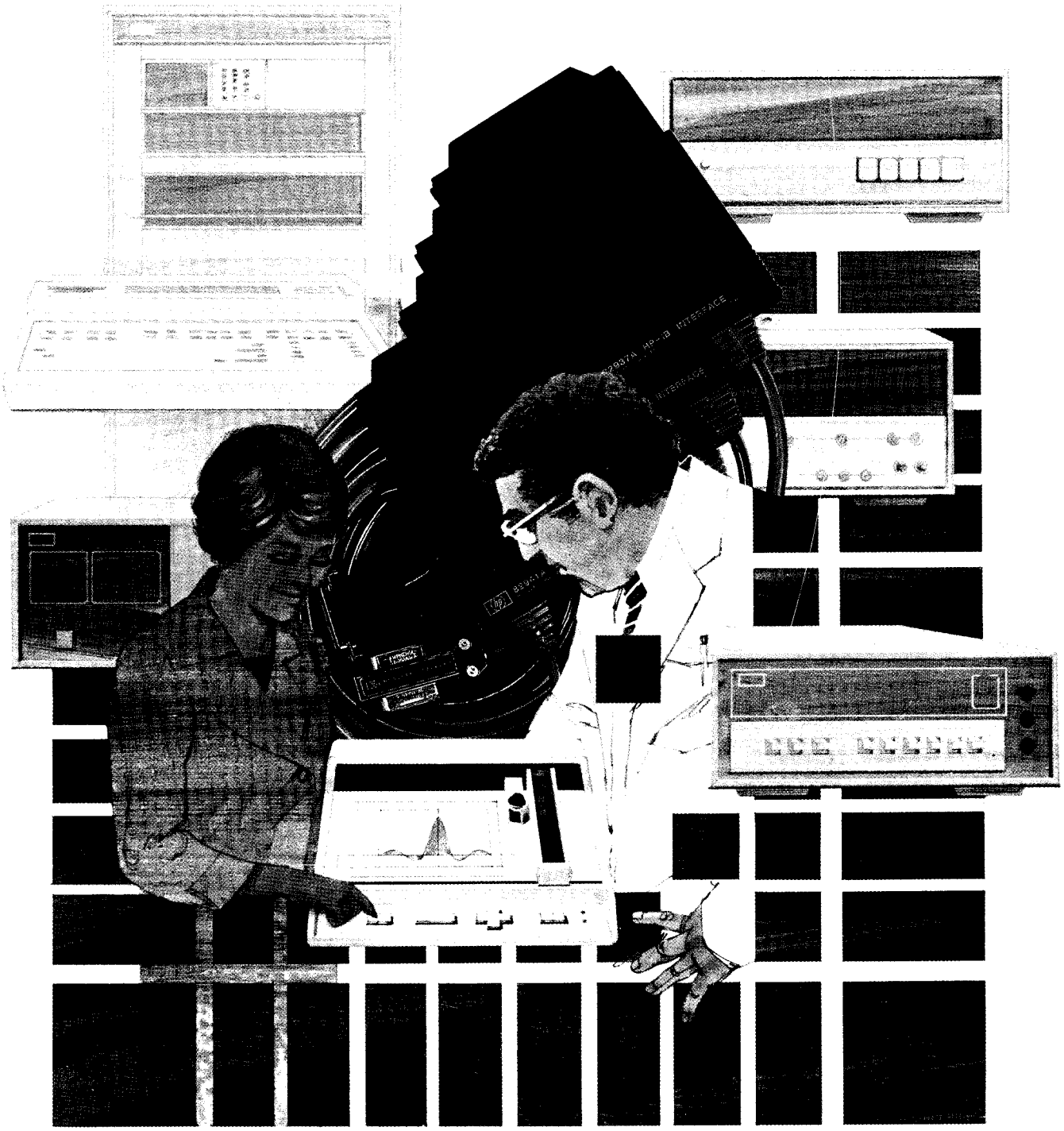
HEWLETT-PACKARD

HP 82939A

# Serial Interface

## OWNER'S MANUAL

## SERIES 80



HEWLETT PACKARD

# HP 82939A
# Serial Interface
# Owner's Manual
# Series 80

March 1982

Reorder Number
82939-90012

Printed in U.S.A.

©Hewlett-Packard Company 1982

# Contents

Section 1

# General Information

## Serial Interface Description

The HP 82939A Serial Interface allows the HP Series 80 Personal Computer to communicate with serial asynchronous data communication devices. The HP 82939A Serial Interface is equipped with both RS-232C and 20 mA current loop drivers and receivers for the data transmitter and receiver. Data can be transferred at standard baud rates of 50 to 9,600. A universal asynchronous receiver/transmitter (UART) integrated circuit is used to manipulate data and provide the basic hardware protocol for asynchronous operation. The interface is available in cable configurations which allow the computer to act as either a data terminal or the digital portion of a modem.

## RS-232C

One of the standards adopted by the Electronics Industries Association (EIA) for use with data communications was the RS-232C, Interface Between Data Terminal Equipment and Data Communications Equipment Employing Serial Binary Data Interchange. The RS-232C is the most common data communications protocol in use today. The International Telegraph and Telephone Consultative Committee, or CCITT, Standards V.24 and V.28, parallel the EIA RS-232C and provide a compatible international standard. RS-232C has been the most successful interfacing standard implemented to date.

A description of the RS-232C standard is contained in Appendix A. Appendix B contains a table of RS-232C functions, listed by pin number.

## I/O ROM

The HP 82939A Serial Interface requires the use of either the I/O ROM (HP part no. 00085-15003 for the HP-85/83 or 00087-15003 for the HP-87) or the Plotter/Printer ROM (HP part no. 00085-15002 for the HP-85/83 or 00087-15002 for the HP-87). The ROM fits into a ROM Drawer which plugs into a port in the rear of the computer.

## Specifications

The pertinent environmental specifications for the serial interface are listed in the chart below. For details of the electrical specifications, see the section entitled "Electrical Characteristics" in Appendix A of this manual. Power for the interface is supplied by the computer through the interface port.

| | |
|---|---|
| **Operating Temperature** | 0° C - 55° C (32° F - 131° F) |
| **Humidity** | 0 - 80% |
| **Dimensions** | Approximately 16.7 × 12.7 × 1.5 cm (6.59 × 5 × 0.59 in). |
| **Weight** | 0.5 kg (1.1 lb) |

## Cable Options

The interface must be used with one of the available option cables. The option number determines which cable is provided with the interface. Table 1-1 describes which equipment is included with the available options.

**Table 1-1. Cable Options**

| Option | Connector | Purpose |
|---|---|---|
| 000 (Standard) | 25 Pin Female RS-232C | For connection to data terminal equipment. |
| 001 | 25 Pin Male RS-232C | For connection to a modem or to other data communications equipment. |
| 002 | None | For connection to 20 mA current loop equipment. |

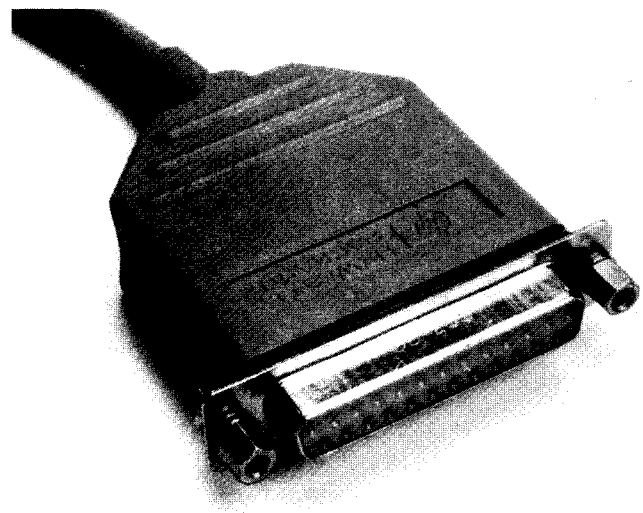
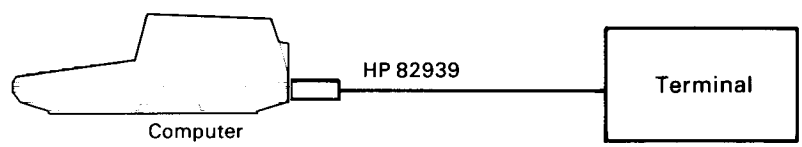


**Figure 1-1. Standard Cable Connector**

Using the standard interface cable, the computer takes the place of a modem or computer communicating with a terminal.




HP 82939A Standard

**Figure 1-2. HP Series 80/Terminal Communication (Standard)**

The option 001 interface cable is shipped with a 2-metre (6.5 ft) cable terminated with a standard male EIA 25-pin connector. The option 001 interface is connected between the computer and modem.

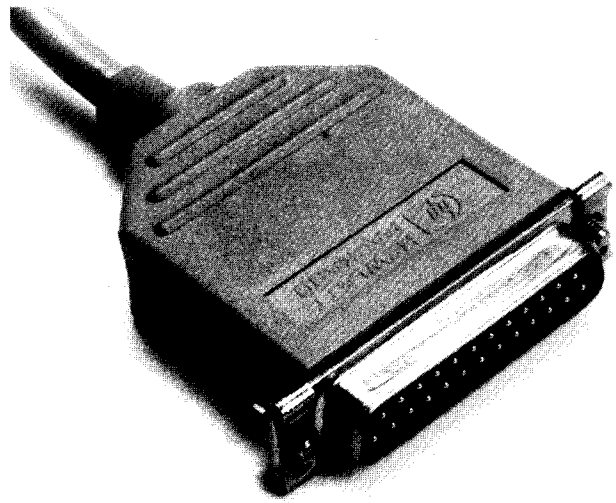
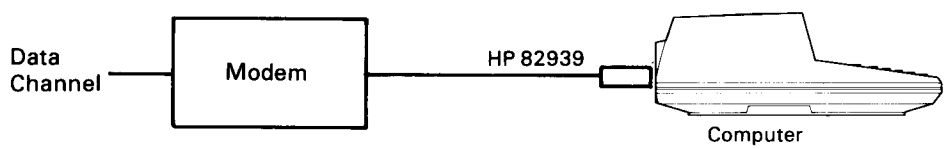


**Figure 1-3. Option 001 Cable Connector**

Using the option 001 interface cable, the computer acts as a data terminal connected to a modem.




