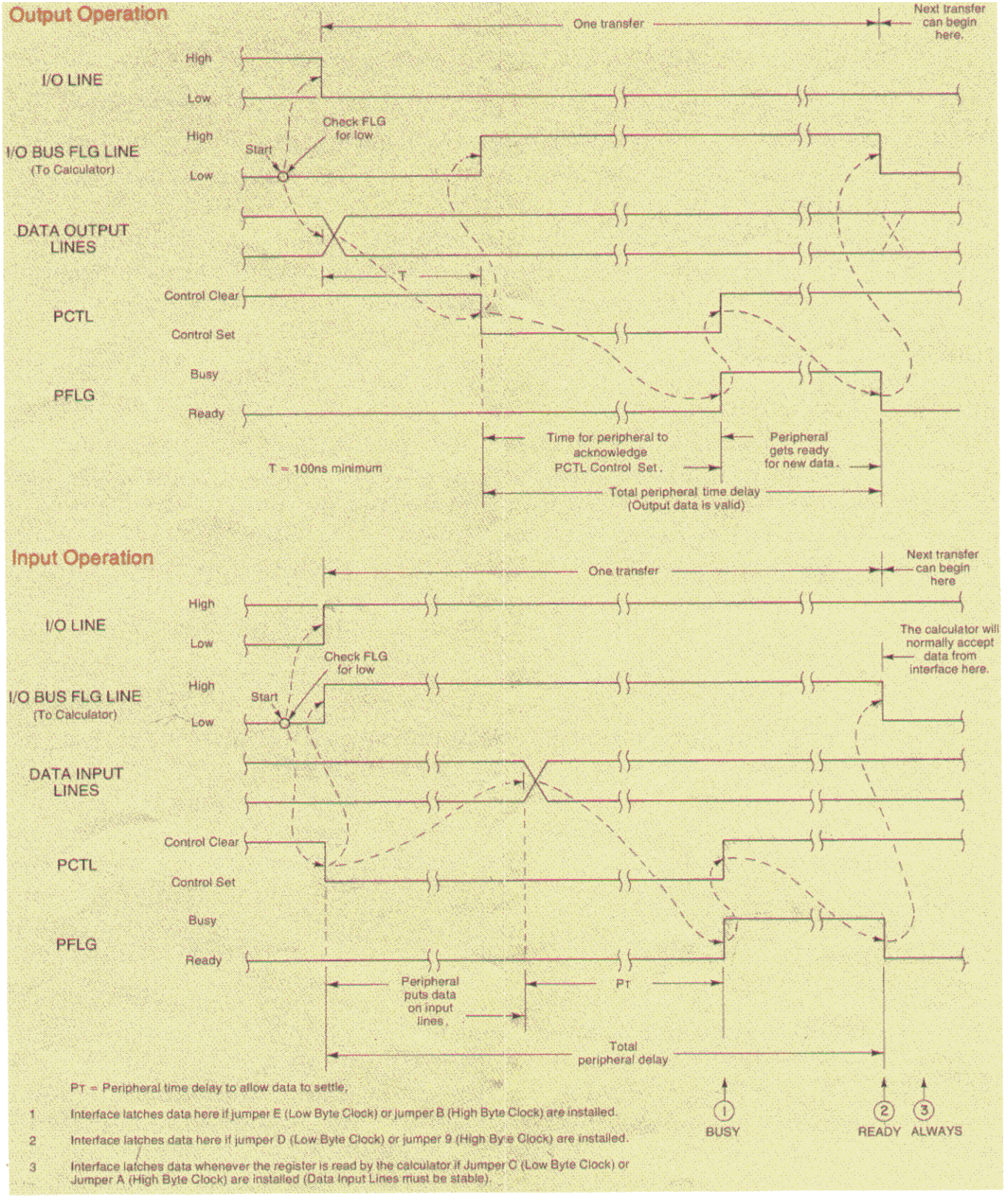
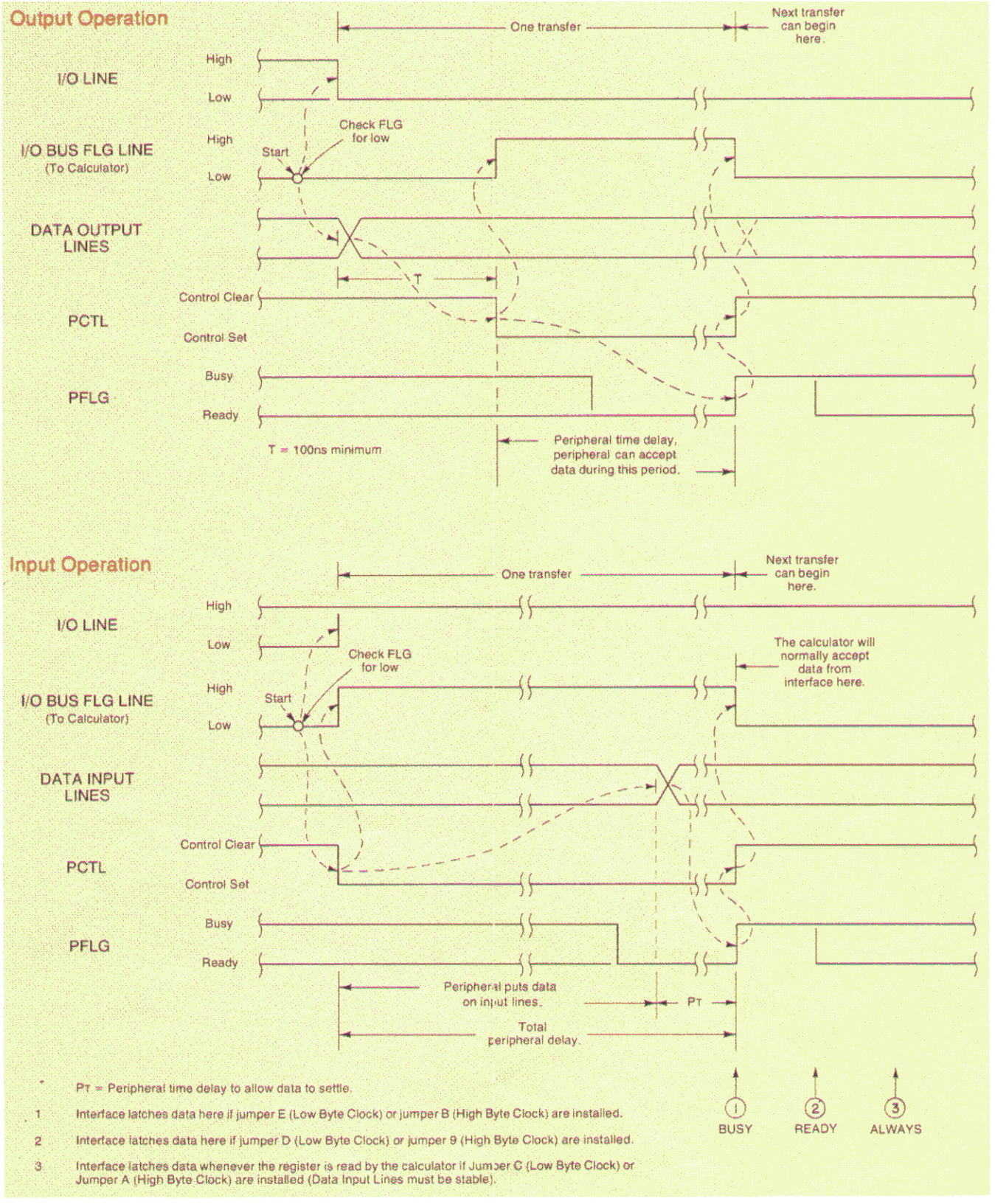



Figure 2-1. Full Mode Timing Diagram
(Jumper 6 Omitted)




PT = Peripheral time delay to allow data to settle.

1 Interface latches data here if jumper E (Low Byte Clock) or jumper B (High Byte Clock) are installed.

2 Interface latches data here if jumper D (Low Byte Clock) or jumper 9 (High Byte Clock) are installed.

3 Interface latches data whenever the register is read by the calculator if Jumper C (Low Byte Clock) or Jumper A (High Byte Clock) are installed (Data Input Lines must be stable).

Figure 2-2. Pulse Mode Timing Diagram
(Jumper 6 Installed)

# Modes of Operation

The interface has three programmable modes of operation. The modes are:

1. Standard Read/Write
2. Interrupt
3. DMA (Direct Memory Access)

The interface returns to the Standard Read/Write mode whenever the interface is reset. The modes are transparent to the peripheral, that is the peripheral does not know which mode is currently in effect (except by convention established by the programmer).

## Standard Read/Write Mode

The Standard Read/Write mode becomes active when the DMA and Interrupt modes are disabled (bits 6 and 7 of register 5 = 0). The calculator checks the I/O Bus FLG line before initiating a transfer, the calculator will wait if the FLG line is not low (Ready). A handshake is initiated for each data transfer, which causes the FLG line to go high (Busy). When the peripheral completes the handshake, using the PFLG line, the calculator can start another transfer.

## Interrupt Mode

The Interrupt mode becomes active when the calculator sets the interrupt enable bit (bit 7 of register 5 = 1) and DMA is disabled (bit 6 of register 5 = 0). An interrupt request is made when the peripheral indicates on the PFLG line that it is ready for the next handshake. Program control will be transferred to the service routine for the peripheral. The service routine must either start another data transfer or it must disable the interrupt mode (to prevent repeating the interrupt).

## DMA Mode

If jumper 7 is installed, the DMA mode becomes active when the calculator sets the DMA and Auto Handshake bits (bits 6 and 4 of register 5 = 1). The Interrupt mode will normally be enabled also. A DMA transfer is requested each time the peripheral indicates on the PFLG line that it is Ready for new data. The DMA transfer request will be repeated for each word of the DMA data block. After the last transfer, the DMA mode will be automatically disabled by the calculator and if the interrupt mode was enabled, an interrupt will be requested.

# Selecting Timing Capacitor

The interface is shipped from the factory with a built-in 100 ns output-time-delay. This delay is in the PCTL line. When the interface is used in a noisy electrical environment or when extra-long cables are used the output-time-delay should be increased to allow more time for the data to settle. Adding a capacitor to the configuration board (see Figure 2-3) will lengthen the output-time-delay. Typically a .002$\mu$f capacitor is installed, this will increase the output-time-delay to about 8$\mu$s.

The formula for the value of the capacitor needed for a required time delay is:

$$C \approx \frac{T-100}{4}$$

Where:

C = Capacitance (in pico-farads)
T = Total time delay required (in Nano-seconds)

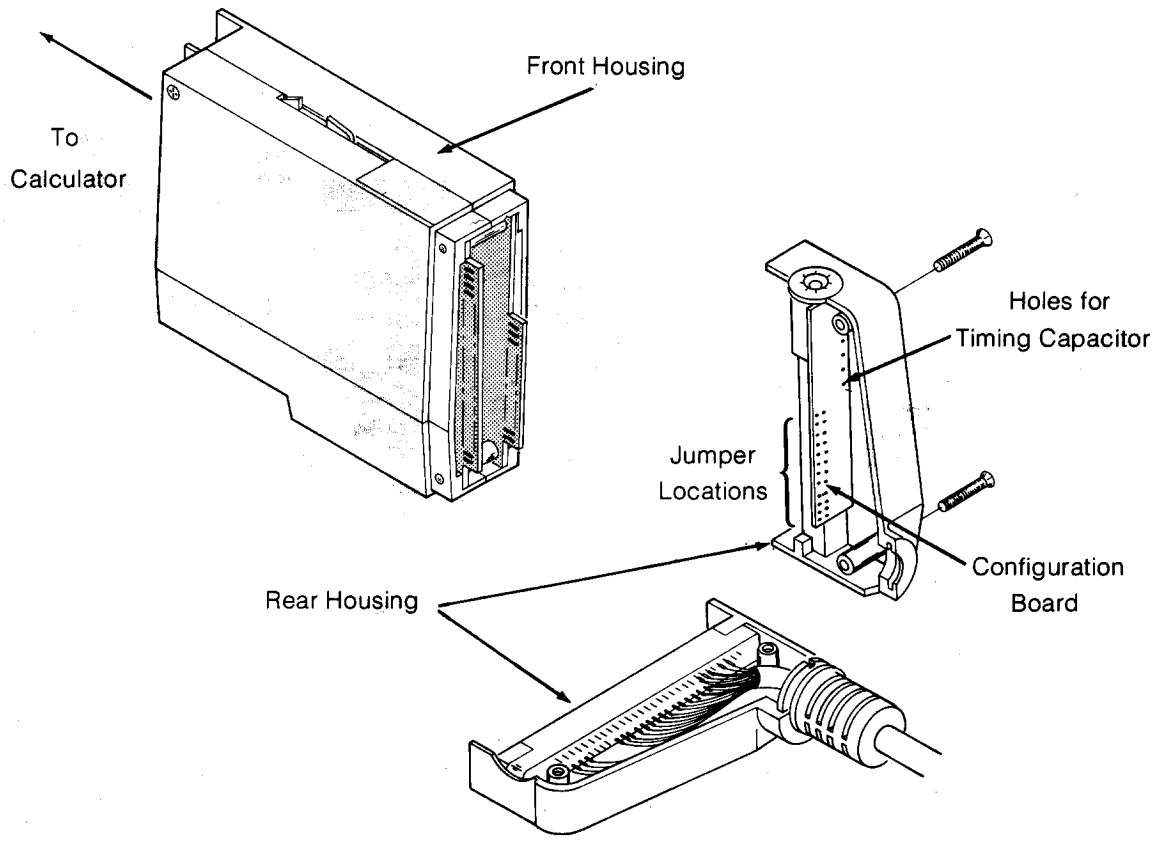



Figure 2-3. Configuration Board

# Cable Preparation

Prepare the end of the interface cable as shown in Figure 2-11. Notice that the outer shield and bare wire are not connected to the peripheral.