HP 82939A Option 001

**Figure 1-4. HP Series 80/Modem Communication (Option 001)**

Figure 1-5 shows a typical point-to-point installation using the HP 82939A Serial Interface. Note that one computer uses the standard connector, while the other uses option 001.

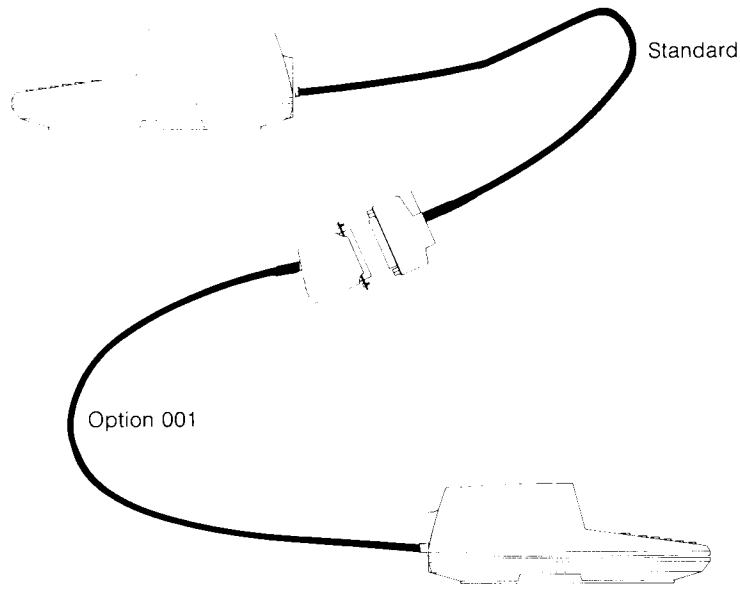



**Figure 1-5. Point-to-Point System**

Figure 1-6 shows a typical installation using modems to convey information over commercial telephone lines. Both computers in this example use the standard connector.

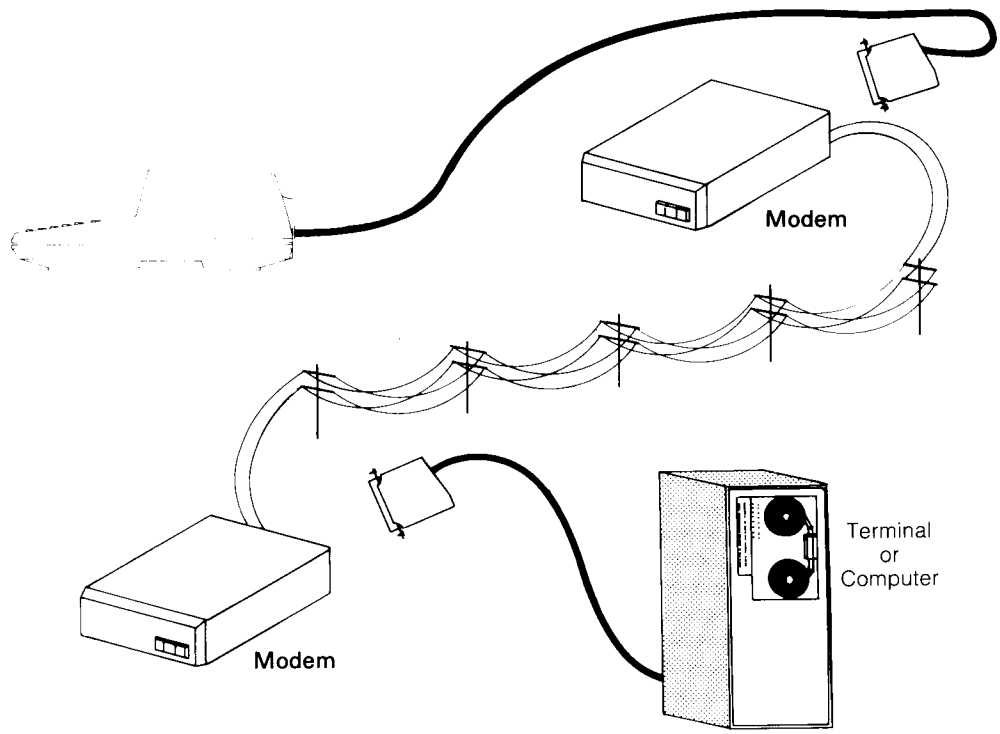



**Figure 1-6. Typical Data Communications System With Modems**

**Table 1-1. Cable Signals**

**Standard Cable**
**(Female Conector Connects to Terminal)**

| Pin No. | Signal Name | Signal Direction |
|---|---|---|
| 1 | Protective Ground | N/A |
| 2 | Transmitted Data | Device to Interface |
| 3 | Received Data | Interface to Device |
| 4 | Request to Send | Device to Interface |
| 5 | Clear to Send | Interface to Device |
| 6 | Data Set Ready | Interface to Device |
| 7 | Signal Ground | N/A |
| 8 | Carrier Detect | Interface to Device |
| 9 – 11 | Unused | |
| 12 | Unassigned | |
| 13 – 18 | Unused | |
| 19 | Unassigned | |
| 20 | Data Terminal Ready | Device to Interface |
| 21 – 22 | Unused | |
| 23 | Data Rate Select | Device to Interface |
| 24 – 25 | Unused | |

**Option 001 Cable**
**(Male Connector Connects to Modem)**

| Pin No. | Signal Name | Signal Direction |
|---|---|---|
| 1 | Protective Ground | N/A |
| 2 | Transmitted Data | Interface to Device |
| 3 | Received Data | Device to Interface |
| 4 | Request to Send | Interface to Device |
| 5 | Clear to Send | Device to Interface |
| 6 | Data Set Ready | Device to Interface |
| 7 | Signal Ground | N/A |
| 8 | Carrier Detect | Device to Interface |
| 9 – 11 | Unused | |
| 12 | Unassigned | |
| 13 – 18 | Unused | |
| 19 | Unassigned | |
| 20 | Data Terminal Ready | Interface to Device |
| 21 – 22 | Unused | |
| 23 | Data Rate Select | Interface to Device |
| 24 – 25 | Unused | |

**Option 002 Cable**
**(Unterminated Cable Connects to Current Loop Device)**

| Wire Color | Signal Name | Signal Direction |
|---|---|---|
| Shield | Protective Ground | N/A |
| Green | Logic Ground | N/A |
| Black | + CL Received Data | Device to Interface |
| Red | – CL Received Data | Interface to Device |
| White | – CL Transmitted Data | Interface to Device |

Section 2

# Installation

## Unpacking and Inspection

If the shipping carton is damaged, ask the carrier's agent to be present when the interface is unpacked. If the interface is damaged or fails to meet electrical specifications, immediately notify the carrier and the nearest HP sales and service office. Retain the shipping carton for the carrier's inspection. The sales and service office will arrange for the repair or replacement of your interface without waiting for the claim against the carrier to be settled.

## Conditions Set by User

Many conditions are determined by the position of switches set by the user. Table 2-1 contains a list of these conditions, the choices allowed by the switches and the factory settings of each.

**Table 2-1. Conditions Set by User**

| Function | Choices | Factory Setting |
|---|---|---|
| Select code | Three switches allow choice of eight codes. | 10 (111) |
| Baud Rate | Four switches select one of 16 baud rates. | 300 (0110) |
| Character Word Length | Two switches select among 5-, 6-, 7-, or 8-bit word length. | 7 (10) |
| Number of Stop Bits | One switch chooses one or two stop bits. | 1 (0) |
| Parity Enable | One switch to enable parity. | Enable (1) |
| Parity Select | Select odd or even parity. | Odd (0) |
| Parity Stick | One switch to make parity bit always 0 or always 1, according to parity select switch. | No (0) |
| Autohandshake | One switch to recognize or ignore "Clear to Send" and "Carrier Detect". | Off (0) |

Note that all of these switches except select code may be overridden by software, once the interface is in operation.

Since these switches are located inside the interface housing, changing any of them requires disassembly of the housing. If this is done, refer to the following disassembly procedure to gain access to the switches.

After the switches have been changed, their position can be determined without disassembling the interface by reading the appropriate status register (see table 3-1).

## Disassembly

Refer to figure 2-1 to see how the interface parts fit together. Place the interface on a flat surface with the side having the seven screws facing upward and the cable coming out to the left. Then use the following steps to disassemble the interface.

1. Using a #1 Pozidriv® screwdriver, remove the seven screws and set them aside.
2. Hold the interface parts together and turn it over so that the seven screw holes are facing downward and the cable is still coming out to the left.
3. Hold the cable strain relief in place and remove the top half of the interface housing.

If you have followed the above steps, switch S1 should be oriented as shown in the following figures. If it isn't, orient it as illustrated before identifying the switch segments.

When you re-assemble the interface, reverse the above procedure, making sure the ground clip is in place. The ground clip should be under the circuit board when the component side is up.

## Switch Settings

As previously noted, many interface functions are switch selectable, and preset at the factory. Switch blocks S1 and S2 are used for this purpose.

Once the interface is disassembled, any switch settings may be verified or changed by referring to the appropriate section of the following material.

They may have either slide or rocker switches. Both types are illustrated in the factory preset positions.

> **Note:** If you change any of the factory settings, make sure that you change the proper switch segments. Do not disturb the settings of adjoining switches. Use a pencil point or similar object for changing switch settings.

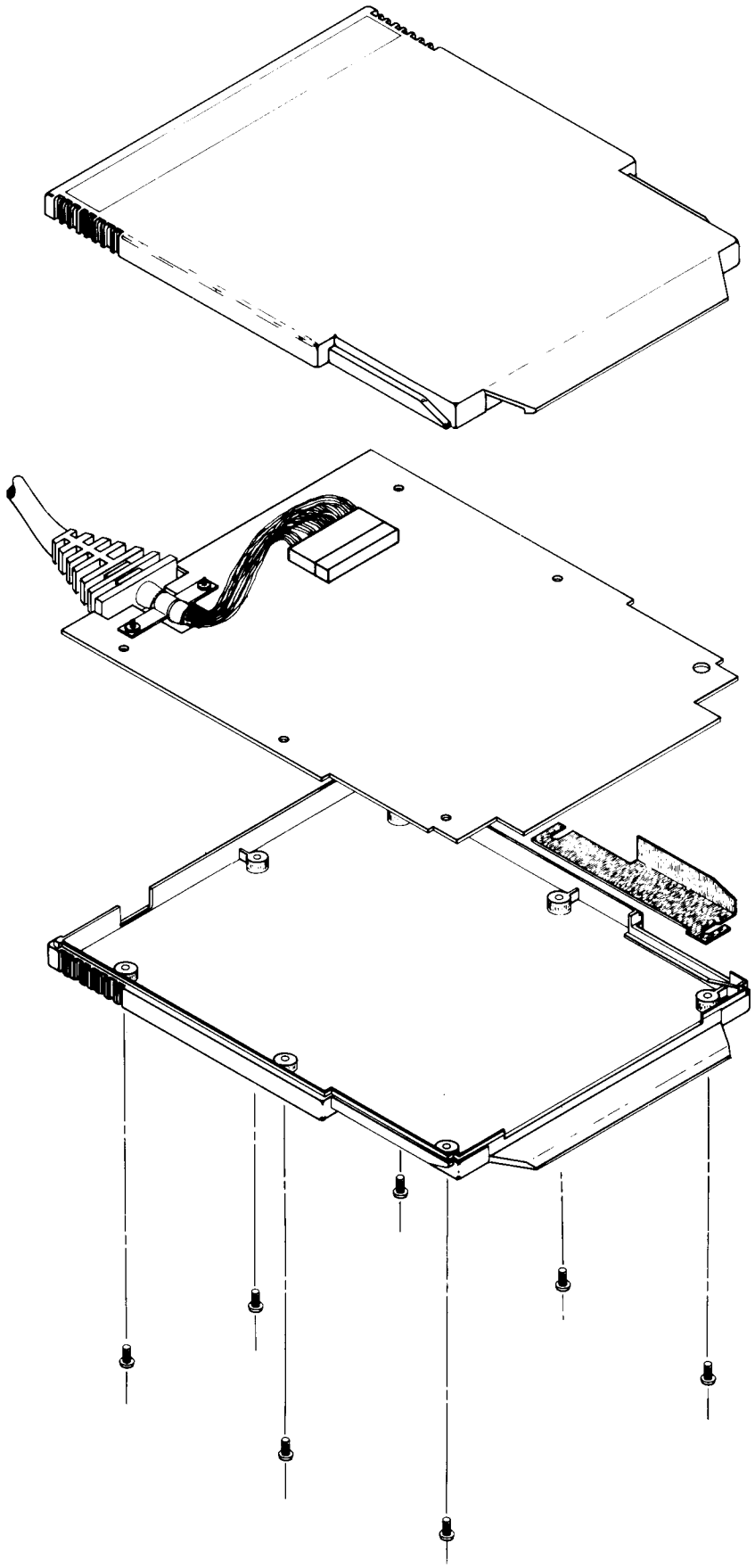
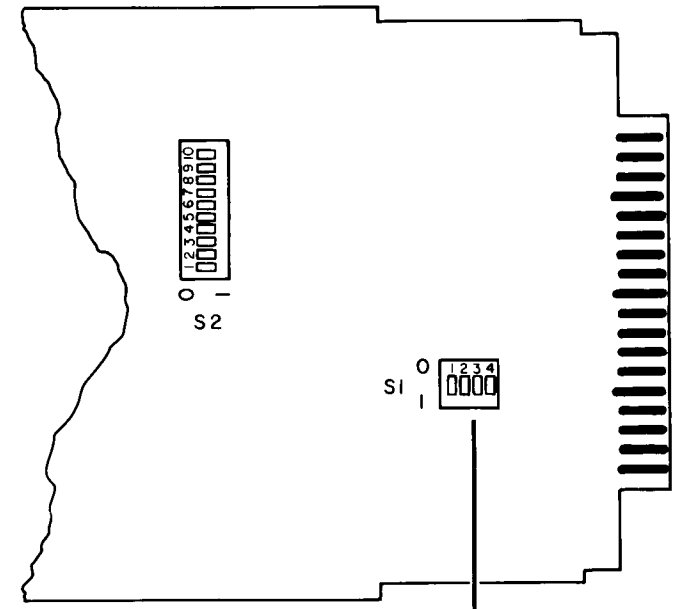


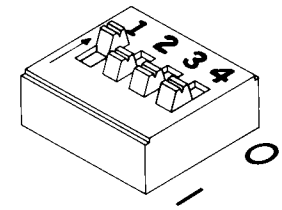
**Figure 2-1. Disassembly**



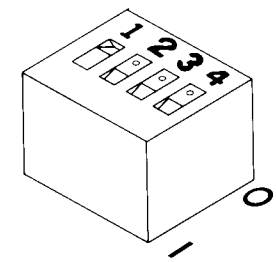

**Switch 1**

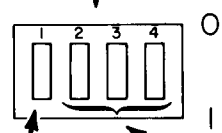
| | | |
|---|---|---|
| Segment 1: | AHS | Autohandshake |
| Segment 2: | SC2 | Select code 2 |
| Segment 3: | SC1 | Select code 1 |
| Segment 4: | SC0 | Select code 0 |

**Preset Positions**




**OR**

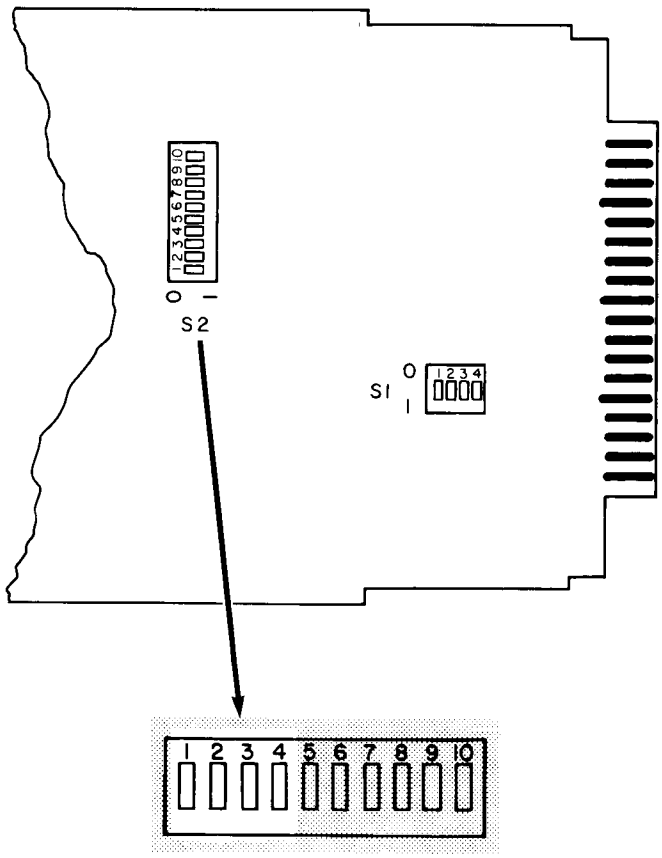



| Autohandshake | S1(1) |
|---|---|
| PRESET → Ignore | 0 |
| Enable | 1 |

| Select Code Bit | S1(2) | S1(3) | S1(4) |
|---|---|---|---|
| 3 | 0 | 0 | 0 |
| 4 | 0 | 0 | 1 |
| 5 | 0 | 1 | 0 |
| 6 | 0 | 1 | 1 |
| 7 | 1 | 0 | 0 |
| 8 | 1 | 0 | 1 |
| 9 | 1 | 1 | 0 |
| PRESET → 10 | 1 | 1 | 1 |

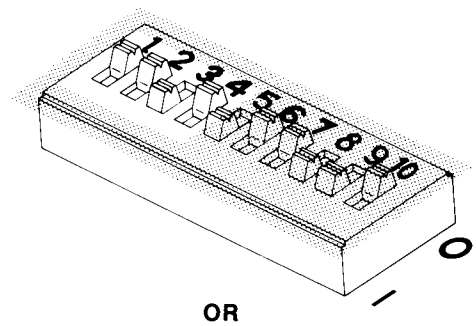
**Figure 2-2. Auto Handshake and Select Code Switches**



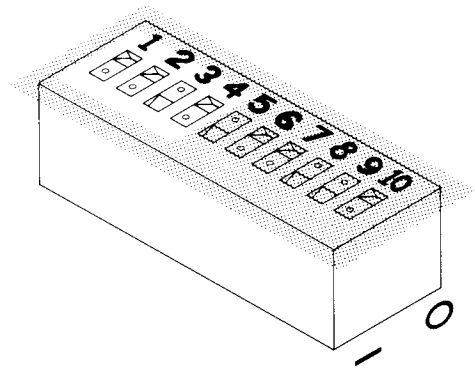

**Switch 2**

| | | |
|---|---|---|
| Segment 1: | SP | Parity transmit |
| Segment 2: | EPS | Select parity |
| Segment 3: | PEN | Parity enable |
| Segment 4: | STB | Number of stop bits |