> **NOTE**
>
> Improper operation will result, due to crosstalk, if the outer shield or outer bare wire is connected to the peripheral ground.

Use heatshrink tubing or tape to insulate the shields as shown in Figure 2-11.

## Procedure

Refer to Figure 2-11.

1. Cut the cable to the required length, allow some length for slack.
2. Strip off the outer plastic jacket about 10 cm (4 inches).
3. Cut off the outer shield and outer bare wire even with the outer plastic jacket.
4. Cover the end of the jacket and outer shield with heatshrink tubing or tape.
5. Cut back the inner shield and its nylon jacket to within 2.5 cm (1 inch) of the outer jacket. DO NOT cut off inner bare wire.
6. Cover the end of the inner shield and nylon jacket with heatshrink tubing or tape.

> **NOTE**
>
> The inner and outer shields must not short together.

7. Strip and connect the cable wires as required by your peripheral.
8. Be sure to connect the logic ground wires (refer to Figure 2-11) to your peripheral's logic ground.
9. Isolate unused wire with heatshrink tubing or tape.

# Recommended Driver Circuits

## Data Lines

Each of the data-input lines on the interface are connected to input latches. A resistive divider is connected to each of the input lines, these dividers hold the voltage at about 3.4 volts when the cable is disconnected. The input voltage to these lines must not exceed 5.5 volts.

Here are typical specifications:

- I in low = 2 mA
- V in max = 5.5 V
- V in high > 2 V
- V in low < 0.7 V

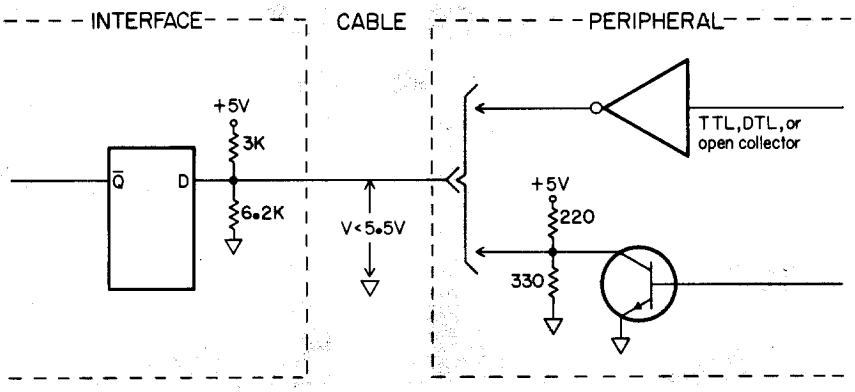



Figure 2-4. Recommended Peripheral Driver Circuit

## Other Lines

The PFLG, PSTS, and EIR signals are received by Schmitt triggers circuits. These circuits accept signals with slow rise and fall times, and provide good noise margins. Although the voltage on these lines must not exceed 5.5 volts, there is no restriction on the input rise and fall time. Either of the driver circuits shown in Figure 2-4 may be used as drivers for these lines.

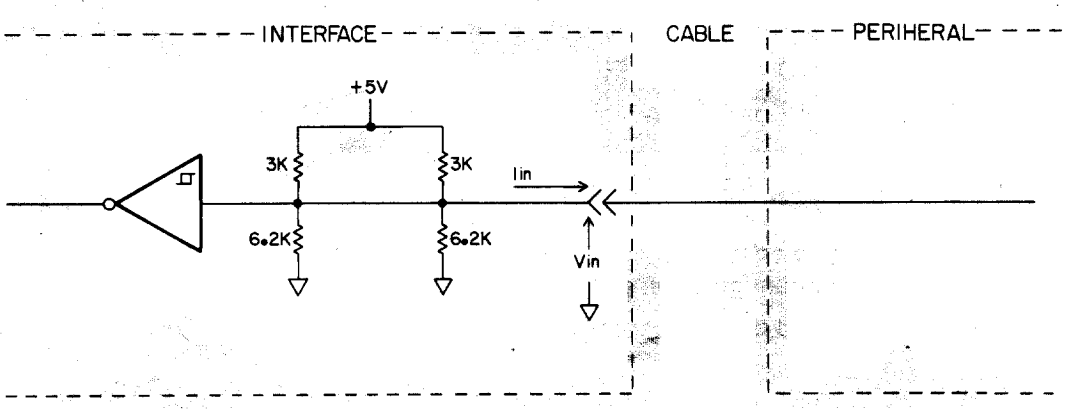



Figure 2-5. Interface PFLG, PSTS, and EIR Receiver Circuit

# Recommended Receiver Circuits

Each output line from the interface is driven by an open-collector circuit. The current-sinking capability of each driver is 40 mA and the breakdown voltage is 30 volts. Do not apply a negative voltage to the output lines.

Here are typical specifications:

- V out low { at (I out low = 16 mA) = 0.4 V max.
  at (I out low = 40 mA) = 0.7 V max.
- V out high (open collector) = 30 V max.
- I out low = 40 mA max.
- I out high at (V out high max.) = 250 $\mu$A

Since each driver has an open collector, the peripheral receiving circuit must have a positive pull-up voltage (not to exceed 30 V) and must be restricted to sourcing less than 40 mA. Recommended receiving circuits are shown in Figure 2-6.

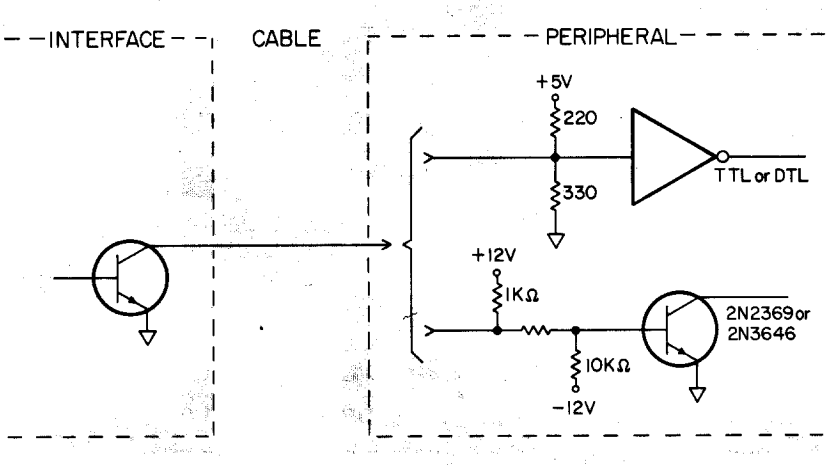



Figure 2-6. Recommended Peripheral Receiver Circuits

# Configuring the Logic

The following sections describe the logic lines available to the user. A description of the function and associated configuration jumpers are given for each logic line. A jumper reference is provided at the end of this chapter.

# Data Input Lines

16 Bits; DI0 through DI15; DI0 = LSB, DI15 = MSB

The 16 data input lines can be configured as two separate 8-bit bytes or one 16-bit word. In either case all data lines are latched by the input registers.

You can select the logic sense, positive or negative true, for the input data lines. The use of negative true logic is recommended for these lines. The inversion necessary, for the interface to operate on positive true logic, is done by the calculator.

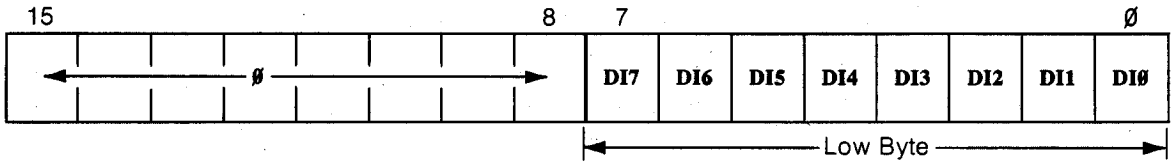
You have a choice of three clock sources that can be used for the data input latches. Refer to Data Handshake in this chapter.

Table 2-2. Data Input Lines

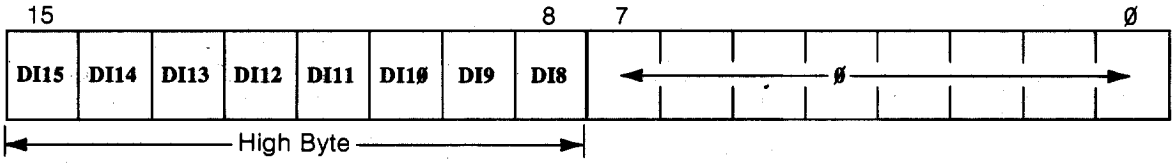
| Mnemonic | | Interface Connector Pin No. | Wire Color Code | |
|---|---|---|---|---|
| Low Byte | DI0 | B-17 | Black | (0) |
| | DI1 | B-16 | Brown | (1) |
| | DI2 | B-15 | Red | (2) |
| | DI3 | B-14 | Orange | (3) |
| | DI4 | B-13 | Yellow | (4) |
| | DI5 | B-12 | Green | (5) |
| | DI6 | B-11 | Blue | (6) |
| | DI7 | B-10 | Violet | (7) |
| High Byte | DI8 | B-9 | White/Brown/Red | (912) |
| | DI9 | B-8 | White/Brown/Orange | (913) |
| | DI10 | B-7 | White/Brown/Yellow | (914) |
| | DI11 | B-6 | White/Brown/Green | (915) |
| | DI12 | B-5 | White/Red/Orange | (923) |
| | DI13 | B-4 | White/Red/Yellow | (924) |
| | DI14 | B-3 | White/Red/Green | (925) |
| | DI15 | B-2 | White/Red/Blue | (926) |

Byte Mode (Jumper B not installed)

Register 4




Register 6




Word Mode (Jumper B installed)

Registers 4 and 6

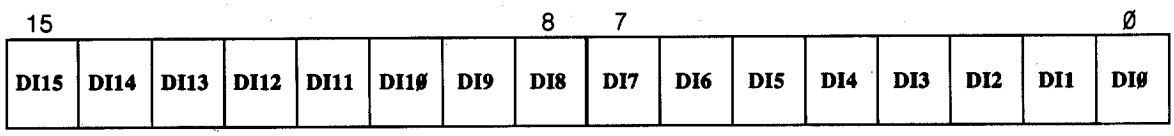



Figure 2-7. Data Input Bit Assignment

## Associated Configuration Jumpers

Positive- or Negative-true Input Logic

- Jumper 1 – When installed, sets bit 3 in the status register to indicate to the calculator that the input data is to be complemented, bit for bit, before being used or stored. This changes the input lines to positive true logic (ground = logic zero).

Word or Byte Mode

- Jumper B – When installed, selects the word input mode.

Low Byte Clock – Install **Only One** of the three jumpers.

- Jumper D – Clocks the low byte input data when PFLG goes ready.
- Jumper E – Clocks the low byte input data when PFLG goes busy.
- Jumper C – Clocks the low byte input data at the time the calculator reads the register.

High Byte Clock – Install **Only One** of the three jumpers.

- Jumper 8 – Clocks the high byte input data when PFLG goes ready.
- Jumper 9 – Clocks the high byte input data when PFLG goes busy.
- Jumper A – Clocks the high byte input data at the time the calculator reads the register.

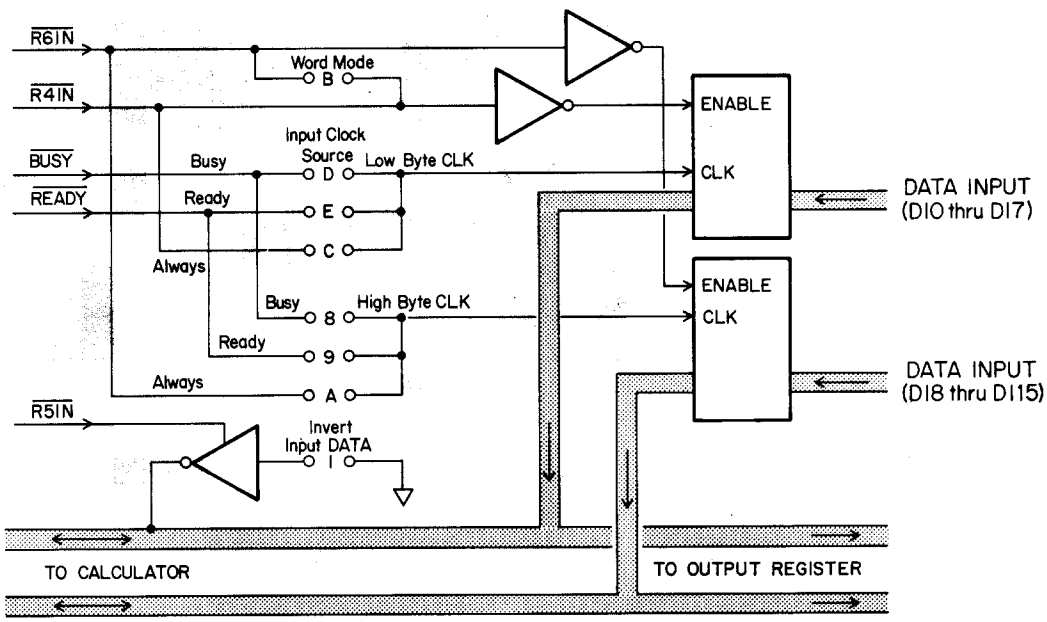



Figure 2-8. Data Input Clocks

# Data Output Lines

16-Bits; DO0 through DO15; DO0 = LSB, DO15 = MSB

The 16 data output lines can be configured as two separate 8-bit bytes or one 16-bit word. In the byte mode only eight of the 16-bits sent to the interface from the calculator are latched. The other 8-bits on the output lines are not affected. The output bits are glitch free and always present at the output connector unless changed by the calculator. Unused lines can be left unconnected or grounded.

You can select the logic sense, positive or negative true, for the output data lines. The use of negative true logic is recommended for these lines. The inversion necessary, for the interface to operate on positive true logic, is done by the calculator.

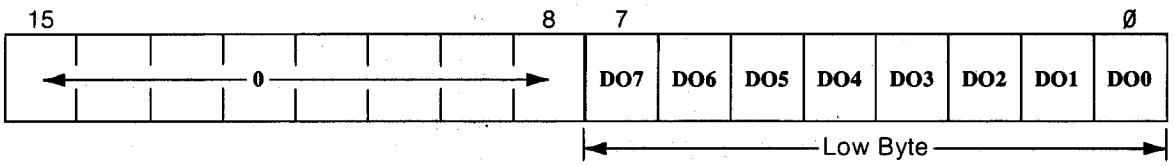
Normally the state of the output data lines is undetermined when the calculator is switched on. By installing jumper wire E1 and diode CR1 (refer to the schematic and component locator), all of the output data lines are preset to low each time the calculator is switched ON or when the RESET key is pressed.

**Table 2-3. Data Output Lines**

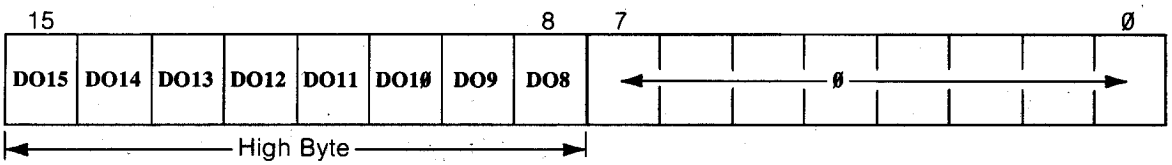
| Mnemonic | | Interface Connector Pin No. | Wire Color Code | |
|---|---|---|---|---|
| Low Byte | DO0 | A-17 | White/Black | (90) |
| | DO1 | A-16 | White/Brown | (91) |
| | DO2 | A-15 | White/Red | (92) |
| | DO3 | A-14 | White/Orange | (93) |
| | DO4 | A-13 | White/Yellow | (94) |
| | DO5 | A-12 | White/Green | (95) |
| | DO6 | A-11 | White/Blue | (96) |
| | DO7 | A-10 | White/Violet | (97) |
| High Byte | DO8 | A-9 | White/Orange/Yellow | (934) |
| | DO9 | A-8 | White/Orange/Green | (935) |
| | DO10 | A-7 | White/Orange/Blue | (936) |
| | DO11 | A-6 | White/Orange/Violet | (937) |
| | DO12 | A-5 | White/Yellow/Green | (945) |
| | DO13 | A-4 | White/Yellow/Blue | (946) |
| | DO14 | A-3 | White/Yellow/Violet | (947) |
| | DO15 | A-2 | White/Yellow/Gray | (948) |

Byte Mode (Jumper F not installed)

Register 4




Register 6




Word Mode (Jumper F installed) YES

Register 4 and 6

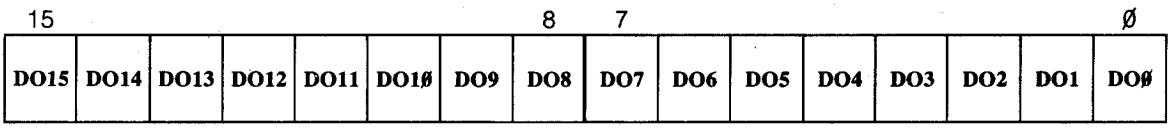



Figure 2-9. Data Output Bit Assignment

## Associated Configuration Jumpers

Positive- or Negative-true Output Logic

- Jumper 2 – When installed, sets bit 2 in the status register to indicate to the calculator that the output data is to be complemented, bit for bit, before being sent to the interface. This changes the output lines to positive true logic (ground = logic zero). YES

Word or Byte Mode

- Jumper F – When installed, selects the word output mode.

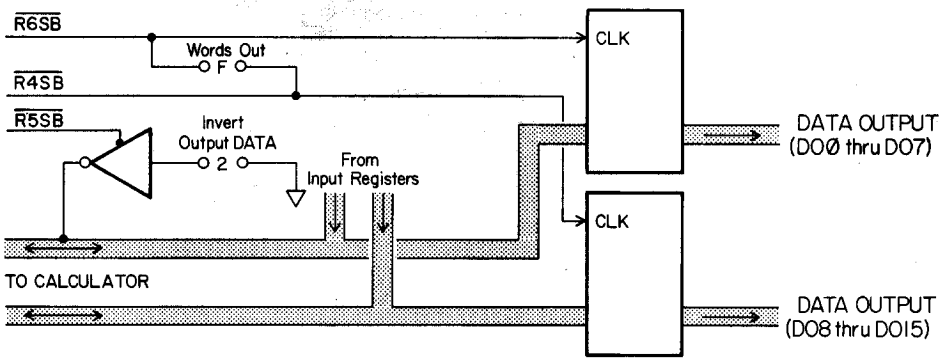



Figure 2-10. Data Output Strobe

# Peripheral Control Line

1 Bit; PCTL

This line is paired with PFLG and is used to synchronize (handshake) the calculator and your peripheral. The two states of this line are control set and control clear. The peripheral clears control on the PCTL line by a ready-to-busy transition on the PFLG line (see Peripheral Flag Line).

PCTL is delayed to allow new output data to settle. This delay time is set to 100 ns minimum at the factory, but it may be changed to a larger value by adding a capacitor to the Configuration Board (see Selecting Timing Capacitor).

The logic sense of the PCTL line can be inverted by installing jumper 3 (control set = high; control clear = low).

Table 2-4. Peripheral Control Line

| Mnemonic | Interface Connector Pin No. | Wire Color Code |
|---|---|---|
| PCTL | A-19 | White/Gray (98) |

## Associated Configuration Jumper

Logic Sense

- Jumper 3 – When installed, complements the logical sense of PCTL (high = control set and low = control clear).

# Peripheral Flag Line

1 Bit; PFLG

This line must be driven to complete a transfer. It is paired with PCTL and is used to synchronize (handshake) the calculator and your peripheral. If no handshake is required, then PFLG must be connected to PCTL at the peripheral end of the cable, and jumper 3 or 4 must be installed (not both).

One of two modes of handshake can be selected: by installing jumper 6 the pulse mode of handshake is selected; by omitting jumper 6 the full mode of handshake is selected. Refer to the "Data Handshake" section in this chapter.

The logic sense of the PFLG line can be inverted, by installing jumper 4 (busy = low; ready = high). This line is also used to request "interrupt." *

Table 2-5. Peripheral Flag Line

| Mnemonic | Interface Connector Pin No. | Wire Color Code |
|---|---|---|
| PFLG | B-19 | Gray (8) |

## Associated Configuration Jumpers

Logic Sense

- Jumper 4 – When installed, complements the logical sense of PFLG, (high = ready and low = busy).

Handshake Mode

- Jumper 6 – When installed, selects the pulse mode of handshake.

*Refer to the appropriate ROM manual for operation.

# Peripheral Status Line

1 Bit; PSTS

Use of the PSTS line is optional, but encouraged. Your peripheral should use this line to signal the calculator that all is "OK". When the peripheral is powered down, interlocks are broken, it is out of paper, etc., a "not OK" signal can be sent to the calculator on this line.

The logic sense of the PSTS line can be complemented by installing jumper 5, making low the OK state and high the not OK state. This allows you to detect an open cable, because when this line is open it will float high – not OK.

**Table 2-6. Peripheral Status Line**

| Mnemonic | Interface Connector Pin No. | Wire Color Code |
|---|---|---|
| PSTS | B-20 | White/Black/Gray (908) |

INSTALL JUMPER 5 AND GROUND THIS LINE AT CONNECTOR!

## Associated Configuration Jumper

Logic Sense

- Jumper 5 – When installed, complements the logic sense of PSTS (high = not OK and low = OK).

# Extended Status Input Lines

2 Bits; STI0, STI1

Use of these two input lines is optional. They can be used for any purpose that reflects the status of the peripheral. The state of these lines can be examined by reading the status register.

**Table 2-7. Extended Status Lines**

| Mnemonic | Interface Connector Pin No. | Wire Color Code |
|---|---|---|
| STI0 | B-22 | White/Brown/Blue (916) |
| STI1 | B-23 | White/Brown/Violet (917) |

# Extended Control Output Lines

2 Bits; CTL0, CTL1

Use of these two output lines is optional. They can be used for any purpose to control the peripheral. These lines are latched and can be set or cleared (1 = low, 0 = high) by outputing to the control register. These lines are undetermined at power up.

Table 2-8. Extended Control Output Lines

| Mnemonic | Interface Connector Pin No. | Wire Color Code | |
|---|---|---|---|
| CTL0 | A-22 | White/Red/Violet | (927) |
| CTL1 | A-23 | White/Red/Gray | (928) |

# Input/Output Direction Control Line

1 Bit; I/O

Use of this output line is optional. This line is always valid during PCTL control set and indicates to the peripheral which direction the data transfer is to go. This line is high for an input operation and low for an output operation.

Table 2-9. I/O Direction Control Lines

| Mnemonic | Interface Connector Pin No. | Wire Color Code | |
|---|---|---|---|
| I/O | A-20 | White/Black/Brown | (901) |

# Peripheral Reset Line

1 Bit; PRESET

Use of this line is optional. It can be used to initialize your peripheral. It is pulsed low when the calculator is switched ON and when the RESET key is pressed. It is also pulsed low when the reset bit is sent to the control register. The minimum Preset pulse width is 230 ns.

Table 2-10. Preset Line

| Mnemonic | Interface Connector Pin No. | Wire Color Code |
|---|---|---|
| PRESET | A-21 | White/Black/Red (902) |

# External DMA Interrupt Request Line

1 Bit; EIR

During DMA (Direct Memory Access) the EIR line can be used to cause the calculator to interrupt or abort the DMA transfer before completion of the entire data block transfer. Normal interrupt operation uses the PFLG line to request interrupt service, not the EIR line.

Table 2-11. DMA Line

| Mnemonic | Interface Connector Pin No. | Wire Color Code |
|---|---|---|
| EIR | B-21 | White/Brown/Gray (918) |

## Associated Configuration Jumper

Allow DMA

- Jumper 7 – When installed, allows the calculator to activate the DMA mode of operation on the interface.

# Grounding

Connect the ground wires indicated in Figure 2-11 to the peripheral's logic ground. Notice that the outer shield and outer bare wire are cut off and not connected to the peripheral.

> **NOTE**
>
> Improper operation may result, due to crosstalk, if the outer shield or outer bare wire is connected to the peripheral ground.

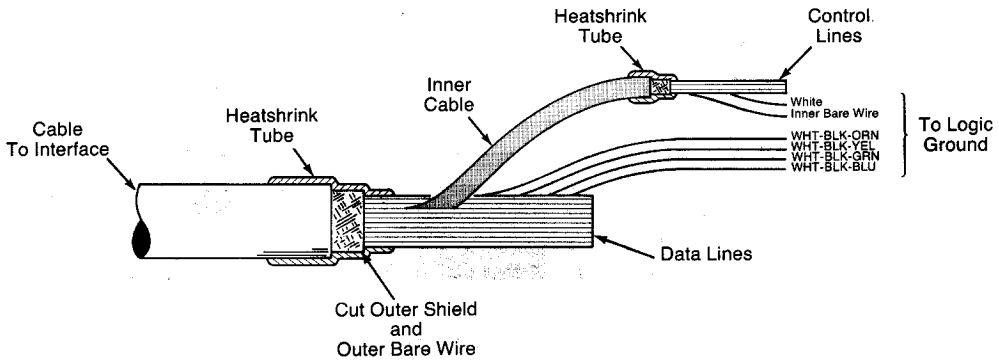



Figure 2-11. Cable Preparation and Ground Wires

# Jumper Reference

Table 2-12. Configuration Jumpers

| Jumper | Function, when installed |
|---|---|
| 1 | Sets bit 3 in the status register, changes the input data lines to positive true logic. |
| 2 | Sets bit 2 in the status register, changes the output data lines to positive true logic. |
| 3 | Complements the logic sense of PCTL; high = control set and low = control clear. |
| 4 | Complements the logic sense of PFLG; high = ready and low = busy. |
| 5 | Complements the logic sense of PSTS; high = not OK and low = OK. |
| 6 | Changes the handshake from full to pulse. |
| 7 | Allows the calculator to activate the DMA (Direct Memory Access) mode of operation. |
| 8 | Clocks the high input byte when PFLG goes busy from ready. |
| * 9 | Clocks the high input byte when PFLG goes ready from busy. |
| A ✓ | Clocks the high input byte at the time the calculator reads the register. |
| B ✓ | Selects the words input mode. |
| C ✓ | Clocks the low input byte at the time the calculator reads the register. |
| * D | Clocks the low input byte when PFLG goes ready from busy. |
| E | Clocks the low input byte when PFLG goes busy from ready. |
| F ✓ | Selects the words output mode. |

*Select only one of these three.

# Chapter 3
# Service

## Introduction

This chapter contains a brief Block Diagram Description, Troubleshooting and Repair information, and a Theory of Operation section. This information will help you service the 98032A Interface.

If you have difficulty repairing the interface or if you would rather have HP repair it, contact the nearest Sales and Service office for assistance; office locations are listed at the back of this manual.

## Block Diagram Description

The interface consists of four registers and control circuits. Refer to the block diagram and schematic (Figure 3-1 and 3-4).

### Registers

The four registers on the interface are registers 4, 5, 6 and 7.

- Register 4 is two 8-bit registers, one for low-byte data-in and one for low-byte data-out.
- Register 6 is two 8-bit registers, one for high-byte data-in and one for high-byte data-out.
- Register 5 (8-bits) is the status "in" register and control "out" register. The bit registers that make up Register 5 are located at various positions on the schematic. To find their locations trace the R5IN and R5SB (U13) lines to the various bit registers. The bit position within Register 5 can be found by following the bit registers input or output to the data bus and noting the data line mnemonic.
- Register 7 has no bits and always returns zero, but outputting data to R7 (R7SB) causes control to be set on the PCTL line, beginning a new handshake.