**Preset Positions**




**OR**



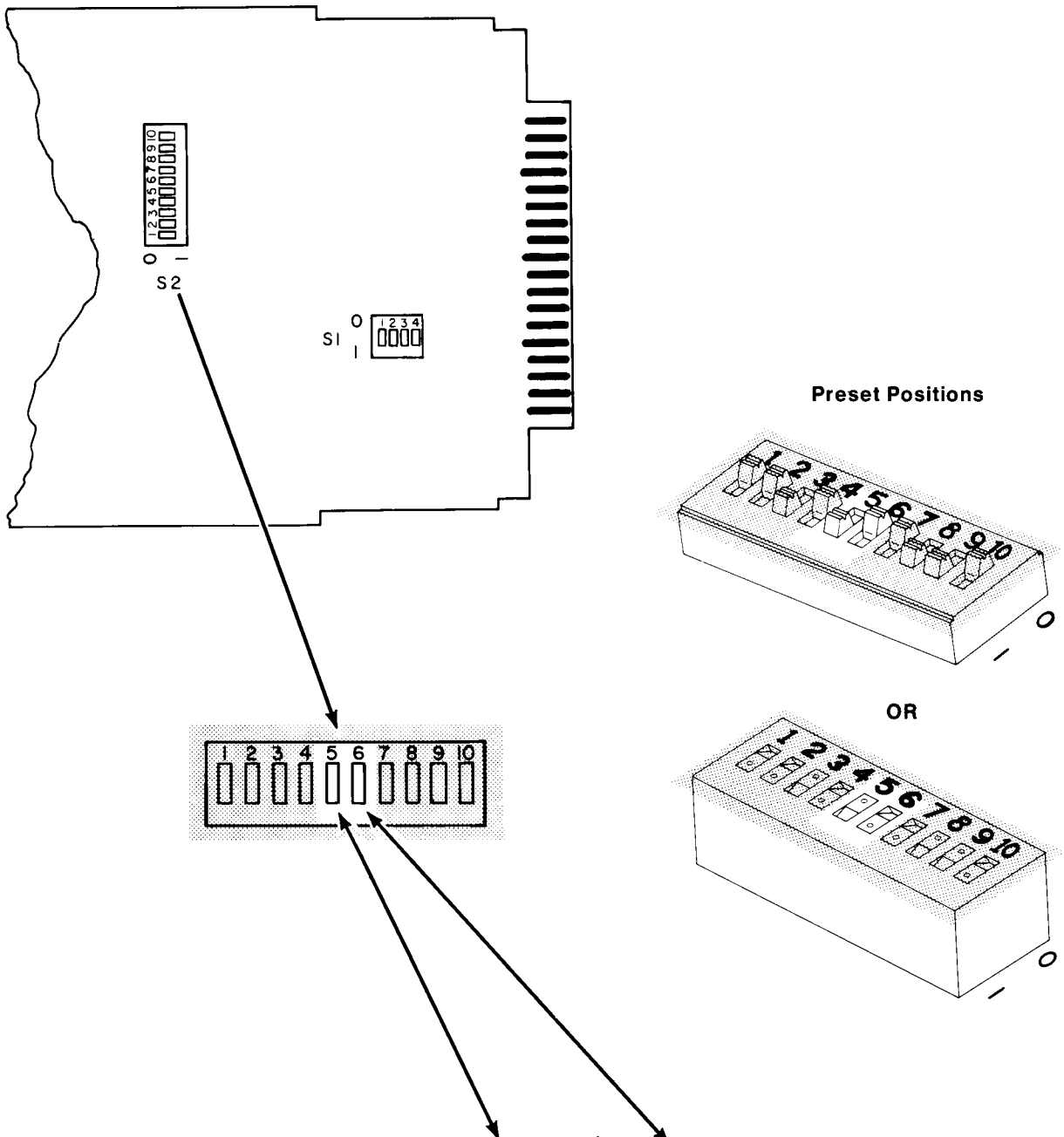

| | Parity Transmit | Segment 1 |
|---|---|---|
| PRESET → | No | 0 |
| | Yes | 1 |

| | Select Parity | Segment 2 |
|---|---|---|
| PRESET → | Odd | 0 |
| | Even | 1 |

| | Parity Enable | Segment 3 |
|---|---|---|
| | Disable | 0 |
| PRESET → | Enable | 1 |

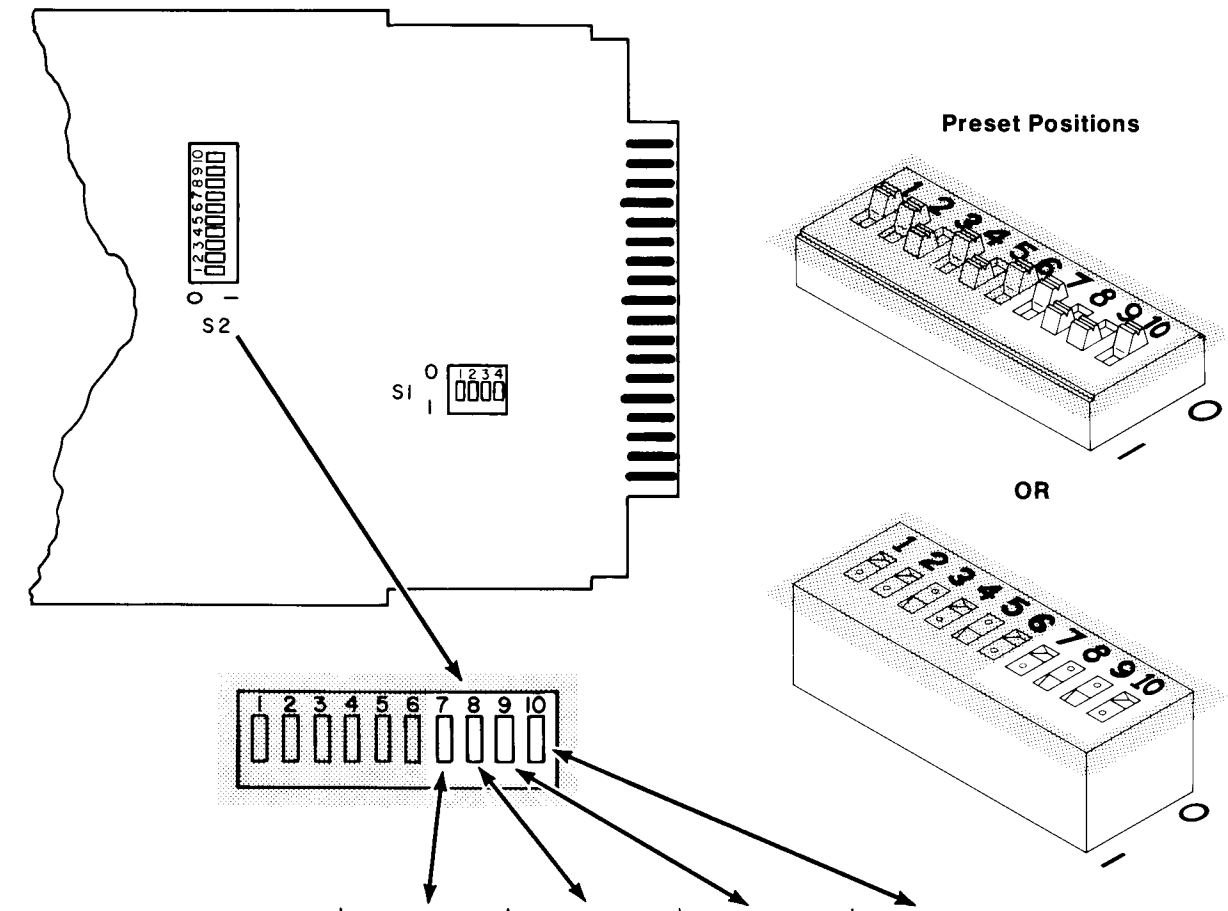
| | Number of Stop Bits | Segment 4 |
|---|---|---|
| PRESET → | 1 | 0 |
| | 2 | 1 |

**Figure 2-3. Parity and Stop Bit Switches**




| Character Word Length | Segment 5 | Segment 6 |
|---|---|---|
| 5 | 0 | 0 |
| 6 | 0 | 1 |
| PRESET → 7 | 1 | 0 |
| 8 | 1 | 1 |

**Figure 2-4. Character Word Length Switches**




| Baud Rate | Segment 7 | Segment 8 | Segment 9 | Segment 10 |
|---|---|---|---|---|
| 50 | 0 | 0 | 0 | 0 |
| 75 | 0 | 0 | 0 | 1 |
| 110 | 0 | 0 | 1 | 0 |
| 134.5 | 0 | 0 | 1 | 1 |
| 150 | 0 | 1 | 0 | 0 |
| 200 | 0 | 1 | 0 | 1 |
| PRESET → 300 | 0 | 1 | 1 | 0 |
| 600 | 0 | 1 | 1 | 1 |
| 1,200 | 1 | 0 | 0 | 0 |
| 1,800 | 1 | 0 | 0 | 1 |
| 2,000 | 1 | 0 | 1 | 0 |
| 2,400 | 1 | 0 | 1 | 1 |
| 3,600 | 1 | 1 | 0 | 0 |
| 4,800 | 1 | 1 | 0 | 1 |
| 7,200 | 1 | 1 | 1 | 0 |
| 9,600 | 1 | 1 | 1 | 1 |

**Figure 2-5. Baud Rate Switches**

## Connecting to the Computer

> **CAUTION**
>
> Before connecting this interface (or any interface), make sure the computer is turned OFF. Installing with power on can damage both the interface and the mainframe. In addition, the mainframe goes through an initialization routine at power-on which checks for presence of the interface, and does not recognize the interface if installed after the initialization routine.

When connecting the HP 82939A Serial Interface to an HP Series 80 Personal Computer, follow this procedure.

1. Set the select code switches. The select code is set to 10 at the factory (see the preceding pages, from Disassembly).
2. Set the default condition switches and reassemble the interface module.
3. Insert the interface into any of the four I/O ports in the back of the mainframe with the computer turned OFF! Make sure the interface is firmly seated in the port (see figure 2-6).
4. Connect the interface cable to an applicable data communications device and turn the peripheral device and the computer ON.

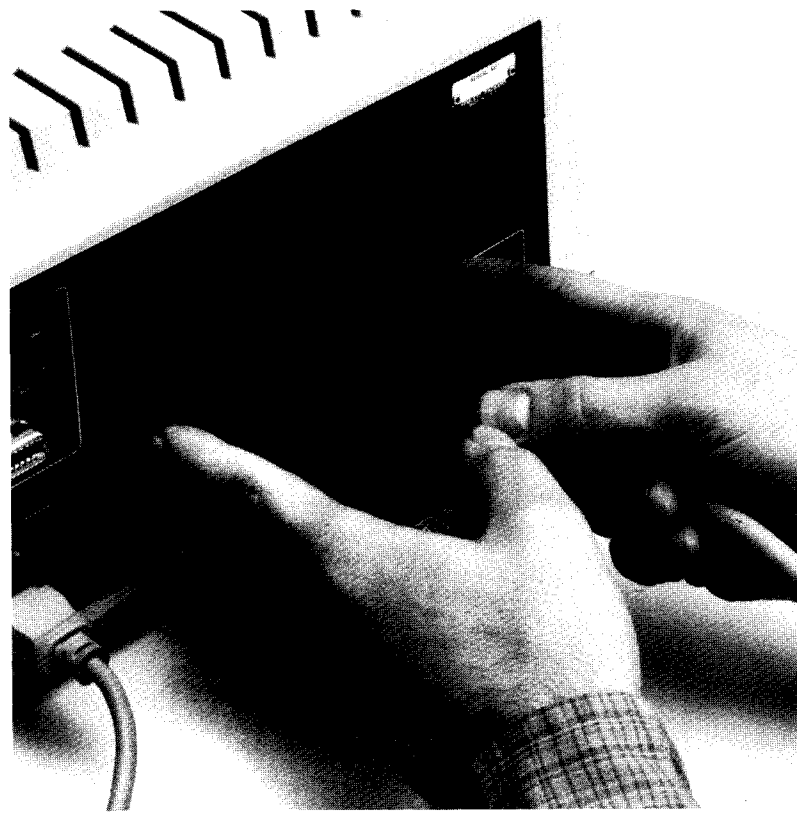


**Figure 2-6. Installing the Interface Into the Backplane**

## 20 mA Current Loop

RS-232C supplies are ±12 V. This gives a ±10V drive capability.