# Program Control of the Registers

### Write and Write Binary

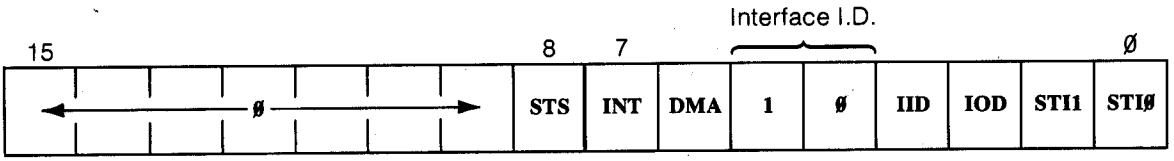
These operations transfer data to the output line in bit-parallel, character-serial fashion. Refer to the appropriate ROM Manual for syntax and operation. Refer to Figure 2-9 for bit assignment.

### Read and Read Binary

These operations clock the input data into the input register and then transfer it to the calculator, in bit-parallel, character-serial fashion. Refer to the appropriate ROM Manual for syntax and operation. Refer to Figure 2-7 for bit assignments.

### Read Status

This operation transfers, to the calculator, the decimal value of the interface status register. The bit locations and their mnemonics are shown below.

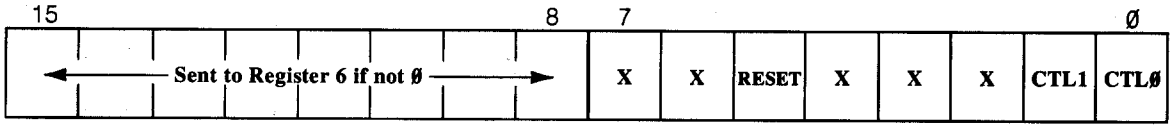



**STIØ** = Status bit Ø
**STI1** = Status bit 1
**IOD** = Invert Output Data
**IID** = Invert Input Data
**DMA** = Direct Memory Access Enable
**INT** = Interrupt Enable
**STS** = Status (Peripheral)

When the 9825A Calculator executes a read status statement in reference to the 98032A Interface, it will receive a value representing the lower 9 bits (bits 0 – 8) of the status register.

### Write Control

This operation allows you to reset the interface and control the peripheral. Bits 0 and 1 can be used to control a peripheral, via the CTL0 and CTL1 lines, respectively. The bits and their mnemonics are shown below.




**CTLØ** = Control Bit Ø
**CTL1** = Control Bit 1
**RESET** = Resets the Interface
**X** = Don't Care

## Control Circuits

The control circuits consist of the select-code switch and decoder, I/O register decoder, interrupt logic, DMA logic and the handshake logic.

### Peripheral Address Decoder

The peripheral address decoder compares the I/O Bus address from the calculator with that of the select-code switch; a true output (MYPA) is generated when the address matches.

### I/O Register Decoder

The I/O register decoder circuit decodes instructions from the calculator. Both the direction of transfer and the register selection are decoded, then an input (IN) or strobe (SB) signal is generated.

### Interrupt Logic

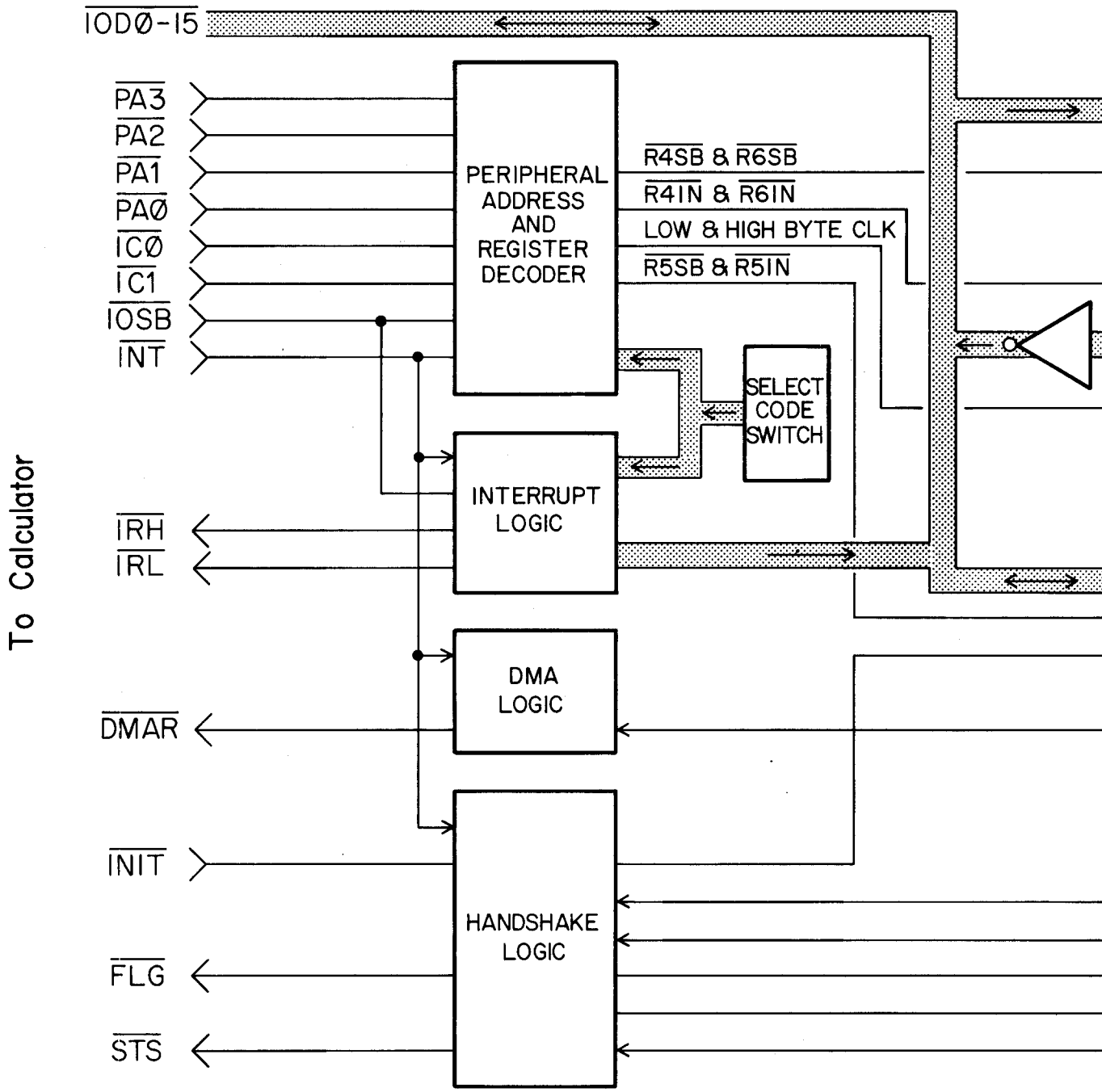
The Interrupt logic circuit checks the peripheral to see if it requires service.

### DMA Logic

The Direct Memory Access (DMA) logic circuit allows data to be transferred directly to or from the calculator memory.

### Handshake Logic

The handshake logic circuit synchronizes the input-output transfer, via the PCTL and PFLG line. Refer to Data Handshake in Chapter 2.


IOD0-15
PA3
PA2
PA1
PA0
IC0
IC1
IOSB
INT
PERIPHERAL ADDRESS AND REGISTER DECODER
R4SB & R6SB
R4IN & R6IN
LOW & HIGH BYTE CLK
R5SB & R5IN
SELECT CODE SWITCH
INTERRUPT LOGIC
IRH
IRL
DMA LOGIC
DMAR
INIT
HANDSHAKE LOGIC
FLG
STS
To Calculator

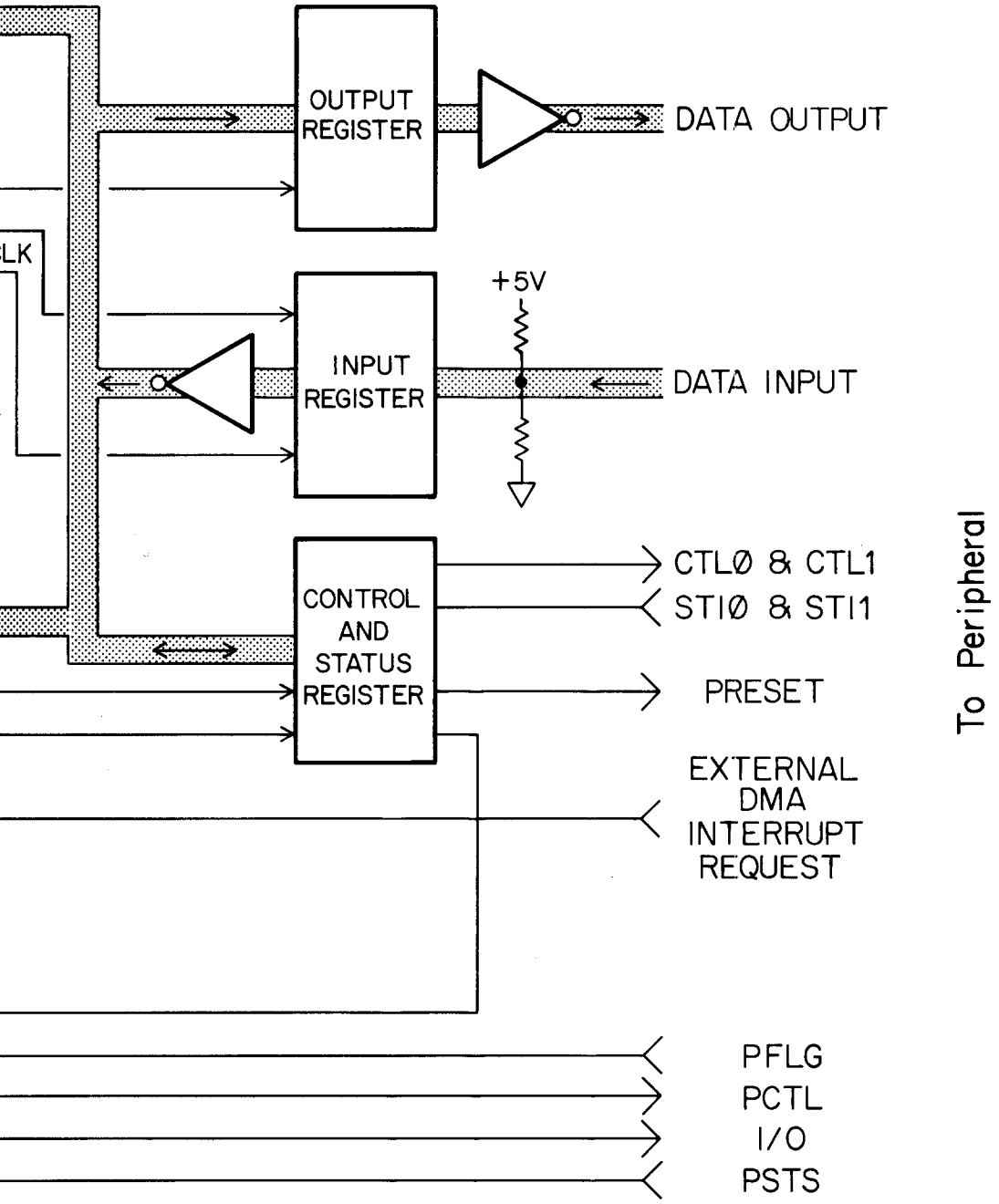



Figure 3-1. Block Diagram

# Interface Operational Test

The interface can be checked for proper operation by removing the rear housing and installing the optional test connector (P/N 98241-67932) and then running an interface test program. An example test program that can be used with the 9825A Calculator is given in the Appendix.

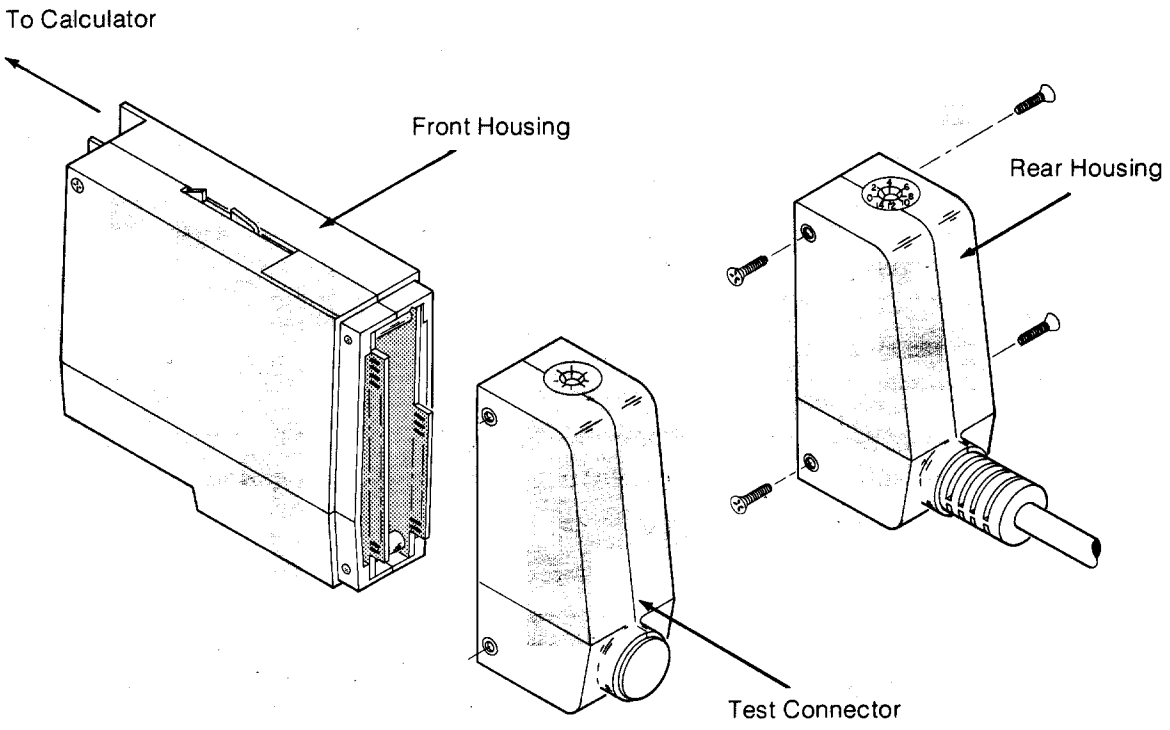



Figure 3-2. Test Connector Installation

The test connector connects the output data lines to the input data lines. This will allow you to write to the interface and then read the same data back into the calculator.