Each of the 20 mA current supplies can supply 20 mA at up to a 9.9V compliance (−20 mA into a 495 ohm load). The load impedance should not drop below 495 ohms. The floating detector uses about 1.9V, and the floating switch uses about 0.2V. Thus, in full-duplex mode (interface connected to a teletype) there is a 9.7V left to the teletype receiver/detector and 8V left to the teletype transmitter switch. This is more than adequate to ensure trouble-free connection to all standard 20 mA devices with high noise immunity.

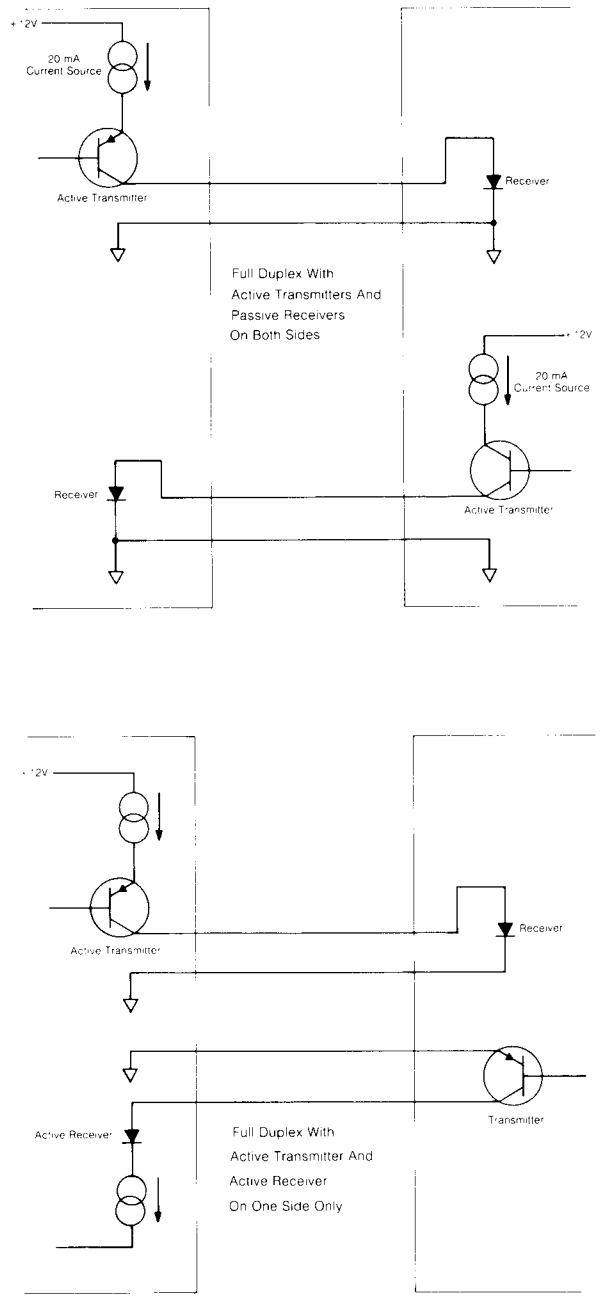



**Figure 2-7. Recommended Current Loop Circuits**

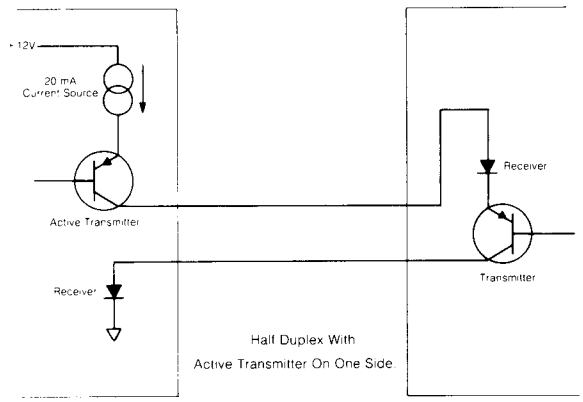



**Figure 2-7. Recommended Current Loop Circuits**

When attempting to interface to current-loop devices it will be necessary to consult an engineer for proper terminations of receivers and transmitters. The device schematic will probably be helpful. Additionally a book on asynchronous communication or data communication such as John E. McNamara's *Technical Aspects of Data Communication* (Digital Equipment Corp. Bedford, Mass, 01730; 1977) may be useful.

### 60 mA Current Loop

The passive switch and detector elements are capable of handling 60 mA current loop if the 20 mA current supplies are not used. The maximum off voltage allowed across the switch element is fifteen volts.

### 20 mA Current Supplies

Q1 is a current sink, and Q2 is a current source. R1 and R2 form a voltage divider and apply a constant voltage to the base of Q1. Q1 places a constant 2V across R3. Since R3 is 100 ohms, a constant 20 mA flows through it. D1 provides temperature compensation for Q1. The current source circuit operates in an anologous manner. High tolerance (1%) resistors are used to limit the total variation of current to ±10% under all conditions.

### Current Detector

A 6N138 opto-isolator (U7) is used as a current detector. The 150-ohm resistor shunting the input diode prevents the opto-isolator from turning on until at least 11 mA is flowing. Above 13 mA, R11 causes Q4 to conduct and draw the remaining current. The 0.1 $\mu$F capacitor between pins 5 and 8 prevents oscillations on the output of the opto-isolator.

### Current Switch

A 6N136 opto-isolator (U9) is used as a current switch. When U9 is turned on, it turns Q3 off.

## Functional Test

The following program listing may be entered and run to functionally test the interface. In most instances, running this test will indicate whether or not the interface is operational.

Use the following steps to enter and run the test from the keyboard:

1. Make sure the interface and I/O ROM are installed. If necessary, refer to page 20 and install the interface and I/O ROM.

2. Peripherals may or may not be connected to the interface.
3. Turn the computer power switch to the ON position.
4. Press the [SHIFT] key and, while this key is depressed, press the [AUTO] key (or simply type in A,U,T,O). This will cause the program lines to be numbered by ten in ascending order. Then press [END LINE].
5. Refer to the following program listing and enter the line by pressing these keys:

   [C][L][E][A][R] then press [END LINE]