The control output lines are connected to the status input lines to allow a similar write/read test of these lines. PCTL and PFLG are connected together to allow the test connector to operate as a peripheral with handshake.

If the interface operates properly with the test connector but fails to operate the peripheral, recheck the peripheral and refer to information concerning peripherals installation and operation.

If the interface fails to operate properly with the test connector, refer to the next section of this chapter.

# Troubleshooting and Repair

The following procedures assume that the calculator, ROM(s) and peripheral device are operating correctly. If necessary, disconnect the interface from the calculator and perform all other applicable test procedures before assuming that the interface is defective.

## Broken Trace Repair

If one or more traces are open or have high resistance, the trace should be bridged using insulated wire. Note – the boards are of multi-layer construction, and therefore require good soldering technique to prevent damage.

> **CAUTION**
>
> To help prevent damage to the circuit boards use a low-temperature soldering iron when making repairs or replacing parts.

## Recommended Equipment

The following is a list of equipment that will aid in troubleshooting the 98032A 16-Bit Interface:

- An HP 10525A Logic Probe (or equivalent)*
- Appropriate calculator and ROM(s)
- A Test Connector (98241-67932)
- An Extender Board (98241-67901)

For checking most signals within the interface, any general-purpose oscilloscope or logic probe can be used; it should be capable of indicating the presence of TTL level signals with pulse widths greater than 200 ns.

*Any device capable of indicating the state of TTL signals.

## Equipment Set Up

To make the following tests or checks it will be necessary to remove the case from the interface circuit boards. To do this, remove the screws from the sides of the interface. Use the Extender Board to reconnect the interface to the calculator.

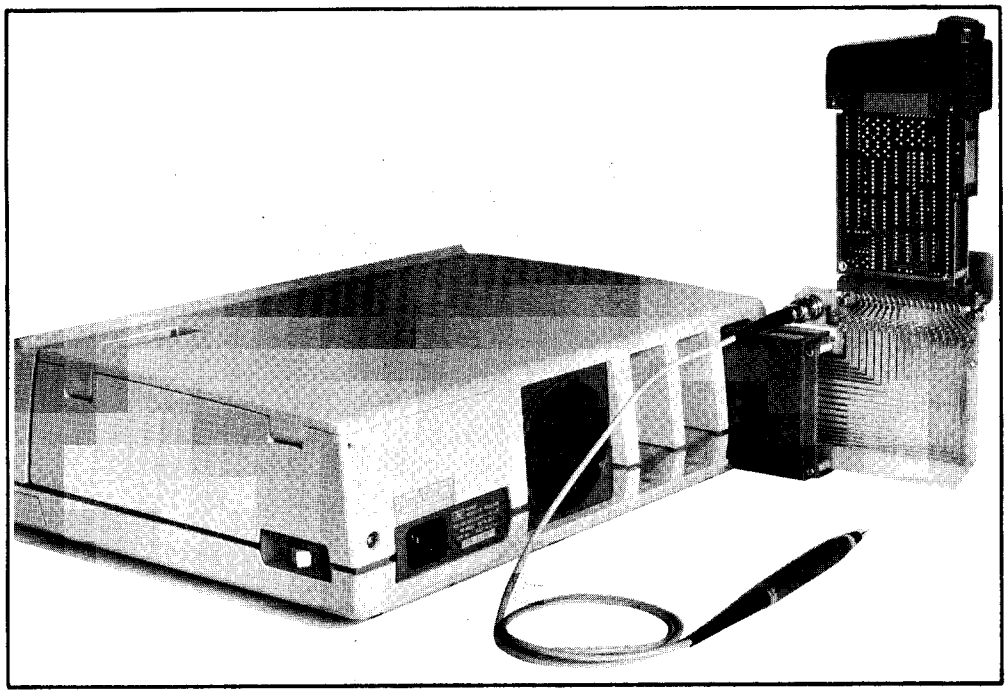


Figure 3-3. Interface Test Set Up

## Procedure

The following procedure will help isolate a problem to a defective circuit on the interface. Troubleshoot the circuit by checking the associated components and signals. Refer to the section of the Theory of Operation that describes the defective circuit.

Refer to the schematic and component location for the test points and circuits referred to in the following procedure:

1. Execute a statement to write a binary value (e.g., `wtb 2,0`*) to the interface. Use the logic probe to check the state of the I/O line (P1-31), it should be low.
2. Execute a statement to read binary (e.g., `rdb (2)`*) from the interface, use the logic probe to check the state of the I/O line (P1-31), it should be high.

*9825A General I/O ROM Syntax

3. Repeat steps 1 and 2 but check the PCTL line (P1-32) for a low pulse and the PFLG line (P1-34) for a high pulse, each time a read or write operation is expected.
4. Execute a statement to write binary the value "−1" to the interface (e.g., `wtb2,-1`), use the logic probe to check the state of each of the output data lines, they should all be high. The output data lines can be checked at the collectors of the output transistors (Q20 thru Q35).
5. Repeat the above step, but write binary the value "O" to the interface, recheck the output data lines, this time they should all be low.
6. Execute a read binary statement, the returned value should be equal to the binary data pattern at the output data lines.

# Theory of Operation

This section describes in detail the operation of the 98032A 16-Bit Interface, refer to the block diagram and schematic (Figures 3-1 and 3-5). The eight major sections of the schematic and block diagram are:

1. Input Data Register
2. Output Data Register
3. Handshake Logic
4. Control and Status Register
5. Peripheral Address Decoder
6. Register Decoder
7. Interrupt Logic
8. Direct Memory Access (DMA) Logic

## Input Data register

The input data register consists of four quad-latches (U22 through U25). The input data lines from the peripheral are connected to resistive dividers (3K ohms to +5V and 6.2K ohms to ground) and the input latches. The terminations cause the input lines to be biased at 3.4 volts. The lower 8-bits go to the low-byte latches while the upper-bits go the high-byte latches. By omitting jumper B (byte mode) the two input latches can be separately controlled. By installing jumper B (word mode) the enable lines are tied together, thus allowing both bytes to be sent to the calculator at the same time. If jumper B is absent, the low byte is received when the calculator reads from I/O register 4 (R4IN) and the high byte is received when the calculator reads from I/O register 6 (R6IN). The output control line (pin 8) on the latches enables the outputs to the I/O bus.

### Input Clocks

There are three choices of clocking signals for the input latches of each byte. The input data can be latched each time the register is ready by installing jumper A and C. The other two choices are related to the transition of the PFLG line from the peripheral. Either the ready-to-busy (jumper B and E) or busy-to-ready (jumper D and 9) transition can be selected as the input data clock. The clock inputs are negative-edge triggered.

## Output Data Register

The output data register consists of four CMOS quad-latches (U26 through U29) which are controlled as two 8-bit bytes. Data from the calculator bi-directional data lines (IOD0 through IOD15) is always applied to the inputs of the latches.

### Output Clocks

The output data is latched on the rising edge of the clock signal (pin 9). The clocking signals (R4SB and R6SB) are generated whenever the calculator outputs to register 4 or register 6. Installing jumper F on the Configuration Board selects the words mode. This ties these two clocks together causing both 8-bit registers to latch new data from the I/O bus whenever either R4SB or R6SB is generated.

### Reset

The clear lines (pin 1) on the output latches are tied together at a pull-up resistor (R25-4). There is a space provided on the Data Board (A3) for a diode (CR1) and a jumper (E1) Which when installed presets the outupt lines to low whenever the interface is reset.

### Output Line Drivers

The output line drivers(Q20 through Q35) are driven by the output latches. These drivers are NPN-grounded-emitter transistors. They will each sink 40ma without exceeding .4 volts saturation voltage. Each transistor will cut-off when its base is driven low. The maximum safe collector voltage is 30 volts.

## Handshake Logic

The Handshake Logic is used to synchronize the data exchange between the peripheral and the calculator. The handshake is initiated by the calculator and terminated by the peripheral. The peripheral may take as little or as much time to respond as it requires (refer to Figures 2-1 and 2-2).

### PCTL Line

PCTL is the handshake signal from the calculator to the peripheral. Its active state is control-set. When the interface is RESET (e.g., during power up) the PCTL line state is set to control-clear. The default sense of the PCTL signal is low = control-set and high = control-clear. The sense may be inverted by installing jumper 3.

### PFLG Line

PFLG is the handshake line from the peripheral to the calculator. The two states of the PFLG line are busy and ready. The default sense of the PFLG line is low = busy and high = ready. The sense may be inverted by installing jumper 4.

### I/O Line

The I/O direction control flip-flop (U14A) controls the I/O line to the peripheral. The flip-flop's state is not affected by RESET and therefore is undetermined at power-up. A write operation will cause the I/O line to go low. A read operation will reset the I/O flip-flop, causing the I/O lines to go high. The I/O line driver (U18B) is an open-collector device capable of sinking 40ma with .7 volt maximum saturation voltage. The I/O line driver can withstand 30 volts maximum.

### STS Line

The STS line is examined by the calculator during an I/O operation. When STS is low, it indicates that the addressed interface is OK. If STS is high, there is either no interface addressed or the peripheral is NOT OK. The PSTS line from the peripheral controls the STS signal.

### PSTS Line

The PSTS line is received on the Data Board by a TTL Schmitt trigger (U30B). The output of the Schmitt trigger is sent to an exclusive-or-gate (U16B) which controls the logic sense of the PSTS line. The normal sense of the PSTS line is high = OK and low = NOT OK. Installing jumper 5 inverts the logic sense of this line. The calculator will issue an error if it tries an I/O operation and finds PSTS NOT OK.

### Output Handshake

When the calculator executes an output operation, data is sent to the output register and the I/O line to the peripheral will go low. After a short delay (100 ns, to allow the data to settle) the PCTL line goes to the control set state. This tells the peripheral that data is available. The peripheral responds by making a ready-to-busy transition on the PFLG line. This will cause the PCTL signal to return to the control clear state. If jumper 6 is installed (pulse mode) the handshake is complete and the peripheral is considered ready for another transfer.

If jumper 6 is not installed (full mode) then the interface will indicate busy until the PFLG line returns to the ready state. In this mode the calculator will not initiate another transfer until the PFLG line returns to the ready state.

### Input Handshake

For an input operation, the calculator waits for a low on the I/O bus FLG line, then changes the I/O line to high and the PCTL line to control set. The calculator then waits for the data on the input lines. The peripheral changes PFLG to the busy state, this returns PCTL to the control clear state. If jumper 6 is installed, the calculator will accept the data at this time. Without jumper 6 the calculator will wait until PFLG is returned to the ready state before accepting the data.

# Control and Status Register

### Interface Control Bits

The Control and Status Register (register 5) consists of a number of flip-flops that control the mode-of-operation of the interface. Auto Handshake, Interrupt, and Direct Memory Access (DMA) are enabled by outputing 1's to bits 4, 6, and 7 of register 5, respectively. These modes may be disabled, individually, by outputing 0's to their respective bits. All three modes are disabled by the RESET line (U10B pin 4) going low (e.g., at power-up). The RESET signal is generated in one of two ways:

1. An INIT initialize signal from the calculator I/O bus.
2. Outputting a 1 to bit 5 of register 5.

The calculator pulses INIT low whenever it powers up and when the RESET key is pressed. All interfaces connected to the calculator are sent INIT, whereas , only the currently addressed interface is reset by bit 5 in register 5. The RESET signal is buffered (U13A) and sent to the peripheral as "PRESET."

### Peripheral Control Bits

In addition to the control signals already mentioned, there are two peripheral control bits (CTL0 and CTL1 from Q36 and Q37) that can be used to further control the peripheral (e.g., to set special modes).

### Interface Status Bits

When the calculator reads register 5, status information about the interface and the peripheral is returned. The upper byte is always zero.

The lower byte gives the state of Interrupt Enable and DMA Enable in bits 7 and 6 respectively. Bits 5 and 4 are interface type-identifier bits; for 98032A Interface they are always 1 and 0 respectively. Bits 3 and 2 are the data inversion bits. If bit 3 equals 1, then jumper 1 is installed, indicating that the input data will be received as positive-true logic. If bit 2 equals 1, then jumper 2 is installed, indicating that the output data will be positive-true logic.

### Peripheral Status Bits

Bits 1 and 0 are the extended status bits (STI1 and STI0) from the peripheral. These two bits are latched just before the status is sent to the calculator. The peripheral status input lines do not have invertable logic sense (low = 1).

## The Peripheral Address Decoder

The Peripheral Address Decoder circuit (U4A through U4C) compares the peripheral address on the I/O bus to the select-code switch (S1) setting. If these two addresses are equal, the peripheral address decoder output (MYPA) is enabled (high). Without the Configuration Board installed MYPA is disabled by U18E. The peripheral address lines are, like all I/O bus lines, negative true. When MYPA is high, the interface can take control of the FLG and STS lines on the I/O bus.

## Register Decoder

MYPA is AND'd (U2A) with not-INT (no interrupt poll in progress) signal, this composite signal enables the Register Decoder (U13). The eight open-collector outputs of this decoder are the I/O register control signals. The I/O bus signal DOUT determines the direction of the transfer. The I/O bus signals IC1 and IC2 determine which register is being addressed. The following table relates the decoder output line mnemonics to its input lines.

Table 3-1. Register Decoder

| Calculator I/O Bus Lines | | | | Decoder Output Line |
|---|---|---|---|---|
| IC1 | IC2 | DOUT | IOSB | Mnemonic |
| 0 | 0 | 0 | X | R4IN |
| 0 | 1 | 0 | X | R5IN |
| 1 | 0 | 0 | X | R6IN |
| 1 | 1 | 0 | X | R7IN |
| 0 | 0 | 1 | 1 | R4SB |
| 0 | 1 | 1 | 1 | R5SB |
| 1 | 0 | 1 | 1 | R6SB |
| 1 | 1 | 1 | 1 | R7SB |

0 = low, 1= high, X = don't care

The Register Decoder output lines are normally high. Four of these signals (R4IN through R7IN) are used to control the input operations and the other four signals (R4SB through R6SB) are used to strobe (clock) the output data. During an interrupt poll the register decoder is disabled by INT.

## Interrupt Logic

The Interrupt logic is enabled and disabled by bit 7 of register 5. When interrupt is enabled (bit 7 equals 1) the select-code switch setting determines which level of priority and which poll response bit the interface will use. The high-level interrupt (high priority) is used by devices with select codes 8 through 15. The low-level interrupt (low priority) is used by interfaces with select codes 0 through 7.

### Poll Response

The lower three bits from the select-code switch are used as the inputs to a three-to-eight line decoder (U1). The open-collector outputs of the decoder are connected to the IOD0 through IOD7 lines. To request an interrupt, the peripheral changes the PFLG line from ready to busy, this causes the interface to ground either IRL or IRH as determined by P3 of the select-code switch. These interrupt request lines are Wire-Or'd on the I/O bus. This allows more than one interface to simultaneously request interrupt service on the same interrupt level. When an interrupt request is received by the calculator and the interrupt system is active, the calculator does an interrupt poll. To initiate a poll the calculator grounds the INT line, then each device requesting service will respond in the following manner. During a low-level poll interfaces on select-codes 0 throgh 7 that are requesting service will ground the bit corresponding to its select code value. Interfaces 8 through 15 that are requesting service will respond to a high-level poll by grounding the bit corresponding to their select-code value minus eight. For example, an interface set to select-code 9 will ground bit 1 (9-8 = 1) during a high-level poll if it is requesting service. Only those interfaces enabled for interrupt and actively requesting service on IRL or IRH will respond to a poll.

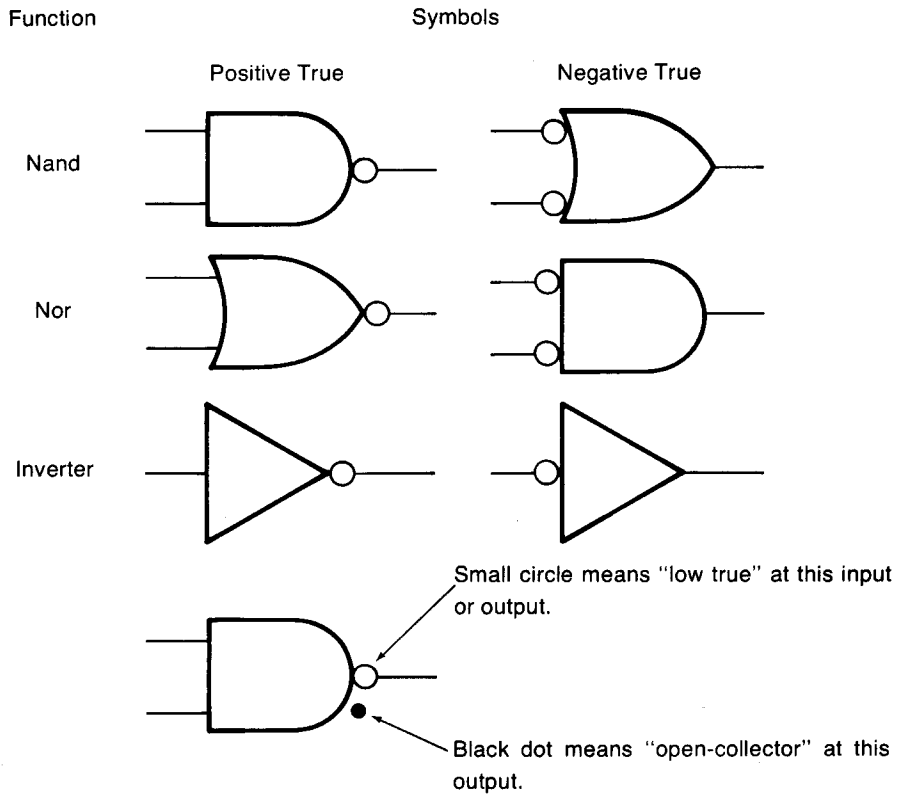
Before an interface actively requests service, it will first check to see if a interrupt poll is already in progress. If a poll is in progress it will wait until INT returns to high. This is accomplished with the Interrupt-active flip-flop (U5A) and an AND gate (U2B). The interrupt request logic sets the Interrupt-active flip-flop when interrupt enable (bit 7 of register 5) is set, Direct Memory Access (DMA) is disabled and FLAG is high, or when External Interrupt Request (EIR) is low.

## Direct Memory Access Logic

To enable the interface for Direct Memory Access (DMA) transfer, the calculator will set the DMA enable flip-flop (U11A) and the Auto Handshake enable flip-flop (U14B). In the DMA mode, a DMA transfer is requested when FLAG is high. Interrupt requests are blocked by U7A when DMA enable is set. Interrupt may be requested when DMA is enabled but it will not be processed until the DMA transfer is complete. The DMAR I/O bus signal is used to initiate a DMA transfer. This signal is connected to the calculator through jumper 7. The DMA enable flip-flop is reset by RESET and by any I/O transfer to R6 or R7. When DMA terminates on the last word transferred, the DMA enable is cleared automatically. This will cause the interface to request service at the end of a DMA transfer if interrupt enable is set. Grounding the EIR line will terminate a DMA transfer that is in progress.

DMAR is released very quickly after the DMA I/O cycle begins in order to avoid extraneous DMA requests.

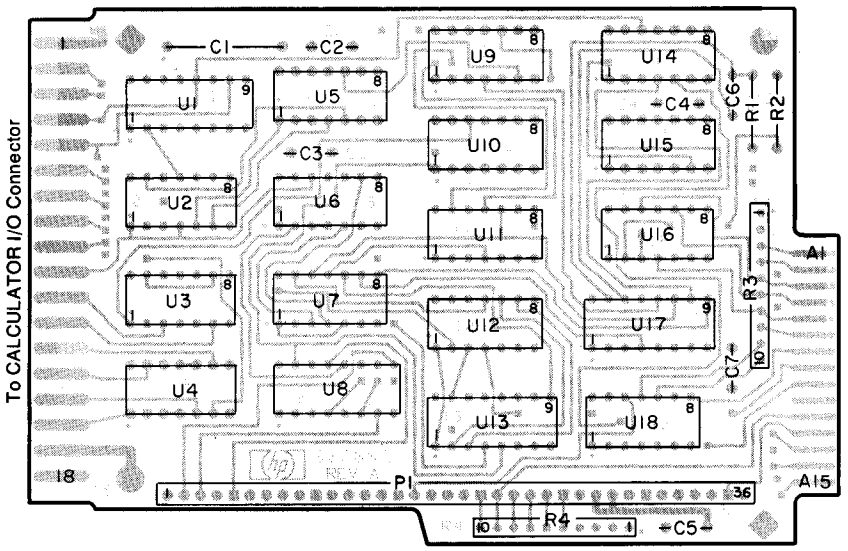
## Logic Symbols



## Table 3-2. Replaceable Parts

| REFERENCE DESIGNATOR | -hp- PART NO. | TQ | DESCRIPTION | MFR. | MFR. PART NO. |
|---|---|---|---|---|---|
| **A1** | **98032-66501** | | **Control Assembly** | | |
| C1 | 0180-0106 | | C: Fxd, 60UF 6V | | |
| C2 - C5 | 0160-4084 | | C: Fxd, .1UF 50V | | |
| C6 | 0140-0149 | | C: Fxd, 470PF 300V | | |
| C7 | 0140-0206 | | C: Fxd, 270PF 300V | | |
| P1 | 1251-4226 | | Conn. 36 Pin | | |
| R1, R2 | 0757-0442 | | R: Fxd, 10k, 1% I/8W | | |
| R3, R4 | 1810-0136 | | R: Fxd - Network | | |
| U1, U13 | 1820-1427 | | IC: 74LS156N | | |
| U2, U15 | 1820-1197 | | IC: 74S00N | | |
| U3 | 1820-1198 | | IC: 74L03N | | |
| U4 | 1820-1297 | | IC: 74S266N | | |
| U5,U11,U14 | 1820-1112 | | IC: 74LS74N | | |
| U6 | 1820-1144 | | IC: 74LS02N | | |
| U7 | 1820-1201 | | IC: 74LS08N | | |
| U8 | 1820-1491 | | IC: 74LS367N | | |
| U9 | 1820-1203 | | IC: 74LS11N | | |
| U10 | 1820-1199 | | IC: 74S04N | | |
| U12 | 1820-1208 | | IC: 74LS32N | | |
| U16 | 1820-1211 | | IC: 74LS86N | | |
| U17 | 1820-1423 | | IC: 74LS123N | | |
| U18 | 1820-0471 | | IC: 7406N | | |
| **A2** | **98032-66502** | | **Configuration Assembly** | | |
| S1 | 3100-3364 | | Switch, Hex | | |
| XA1 | 1251-4148 | | Conn. 2 × 15 Pin | | |
| **A3** | **98032-66503** | | **Data Assembly** | | |
| C1 - C20 | – | | Not Assigned | | |
| C21 | 0160-4084 | | C: Fxd .1UF 50V | | |
| C23 - C25 | 0160-4084 | | C: Fxd .1UF 50V | | |
| C22 | 0180-0106 | | C: Fxd 60UF 6V | | |
| Q1 - Q19 | – | | Not Assigned | | |
| Q20 - Q37 | 1854-0215 | | XSTR: 2N3904 | | |
| R1 - R20 | – | | Not Assigned | | |
| R21, R22 | 0698-4460 | | R: Fxd 649 Ohms 1% | | |
| R23, R24 | 0757-0274 | | R: Fxd 1210 Ohms 1% | | |
| R25,R28,R29 | 1810-0136 | | R: Fxd Network | | |
| R26, R27 | 1810-0037 | | R: Fxd Network | | |
| U1 - U20 | – | | Not Assigned | | |
| U21 | 1820-1411 | | IC: 74S75 | | |
| U22 - U25 | 1820-1296 | | IC: 74S295A | | |
| U26 - U29 | 1820-1562 | | IC: 74C175N | | |
| U30 | 1820-1416 | | IC: 74LS14N | | |
| | **98032-31000** | | **Rear Housing (Standard)** | | |
| | 5040-7803 | | Case - left | | |
| | 5040-7804 | | Case - right | | |
| | 98032-61601 | | Molded Cable & Conn. (std) | | |
| | 5040-7860 | | Molded Cable | | |
| | 1251-4147 | | Conn. 2 × 25 Pin | | |
| | 0590-0663 | | Nut lock 4-40 | | |
| | 2200-0510 | | Screw, Front Housing 4-40 | | |
| | – | | **Front Housing** | | |
| | 5040-7801 | | Case - left | | |
| | 5040-7802 | | Case - right | | |
| | 1480-0292 | | Pin - Dwl .0625 | | |
| | 5040-7836 | | Spring - latch | | |
| | 2200-0536 | | Screw 4-40 | | |
| | 7120-4785 | | Name Plate (std) | | |
| | | | **Miscellaneous** | | |
| | 98241-67932 | | Test Connector | | |
| | 98241-67901 | | Extender Board | | |