   Continue this procedure until the entire program is entered, making sure to press [END LINE] as each numbered program line is entered.

```
10 ! 82939A SERIAL I/O
20 ! DIAGNOSTIC PROGRAM
30 CLEAR
40 N=0 ! SET TO NON-ZERO FOR CONTINUOUS RUNNING
50 R=0
60 DISP "82939A EXERCISER" @ PRINT "82939A EXERCISER" @ PRINT
70 S=0 ! FIND RS-232 CARD
80 ON ERROR GOTO 120
90 FOR I=3 to 10
100 STATUS I,0 ; A
110 IF A=2 THEN S=I @ I=10
120 NEXT I
130 IF I=0 THEN BEEP @ DISP "NO RS-232 CARD IN SYSTEM" @ STOP
140 OFF ERROR
150 RESET S
160 STATUS S,3 ; Z ! OPTION 000 OR OPTION 001?
170 Z=BIT(Z,2) ! SET OPTION FLAG
180 GOSUB 320 ! XLATOR/PROCESSOR TEST
190 GOSUB 570 ! SERIAL PORT TEST #1
200 GOSUB 630 ! SERIAL PORT TEST #2
210 GOSUB 850 ! STATUS BYTES TESTED
220 IF N#0 THEN GOTO 150
230 DISP "TEST COMPLETE" @ PRINT "TEST COMPLETE"
240 BEEP 100,200
250 PRINT
260 PRINT
270 PRINT
280 PRINT
290 PRINT
300 PRINT
310 END
320 DISP "TRANSLATOR AND PROCESSOR TEST"
330 N1=0 @ A4=0
340 WIO S,0;2
350 WIO S,1;113
360 WIO S,0;0
370 A5=1
380 IF RIO(S,0)=35 THEN 410
390 N1=N1+1 @ IF N1>2 THEN GOSUB 910
400 GOTO 380
410 A5=0
420 A1=RIO(S,1)
430 IF A1#113 THEN GOSUB 1060
440 FOR A2=0 TO 255
450 WIO S,1;A2
460 IF RIO(S,0)=35 THEN 530
470 N1=N1+1 @ IF N1>2 THEN GOSUB 910
480 IF N1<=2 THEN 460
490 PRINT "ON TO THE NEXT TEST"
```

```
500 A2=255
510 A3=RIO(S,1)
520 GOTO 550
530 A3=RIO(S,1)
540 IF A3#A2 THEN GOSUB 970
550 NEXT A2
560 RETURN
570 C=0
580 DISP "PORT 1 TEST"
590 B1=116
600 B3=10
610 GOSUB 700
620 RETURN
630 C=1
640 DISP "PORT 2 TEST"
650 B4=0
660 B1=117
670 B3=246
680 GOSUB 700
690 RETURN
700 WIO S,0;2
710 WIO S,1;B1
720 IF RIO(S,0)<128 THEN 760
730 N1=N1+1 @ IF N1>4 THEN GOSUB 910
740 IF N1>4 THEN 760
750 GOTO 720
760 WIO S,0;0
770 IF RIO(S,0)=35 THEN 810
780 N1=N1+1 @ IF N1>4 THEN GOSUB 910
790 IF N1>4 THEN 800
800 GOTO 770
810 B2=RIO(S,1)
820 IF RIO(S,0)#32 THEN GOSUB 1080
830 IF B2#B3 THEN GOSUB 1100
840 RETURN
850 DISP "STATUS TEST"
860 STATUS S,0 ; D0,D1,D2,D3,D4,D5,D6,D7,D8,D9,C1,C2
870 IF D0#2 OR D1#0 OR D2#0 OR D8#0 OR D9#137 OR C1#32
OR C2#0 THEN GOSUB 1350
880 IF BIT(D3,2)#Z THEN GOSUB 1350
890 IF D4#10 OR D5#0 OR D6#1 OR D7#128 THEN GOSUB 1410
900 RETURN
910 PRINT "HANDSHAKE FAILURE OR INTERFACE NOT LOGGED IN"
920 IF A5=0 THEN RETURN
930 PRINT "CHECK THAT THE SWITCHES ARE SET TO SELECT
CODE"; S
940 PRINT "NON-RECOVERABLE ERROR-TEST ABORT" @ BEEP
950 RESET S
960 GOTO 240
970 PRINT "OUTPUT";A2;"TO O.B. RECIEVED";A3
980 A4=A4+1
990 If A4>10 THEN 1010
1000 RETURN
1010 PRINT "GREATER THAN 10 ERRORS ";"CONTINUING WITH NEXT
TEST"
1020 A2=255
1030 RETURN
1040 PRINT "HANDSHAKE FAILURE-PROBABLE ";"PROCESSOR OR
TRANSLATOR FAILURE"
1050 RETURN
```

```
1060 PRINT "RECEIVED INCORRECT LOOPBACK ";"RESPONSE-
PROBABLE TRANSLATOR ";"FAILURE"
1070 RETURN
1080 PRINT "HANDSHAKE FAILURE-PSR NOT ";"CLEARED AFTER
HANDSHAKE"
1090 RETURN
1100 IF C=1 THEN GOTO 1200
1110 IF B2>127 THEN PRINT "PROCESSOR READS 8250 MR
LINE ";"PULLED-PROCESSOR OR 8250 ERROR"
1120 IF BIT(B2,6) THEN PRINT "S1-1 OPEN"
1130 B2=BINAND(B2,63)
1140 IF B2=10 THEN GOTO 1190
1150 PRINT "SWITCHES DID NOT READ AS FACTORY SETTINGS"
1160 A$=DTB$(B2)
1170 PRINT "S2-1 TO S2-6 READ AS "&A$[11,16]
1180 PRINT "---------->EXPECTED 001010"
1185 PRINT
1190 RETURN
1200 IF Z=0 THEN GOTO 1230
1210 IF BIT(B2,6) THEN PRINT "P26 READ HIGH-EXPECTED LOW"
1215 PRINT
1220 GOTO 1250
1230 IF BIT(B2,6) THEN GOTO 1250
1240 PRINT "P26 READ LOW EXPECTED HIGH"
1245 PRINT
1250 IF NOT BIT(B2,7) THEN PRINT "RLSD LINE READ IN WRONG
STATE" @ PRINT
1260 IF NOT BIT(B2,5) THEN PRINT "DSR LINE READ IN WRONG
STATE" @ PRINT
1270 IF NOT BIT(B2,4) THEN PRINT "CTS LINE READ IN WRONG
STATE" @ PRINT
1280 B2=BINAND(B2,15)
1290 IF B2=6 THEN GOTO 1340
1300 PRINT "SWITCHES DID NOT READ AS FACTORY SETTINGS"
1310 B$=DTB$(B2)
1320 PRINT "S2-7 TO S2-10 READ AS "&B$[13,16]
1330 PRINT "---------->EXPECTED 0110"
1335 PRINT
1340 RETURN
1350 PRINT "WARNING!! - STATUS BYTES NOT AT RESET DEFAULT"
1360 PRINT "STATUS BYTES=";D0;D1;D2;D3;D4;D5;D6;D7;D8;D9;
C1;C2
1370 PRINT "--->EXPECTED  2  0  0 ";
1380 IF Z=1 THEN PRINT " 4 ";ELSE PRINT " 0 ";
1390 PRINT " 10  0  1  128  0  137  0  0"
1395 PRINT
1400 RETURN
1410 IF D4=10 THEN GOTO 1460
1420 IF BIT(D4,6) THEN PRINT "BREAK ERROR"
1430 IF NOT BIT(D4,6) THEN PRINT "STATUS READS";D4 @ PRINT
1440 B2=BINAND(D4,63)
1450 GOSUB 1130
1460 IF D5=0 THEN GOTO 1510
1470 IF BIT(D5,4) OR BIT(D5,5) THEN PRINT "STATUS READS
AUTOHANDSHAKE SWITCH OPEN"
1480 IF BIT(D5,4) OR BIT(D5,5) THEN CONTROL S,5 ; 0
1490 IF NOT BIT(D5,4) OR NOT BIT(D5,5) THEN GOTO 1500 ELSE
GOTO 1520
1500 PRINT "MODEM HANDSHAKE STATUS ERROR- READ
";D5;"EXPECTED 0"
```

```
1510 PRINT
1520 PRINT "STATUS READS BAUD RATE SWITCHES NOT AT FACTORY
SETTINGS"
1530 PRINT "STATUS BYTE 6 READS";D6;"FACTORY SETTING=1"
1540 PRINT
1550 PRINT"STATUS READS BAUD RATE SWITCHES NOT AT FACTORY
SETTINGS"
1560 PRINT "STATUS BYTE 7 READS";D7;"FACTORY SETTING=128"
1565 PRINT
1570 RETURN
1580 END
```

6. After the program is entered press (RUN). The printer should print:

```
82939A EXERCISER
```

To run the test *indefinitely,* enter a nonzero value for N (at line 40); the test will run only once if N = 0 and when done will print

```
TEST COMPLETE
```

Any errors that occur will be recorded (printed) each time they occur. Be sure to specify your printer correctly and leave an adequate supply of paper if you begin the test to run continuously and then let it run without supervision.

## Re-Configuring the Cable

**CAUTION**

If it is necessary to re-configure the cable, please proceed with extreme caution. Improper configuration of the cable can result in seriously impaired operation or in permanent damage to the interface, the mainframe and the peripheral or terminal to which they are connected.

Individual modems may have features that are not compatible with the interface cable furnished with the HP 82939A Serial Interface. These features may require re-configuring of the interface cable before connecting it to the modem. This is accomplished by rewiring the 28-pin connector inside the interface housing.

### Rewiring the 28-Pin Connector

Rewiring is accomplished by removing a pin from one position in the connector and replacing it in another position. Refer to the interface schematic diagram (figure 3-4) for pin assignments.

The pin which is crimped onto each wire is held in place in the connector body by a locking tab (see figure 2-8). Holes on the side of the connector provide access to the locking tabs.

To remove a wire, insert the end of a paper clip in the hole by the appropriate pin. Press in to release the locking tab, then pull the wire and its pin from the connector body.

To install a wire, insert it in the desired pin position with the locking tab to the outside edge of the connector. Once it is installed, pull gently on the wire to ensure that the locking tab is locking the pin in place.

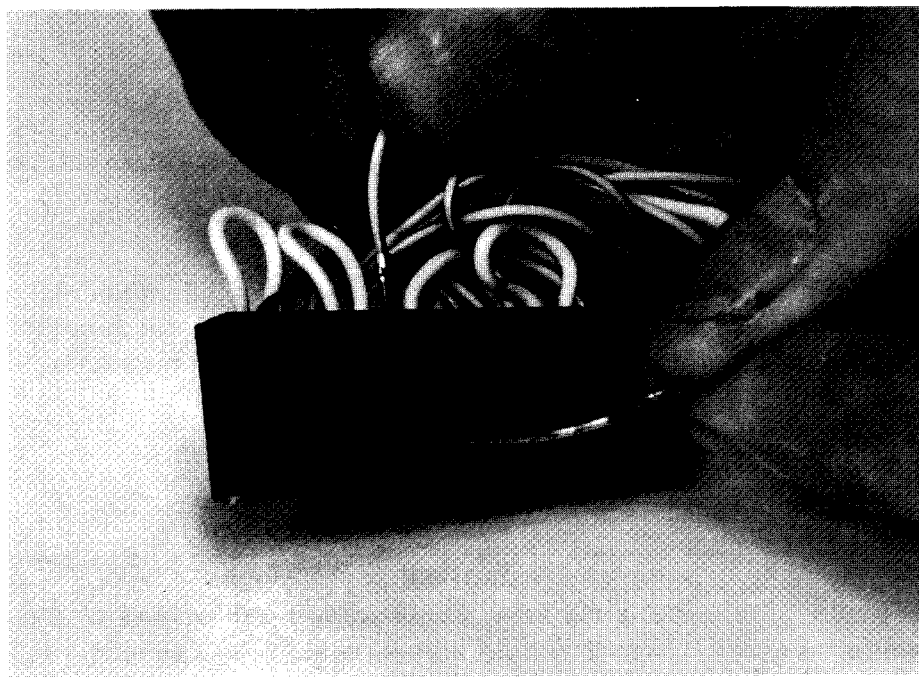
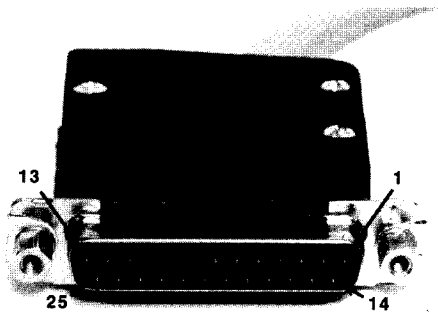
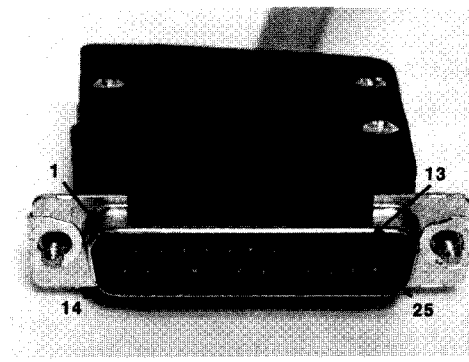


**Figure 2-8. Removing a Wire From the Interface Connector**

**Table 2-2. RS-232 Connector Pin Assignments**




| Pin No. | Wire Color | Signal Name |
|---|---|---|
| 1 | Shield | Protective Ground |
| 2 | Red | Transmitted Data |
| 3 | Orange | Recieved Data |
| 4 | Yellow | Request to Send |
| 5 | Green | Clear to Send |
| 6 | Blue | Data Set Ready |
| 7 | Violet | Signal Ground |
| 8 | Gray | Received Line Signal Detector |
| 12 | Black/White | Unassigned |
| 19 | Brown/White | Unassigned |
| 20 | Black | Data Terminal Ready |
| 23 | Brown | Data Rate Select |

Section 3

# Using the Serial Interface

## Introduction to Serial Interfacing

This topic explains serial I/O concepts and provides simple descriptions of the terms that are used in the printer and teletype discussions.