# Component Locators

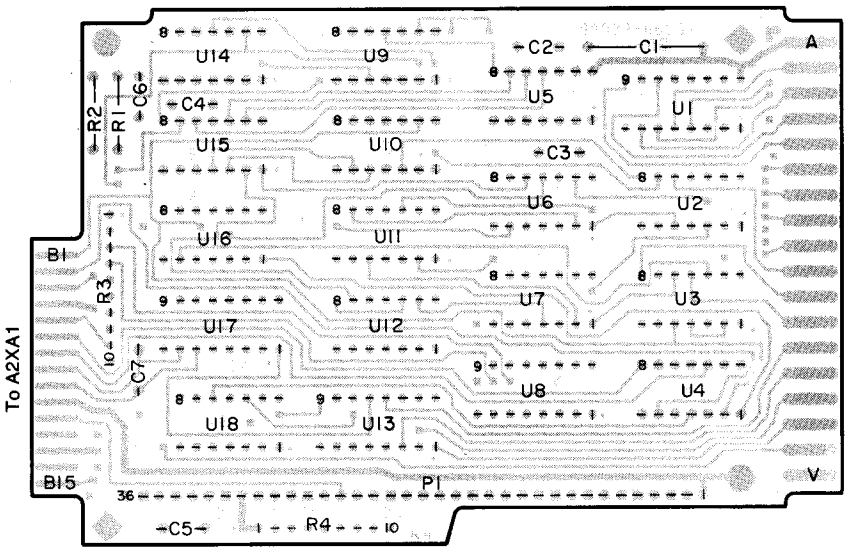



98032A-L-50772

## COMPONENT SIDE

## A1

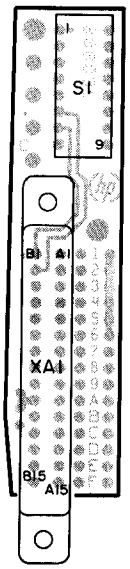
-hp- Part No. 98032—66501 Rev A




## CIRCUIT SIDE

## A1

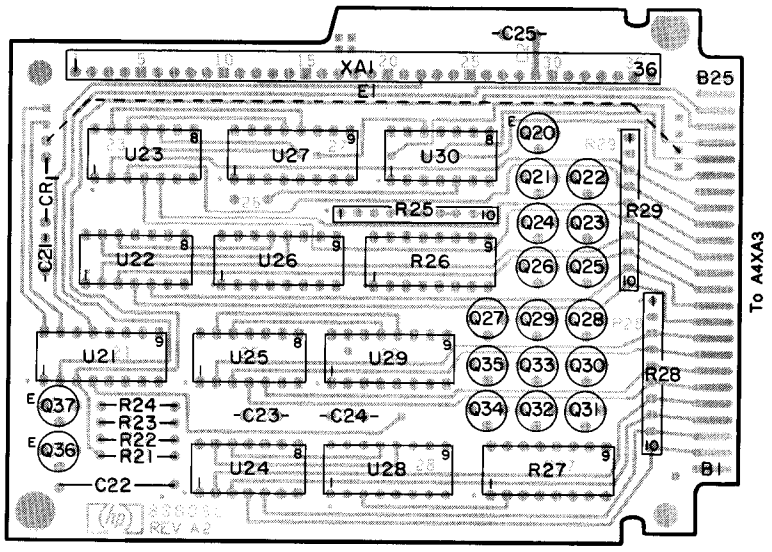
-hp- Part No. 98032—66501 Rev A




## COMPONENT SIDE

## A2

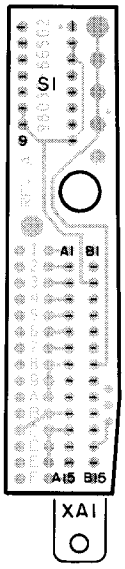
-hp- Part No. 98032-66502 Rev A




## COMPONENT SIDE

## A3

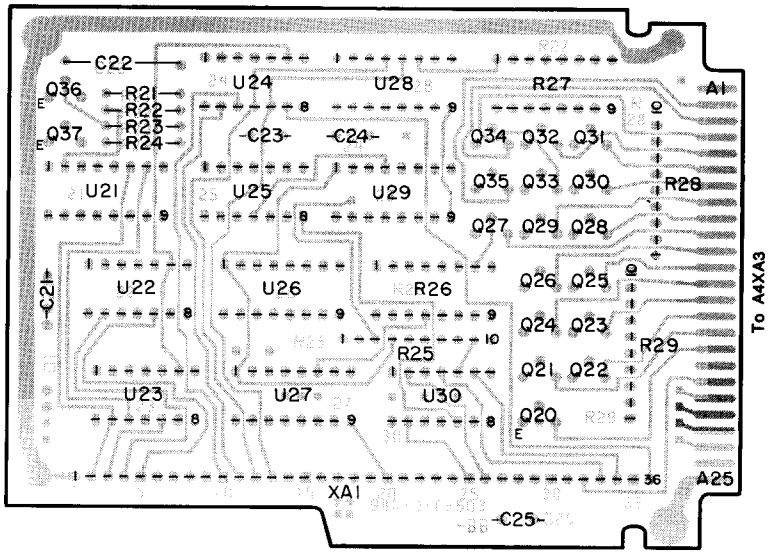
-hp- Part No. 98032-66503 Rev A2




## CIRCUIT SIDE

## A2

-hp- Part No. 98032-66502 Rev A




## CIRCUIT SIDE

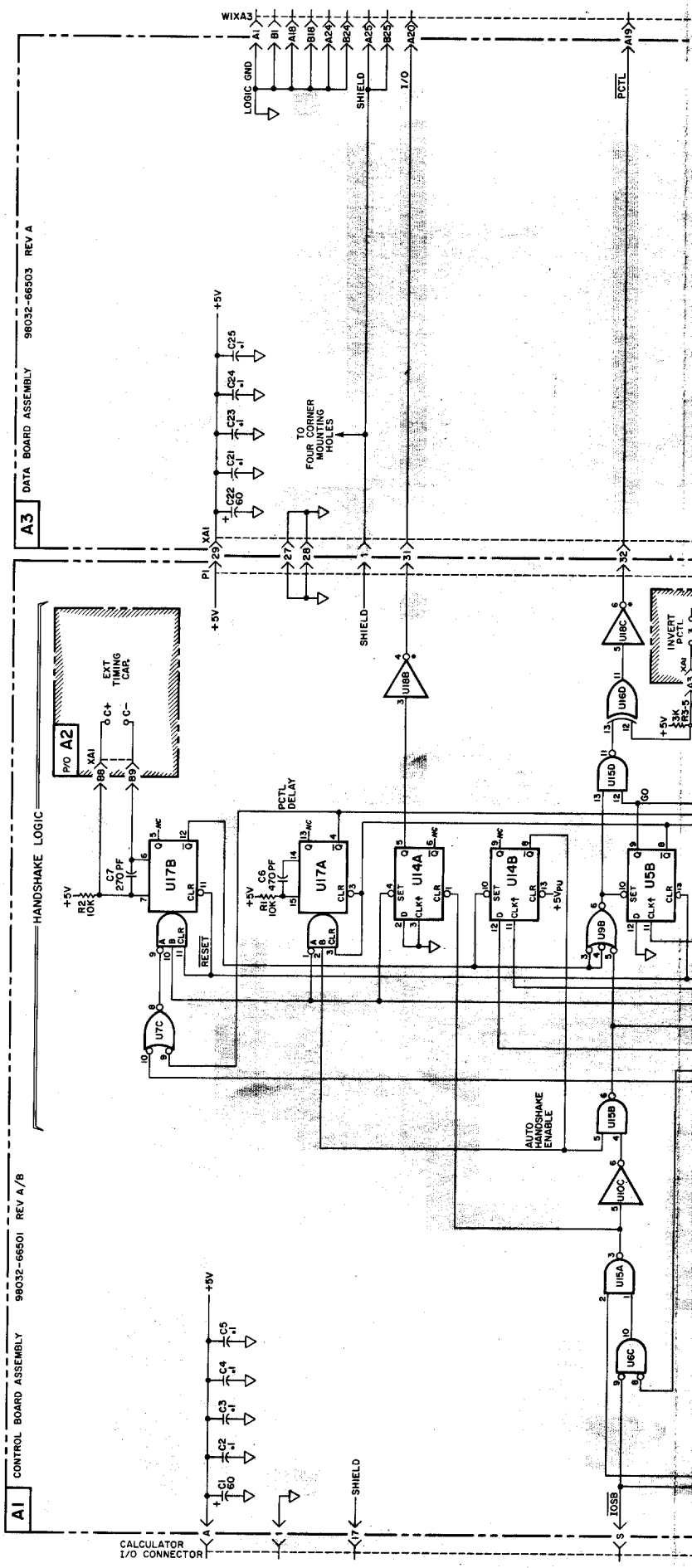
## A3

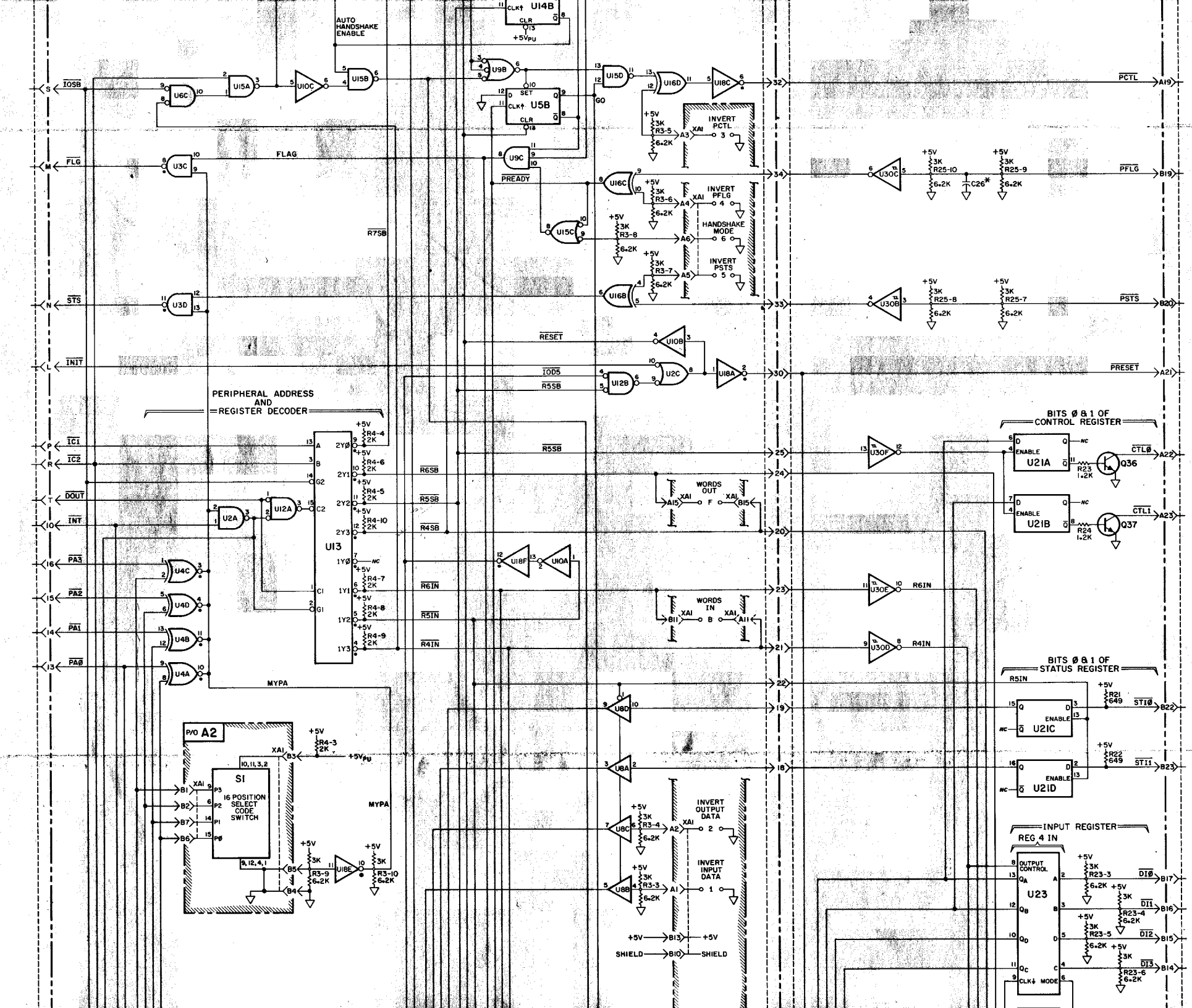
-hp- Part No. 98032-66503 Rev A2

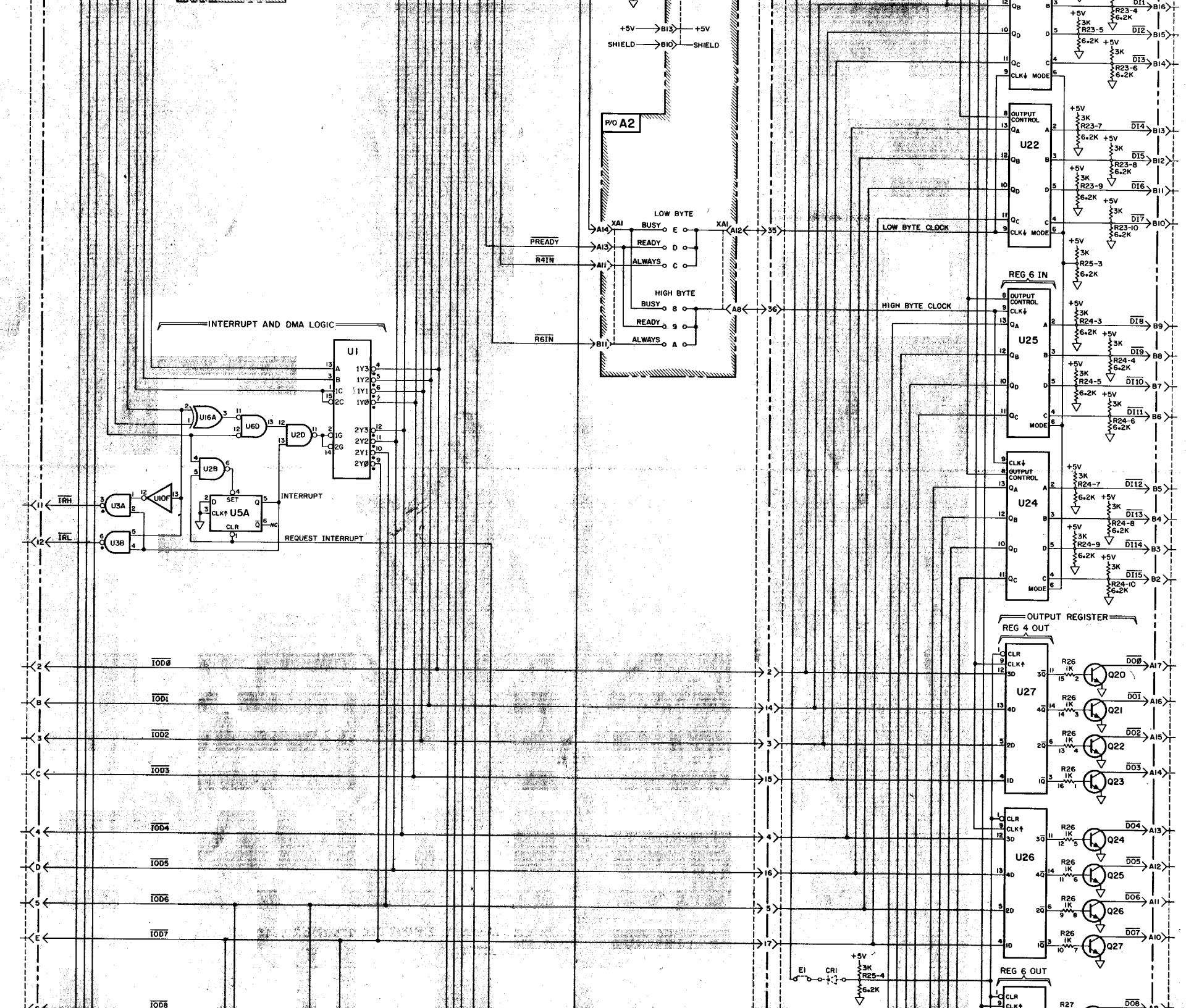
# Standard Interface Cable

| WIRE COLOR | SIGNAL | A | B | SIGNAL | WIRE COLOR |
|---|---|---|---|---|---|
| WH-BK-GN | GND | 1 | 1 | GND | WH-BK-BU |
| WH-YL-GY | DO15 | 2 | 2 | DI15 | WH-RD-BU |
| WH-YL-VIO | DO14 | 3 | 3 | DI14 | WH-RD-GN |
| WH-YL-BU | DO13 | 4 | 4 | DI13 | WH-RD-YL |
| WH-YL-GN | DO12 | 5 | 5 | DI12 | WH-RD-OR |
| WH-OR-VIO | DO11 | 6 | 6 | DI11 | WH-BN-GN |
| WH-OR-BU | DO10 | 7 | 7 | DI10 | WH-GN-YL |
| WH-OR-GN | DO9 | 8 | 8 | DI9 | WH-BN-OR |
| WH-OR-YL | DO8 | 9 | 9 | DI8 | WH-BN-RD |
| WH-VIO | DO7 | 10 | 10 | DI7 | VIO |
| WH-BU | DO6 | 11 | 11 | DI6 | BLU |
| WH-GN | DO5 | 12 | 12 | DI5 | GRN |
| WH-YL | DO4 | 13 | 13 | DI4 | YEL |
| WH-OR | DO3 | 14 | 14 | DI3 | ORG |
| WH-RD | DO2 | 15 | 15 | DI2 | RED |
| WH-BN | DO1 | 16 | 16 | DI1 | BRN |
| WH-BK | DO0 | 17 | 17 | DI0 | BLK |
| WHT | GND | 18 | 18 | DRAIN | (INNER BARE WIRE) |
| WH-GY | PCTL | 19 | 19 | PFLG | GRY |
| WH-BK-BN | I/O | 20 | 20 | PSTS | WH-BK-GY |
| WH-BK-RD | PRESET | 21 | 21 | EIR | WH-BN-GY |
| WH-RD-VIO | CTL0 | 22 | 22 | STI0 | WH-BN-BU |
| WH-RD-GY | CTL1 | 23 | 23 | STI1 | WH-BN-VIO |
| WH-BK-OR | GND | 24 | 24 | GND | WH-BK-YL |
| (OUTER BARE WIRE) | DRAIN | 25 | 25 | NC | |

NOTE 1: U7B pin 4 is connected to the IOSB line on revision A boards and to U15B pin 6 on revision B boards.




AUTO HANDSHAKE ENABLE
U14B
U9B
U5B
U9C
U15C
U16C
U16B
U16D
U15D
U18C
U30C
U30B
U10B
U2C
U18A
U12B
U15A
U10C
U15B
U6C
U3C
U3D
FLAG
PREADY
R7SB
RESET
IOD5
R5SB
INVERT PCTL
INVERT PFLG
HANDSHAKE MODE
INVERT PSTS
PCTL
PFLG
PSTS
PRESET
PERIPHERAL ADDRESS AND REGISTER DECODER
U13
U2A
U12A
U4A
U4B
U4C
U4D
MYPA
R6SB
R5SB
R4SB
R6IN
R5IN
R4IN
WORDS OUT
WORDS IN
U18F
U10A
U30F
U30E
U30D
BITS Ø & 1 OF CONTROL REGISTER
U21A
U21B
Q36
Q37
CTLØ
CTL1
BITS Ø & 1 OF STATUS REGISTER
U21C
U21D
STIØ
STI1
INPUT REGISTER
REG 4 IN
U23
DIØ
DI1
DI2
DI3
P/O A2
S1
16 POSITION SELECT CODE SWITCH
U8D
U8A
U8C
U8B
U18E
INVERT OUTPUT DATA
INVERT INPUT DATA
SHIELD
GO
IOSB
INT
DOUT
IC1
IC2
PA3
PA2
PA1
PA0
+5V
3K
6.2K
C26*


P/O A2
LOW BYTE