Serial I/O simply means the transmission of data, one bit after another, over a line. Contrast this with parallel I/O, which transfers eight or more data bits simultaneously. Each method of data transmission has unique advantages and disadvantages. Parallel I/O can transfer eight or more data bits at a time but requires one wire (or line) for each bit and one ground (or common) wire. Serial I/O transfers data one bit at a time but only requires one wire for the data and one wire for the ground. The cost and logistics of parallel I/O become prohibitive when considering communication over distances greater than 50 feet. Serial I/O allows for inexpensive long-distance communication through use of an existing telephone system.

## Equipment Configuration

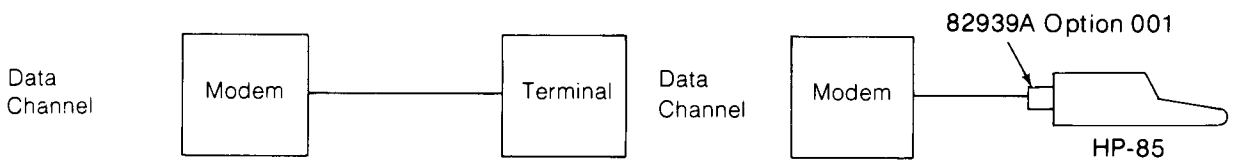
Serial I/O devices are described according to the functions that they perform. These functional descriptions are: Data Terminal Equipment, and Data Communications Equipment.

### Data Terminal Equipment

Data Terminal Equipment (DTE) is any location in a network where information can enter or exit. Items included as DTE are:

- The remote terminal.
- The remote terminal interface.
- The host computer.
- The host computer interface.

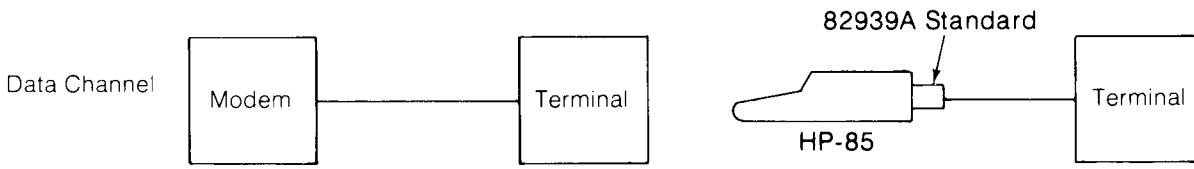
The option 001 interface configures the computer as a DTE device. The following drawings show a typical configuration for HP Series 80 Personal Computers.



### Data Communications Equipment

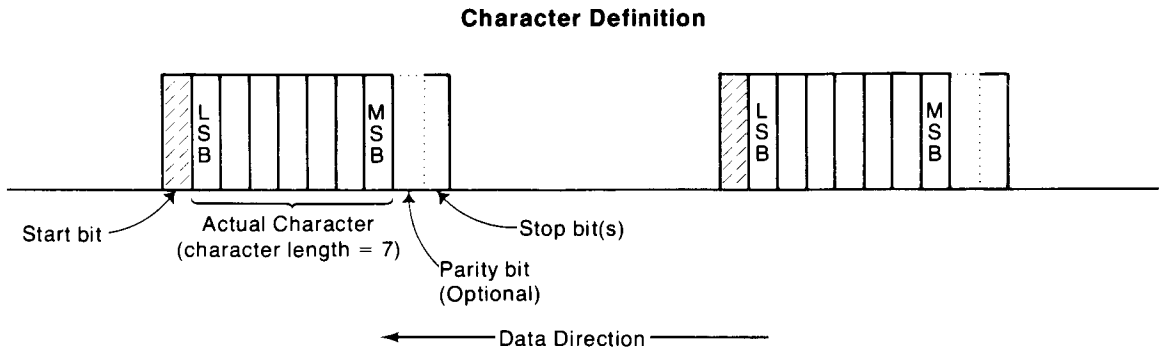
Data Communications Equipment (DCE) is the equipment used to convey information between locations. Items included as DCE are:

- The modems.
- The modem interfaces.
- The link (e.g., telephone lines).




## Asynchronous Data Transmission

The HP 82939A Interface communicates asynchronously with external devices. Asynchronous communication means simply that each character is sent over the line (or link) with synchronization built into the character. The next drawing shows typical asynchronous transmission of characters.

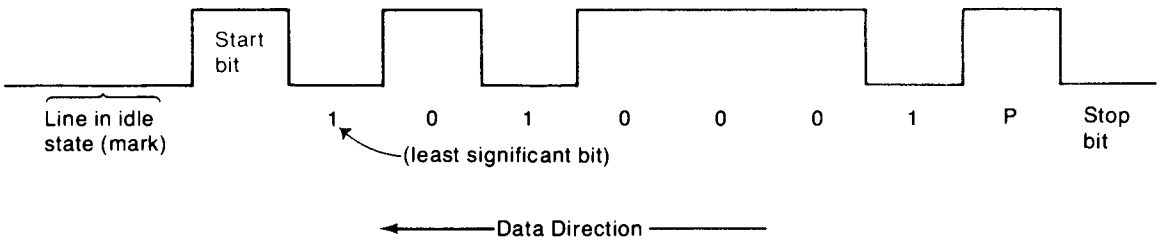



The actual data is transmitted over the line using two voltage levels to represent the two possible states of a binary digit. The following table shows the two binary states and the voltage levels assigned to each state.

| Binary State: | logic 0 | logic 1 |
|---|---|---|
| Voltage Range: | + 3V to + 25V | − 3V to − 25V |
| Level Name: | SPACE | MARK |
| Line State: | high | low (idle) |

When data is not being transmitted, the line is held in the low (or mark) state. When the transmitting device has data to send, it places the line in the high (space) state for one bit time. This change to the high state for one bit time is called the start bit. The remaining digital data is then transmitted to the receiving device. The following drawing shows how a typical character (an ASCII "E") is transmitted over the link.

**Character Transmission**




The previous drawings show characters that consist of a start bit, the actual ASCII character, a parity bit, and a stop bit. The meaning of each of these parts of the character is explained next.

## Start Bit

The start bit is inserted at the beginning of the character by the interface. The start bit is used to signal the receiving device that the transmission of a character is starting. When the start bit is detected, the receiver starts its internal clock to synchronize the receiver to the input data.

## Data Character

The character is the binary bit-pattern of the actual transmitted character. For example, the bit pattern 0 110 010 is transmitted for the ASCII "2" character. Assuming that the receiver expects ASCII characters, it interprets the bit pattern as a "2".

Note that the ASCII code is made up of seven bits. Other codes may be made up of five, six, or eight bits. The interface provides a "character length" specification to define the number of bits that make up a character. This "character length" specification does not include start, stop, or parity bits. The factory setting for the reset default switches specifies seven bit character length. See Register 4 in appendix B for further discussion about character length.

## Parity

Parity provides a method of error checking. The parity bit (if specified) always follows the character. No parity bit is added when parity is not specified (parity = none). The parity bit is always a "1" when parity = 1 is specified, and the parity bit is always a "0" when parity = 0 is specified. Parity = even and parity = odd specify that the parity bit is determined as shown next.

| Number of "1" Bits in Character | Parity Specified | Parity Bit |
|---|---|---|
| Odd | Odd | 0 |
| Even | Odd | 1 |
| Odd | Even | 1 |
| Even | Even | 0 |

For example, the bit pattern for the ASCII "2" character is 0 110 010. There are three "1" bits in this pattern. If odd parity is specified, the parity bit is "0". If even parity is specified, the parity bit is "1". The factory setting for the reset default switches specifies odd parity. See Register 4 in appendix B for further discussion of parity.

### Stop Bits

Stop bits are added following the parity bit by the interface. The stop bits are not really bits. The transmitter holds the line in the "idle" state for the amount of bit times specified by the stop bits parameter. This amount of bit times is referred to as stop bits. Allowable stop bits parameters are 1 and 2. The factory setting for the reset default switches specifies one stop bit. See Register 4 in appendix B for further discussion of stop bits.

### Transfer Rates (or Baud)

When two devices are communicating, they must transfer and receive information at compatible data rates. If the transmitter sends data at a faster rate than the receiver is expecting the data, information will be lost. Most devices (such as printers) provide a switch to select the data rate. The serial I/O interface provides programmable data rates and a switch selectable default data rate (see Control Register 3).

### Handshakes

Handshakes are used to communicate status information from one device to another. The handshakes are used to indicate a buffer full condition, received data errors, and modem status. Some devices use modem lines to indicate an input buffer full condition (see Printer Interfacing). Other types of handshake protocols (DC1/DC3 and ENQ/ACK) are explained in the Advanced Serial Interfacing section.

## Printer and Terminal Interfacing

This topic explains how to connect the HP Series 80 computer to a printer or teletype in order to produce hard-copy output and to a terminal to input data. Example programs are shown for typical applications.

### Printer Interfacing

Interfacing to a printer is not difficult if you first determine the printer's requirements. Interfacing information can be found in the owner's or operator's manual supplied with the printer. Look for key information that may be listed as:

- Technical Data.
- Technical Specifications.
- Performance Specifications.
- Transfer Rates.
- Baud Rates.
- Character Set.
- Interfacing Diagrams.
- Handshake.