HIGH BYTE
BUSY
READY
ALWAYS
PREADY
R4IN
R6IN
LOW BYTE CLOCK
HIGH BYTE CLOCK
INTERRUPT AND DMA LOGIC
U1
U16A
U6D
U2D
U2B
U3A
U3B
U10F
U5A
INTERRUPT
REQUEST INTERRUPT
REG 6 IN
U22
U25
U24
OUTPUT REGISTER
REG 4 OUT
U27
U26
REG 6 OUT
Q20
Q21
Q22
Q23
Q24
Q25
Q26
Q27
+5V
3K
6.2K
SHIELD

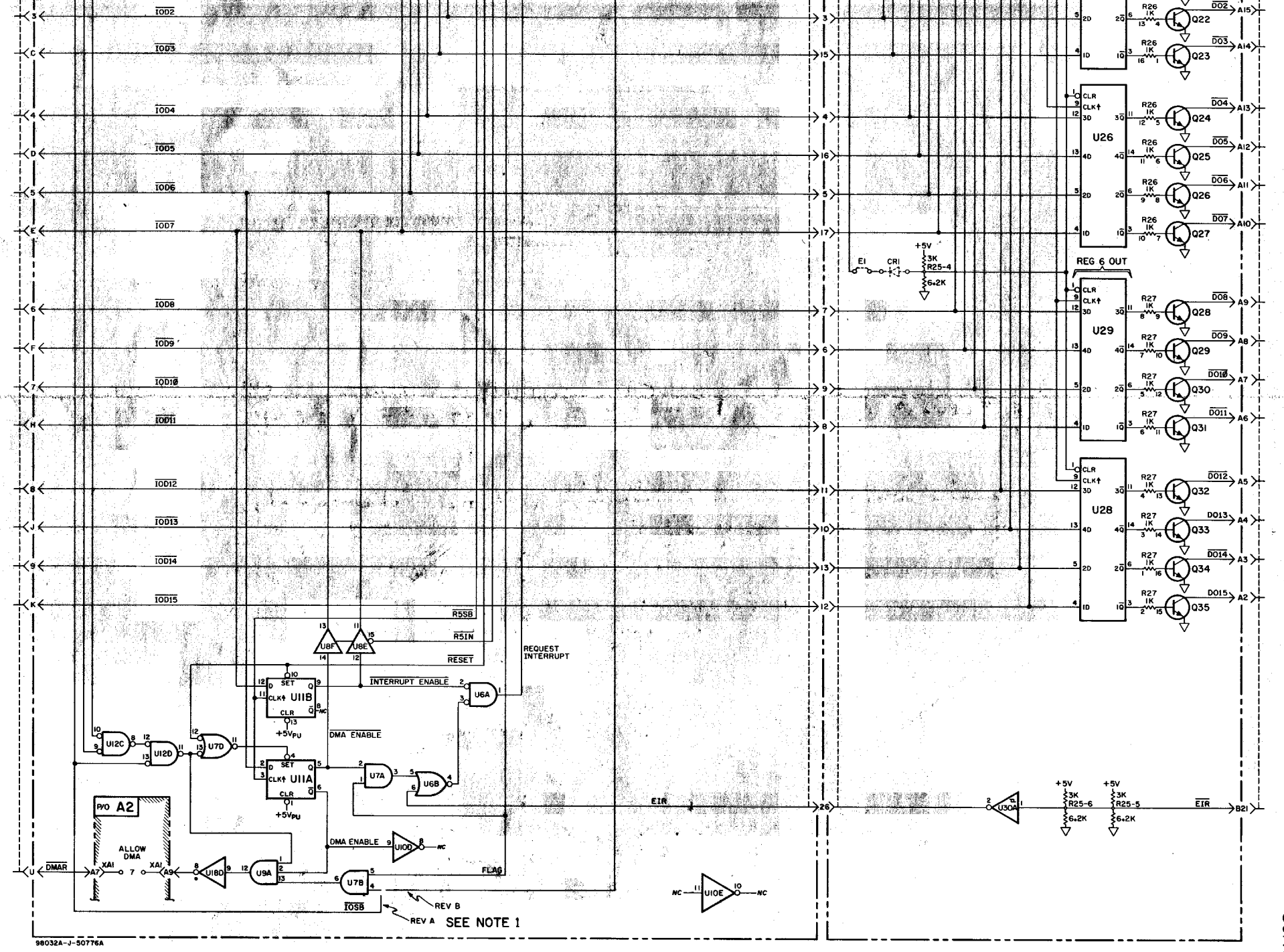
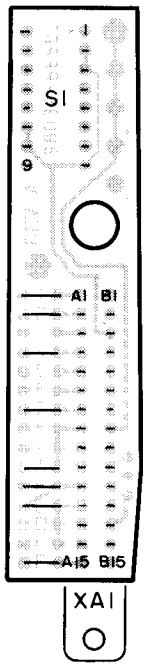



Figure 3-4. 98032A Interface Schematic

# Appendix A
# Test Connector

98032A-L-50772




Circuit Side

Test Connector Configuration Jumpers

Test Connector Input/Output Jumpers

| SIGNAL | A | | B | SIGNAL |
|---|---|---|---|---|
| GND | 1 | o—o | 1 | GND |
| DO15 | 2 | o—o | 2 | DI15 |
| DO14 | 3 | o—o | 3 | DI14 |
| DO13 | 4 | o—o | 4 | DI13 |
| DO12 | 5 | o—o | 5 | DI12 |
| DO11 | 6 | o—o | 6 | DI11 |
| DO10 | 7 | o—o | 7 | DI10 |
| DO9 | 8 | o—o | 8 | DI9 |
| DO8 | 9 | o—o | 9 | DI8 |
| DO7 | 10 | o—o | 10 | DI7 |
| DO6 | 11 | o—o | 11 | DI6 |
| DO5 | 12 | o—o | 12 | DI5 |
| DO4 | 13 | o—o | 13 | DI4 |
| DO3 | 14 | o—o | 14 | DI3 |
| DO2 | 15 | o—o | 15 | DI2 |
| DO1 | 16 | o—o | 16 | DI1 |
| DO0 | 17 | o—o | 17 | DI0 |
| GND | 18 | o—o | 18 | DRAIN |
| PCTL | 19 | o—o | 19 | PFLG |
| I/O | 20 | o | 20 | PSTS |
| PRESET | 21 | o—o | 21 | EIR |
| CTL0 | 22 | o—o | 22 | STI0 |
| CTL1 | 23 | o—o | 23 | STI1 |
| GND | 24 | o—o | 24 | GND |
| DRAIN | 25 | o—o | 25 | NC |

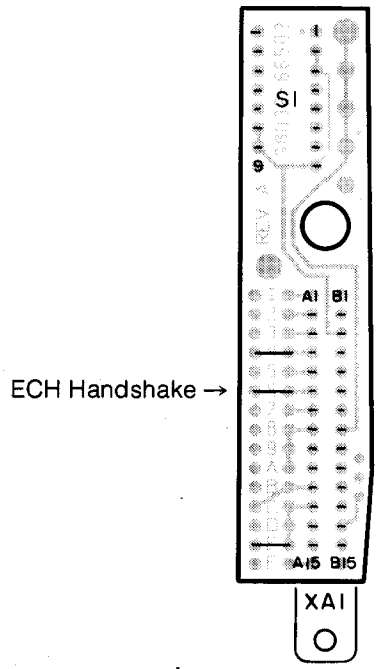
# Appendix B
# Test Program

The following 9825A Calculator Interface Test Program can be used to test the operation of the 98032A Interface. This program requires the use of the General I/O ROM.

```
0: prt "98032A
 TEST";spc 2
1: 0→C
2: "t1":
3: dsp "Doing
 DATA PATTERNS
 TEST"
4: -32768+int(rn
 d(1)2*32767)→X
5: wtb 2,X;rdb(2
 )→Y;if X#Y;prt
 "ERROR","Correc
 t",X,"incorrect
 ",Y;spc 2;stp
6: if C/100=int(
 C/100);beep
7: if (C+1→C)<10
 00;gto "t1"
8: 0→C
9: "t3":
10: dsp "Doing
 EXTENDED CTRL
 and STATUS"
11: int(rnd(1)*
 3)→X
12: wtc 2,X;rds(
 2)mod4→Y;if
 X=Y;jmp 2
13: prt "ERROR",
 "Correct",X,
 "incorrect",Y;
 spc 2;stp
14: if C/100=int
 (C/100);beep
15: if (C+1→C)<1
 000;gto "t3"
16: prt "TEST
 COMPLETE";spc
 2;dsp "Test of
 98032 COMPLETE!
 ";stp ;gto 0
17: end
*8538
```

# Appendix C
# Jumpers to Simulate HP 11202A

To simulate the HP 11202A Interface operation (interface used with HP 9830, 9821, 9820, and 9810 Calculators) install configuration jumpers 4 and E on the 98032A. If ECH handshake is to be simulated, also install configuration jumper 6.




Circuit Side

Scan Copyright ©
The Museum of HP Calculators
www.hpmuseum.org

Original content used with permission.

Thank you for supporting the Museum of HP Calculators by purchasing this Scan!

Please to not make copies of this scan or make it available on file sharing services.