### Example 1 (Printer)

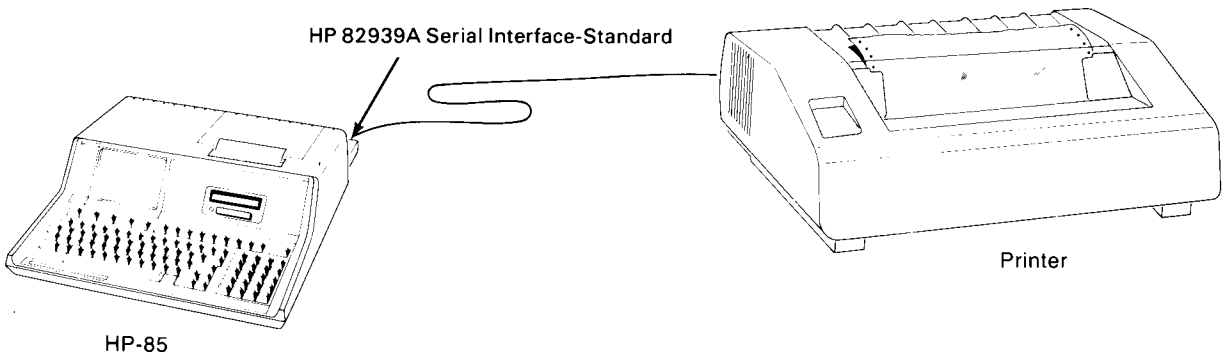
As an example, let's assume that the following information is obtained from the owner's manual supplied with your printer:

- Character Set—Asynchronous bit serial. Seven data bits and one parity bit or eight data bits and no parity. The eighth bit is ignored. The printer accepts one or two stop bits. Full 96-character ASCII coding.

- Serial Baud Rate—Switch selectable rates of 110, 300, 600, and 1200 bits per second.
- Handshake—The printer generates the Data Terminal Ready signal to regulate Received Data input and prevent input buffer overflow. When the input buffer is full, the Data Terminal Ready signal is dropped. This prevents further data transfer until the printer input buffer can accept more data from the computer.
- Interfacing—The following diagrams show the connections for Data Terminal Equipment (DTE) and Data Communications Equipment (DCE) configurations.

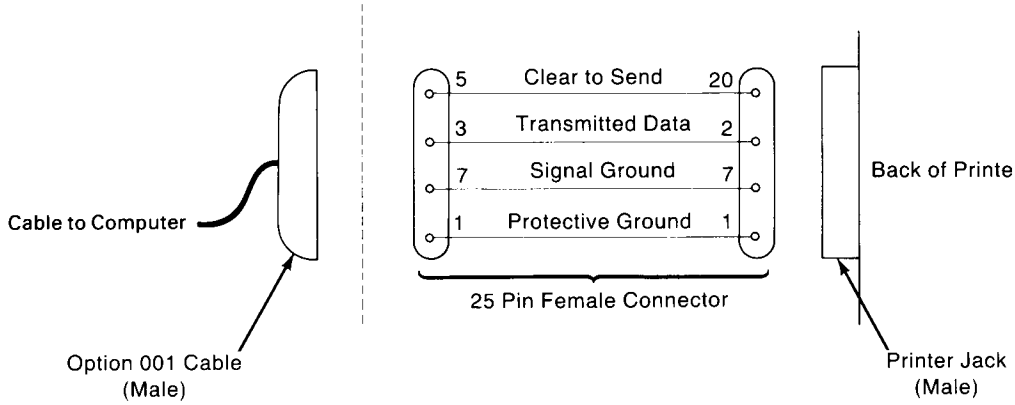
Here's how the printer is interfaced with the HP-85.

**Printer to Standard Interface**




When connecting this printer to the Option 001 interface, the following adapter cable must be used.

**Printer to Option 001 Interface**




Let's use the data that we have collected thus far. The interface is preset at the factory to the following default values:

- Interface select code = 10.
- Transfer rate = 300 baud.
- Autohandshake = Off.
- Character length = 7.
- Parity = Odd
- Stop bits = 1.

If we compare the requirements of the printer with the defaults provided by the interface, we see that the printer baud rate must be set to 300, and the interface autohandshake feature must be enabled.

This example shows the simple programming steps that enable the HP Series 80 Personal Computer using printer commands.

Set the printer Baud Rate Switch to the 300* baud position. Key the following program into your computer.

```
10 PRINTER IS 10
20 SET I/O 10,5,48  ! ENABLE AUTO HANDSHAKE
30 FOR X=1 TO 10
40 PRINT; "NUMBER=";X
50 NEXT X
60 STOP
```

The printer should print the data as shown next.

```
NUMBER= 1
NUMBER= 1
NUMBER= 2
NUMBER= 3
NUMBER= 4
NUMBER= 5
NUMBER= 6
NUMBER= 7
NUMBER= 8
NUMBER= 9
NUMBER= 10
```

When using the I/O ROM, the `SET I/O` command is replaced with `CONTROL 10,5 ; 48` and `PRINT` is replaced with `OUTPUT`.

*The example shown here uses default values provided by the serial I/O interface whenever possible. Since the printer uses a handshake to prevent buffer overrun, the baud rate can be set to maximum rate allowed by the printer (in this case 1200) to obtain maximum throughput.

You should note that many commercial printers have a female output connector, but the connector is wired as a DTE (male) device. This configuration must be changed when the interface option 001 cable is used. Pins 2 and 3 on the interface cable or on the printer connector must be interchanged. A cable is available that implements these wiring changes. Order Hewlett-Packard P/N 8120-3097.

Example 1 implements the auto-handshake feature provided by the interface. Other handshakes such as "Enquire/Acknowledge" and XON/XOFF may be used. See the Advanced Serial Interfacing section for an explanation of these handshakes.

## Terminal Interface

The standard interface is supplied with a 25-pin female EIA connector. The standard interface is used to connect to a terminal. Check the owner's manual supplied with the terminal for key information such as baud rate, character set, interfacing diagrams, and technical specifications. Most terminals provide a half/full duplex switch. The serial I/O interface can operate in either half or full duplex mode. Full duplex terminals usually require that their transmitted data be echoed by the host computer. The host computer simply retransmits each received character back to the terminal. This retransmission (or echo) of characters provides the terminal operator with a visual indication of the data that was transmitted to the computer. See Register 9, bit 1 for a description of the auto-echo feature.

### Example 2 (Terminal)

Here's how to interface with a terminal. Let's assume that the following information is obtained from the owner's manual supplied with your terminal:

- Character Set—Asynchronous bit serial; 1 start bit with 7 data bits, odd parity and 1 or 2 stop bits. Full 96-character ASCII coding.
- Serial Baud Rate—Switch selectable rates of 110, 300, 600, and 1200 bits per second.
- Duplex—Switch selectable half and full-duplex operation.
- Handshake—The terminal uses XON/XOFF handshake (see the Advanced Serial Interfacing Section for XON/XOFF handshake details).
- The terminal requires a DC1, CHR$(17), prompt from the remote device in order to begin transmitting.
- The terminal transmits a line-feed character as its output end-of-line terminator.
- The terminal requires a carriage-return/line-feed as its input end-of-line terminator.

Here's how to interface the terminal with an HP Series 80 Personal Computer:

Set the terminal switches as follows. Baud Rate = 300, Duplex = Full. The reset defaults for the serial I/O interface specify 300 Baud, and seven bit ASCII characters with odd parity. The only interface card specifications that must be programmed are: enable the auto-echo feature, and implement the XON/XOFF handshake protocol. The following program shows how to set up the interface to communicate with the terminal when using the I/O ROM.

Key the following program into your computer.

```
10 DIM A$[50]
20 CONTROL 10,9 ; 139 ! ADD AUTO ECHO FEATURE TO REG 9
30 CONTROL 10,11 ; 194,10 ! ENABLE XON/XOFF-REG 12
(CR=13,LF=10)
40 CONTROL 10,14 ; 17,19 ! REG 14 = DC1, REG 15 = DC3
50 OUTPUT 10 ;"START TERMINAL";CHR$(17)
60 ENTER 10 USING "%,%K" ; A$
70 OUTPUT 10 USING "#,A' ; CHR$(10)
80 P=POS(A$,CHR$(8))
90 IF NOT P THEN 120
100 A$[P-1]=A$[P+1]
110 GOTO 80
120 DISP A$
130 GOTO 60
140 END
```

Line 30—Activates Transmitter Flag Enable/Disable feature and specifies Control Register 12 character as an input termination character. Line 30 also sets the value 10 (for Line-Feed) to register 12.

Line 40—Sets the value 17 (DC1) to register 14 and the value 19 (DC3) to register 15. These characters control the Transmitter Flag Enable/Disable feature.

Line 60—Enters the data from the terminal. The "%,%K" image specifier allows the entry into A$ to be terminated by an "EOI". This interface generates an "EOI" type signal when the character defined by register 12 is detected in the incoming data.

Line 80 through Line 110—Interprets the Backspace character.

You can now enter data from the terminal.

## Teletype Interface

The option 002 interface is supplied with an unterminated cable for use with a teletype device. See section 1 for a description of the option 002 cable. Most teletypes use a 20 milliamp current loop mode for data transfer. The interface provides 20 milliamp drivers for current loop operation. Most teletypes are configured as shown next:

- Data Rate—110 baud.
- Character—7 bits.
- Parity—Even.
- Stop bits—2 stop bits.

### Example 3 (Teletype)

The following program transmits data to a teletype that is configured as shown:

```
10 DIM A$[50]
20 CONTROL 10,3 ; 2 ! 110 BAUD
30 CONTROL 10,4 ; 30 ! 7 BITS, EVEN PARITY, 2 STOP BITS
40 ! THE 20 NULL SEQUENCE ALLOWS 200 MILLISECONDS @ 110
50 ! BAUD FOR TELETYPE TO EXECUTE THE CR/LF SEQUENCE
60 CONTROL 10,16 ; 22 ! EOL = CR/LF AND 20 NULLS
70 INPUT A$
80 OUTPUT 10 ;A$
90 GOTO 70
100 END
```

# Advanced Serial Interfacing

This topic explains advanced interfacing techniques. Items discussed include handshakes, modems, and long distance communication over telephone links.

## Handshakes

In the printer discussion, the printer used the Data Terminal Ready signal to indicate a buffer-full condition to the computer. Other types of handshakes exist, as discussed here.

### ENQuire/ACKnowledge

Some peripherals and host computer systems use the ENQuire/ACKnowledge protocol for buffer-full or not-ready indication. When the transmitting device sends a line of text to a receiver, an ENQuire character, e.g., CHR$(5), is also transmitted following the text. The transmitter waits for an ACKnowledge character, e.g., CHR$(6), from the receiver before sending another line of text. The receiver responds with the ACKnowledge response if the input data contained no errors and there is sufficient space in the input buffers for at least one more line of test. This discussion uses the ASCII ENQ and ACK characters for the handshake sequences. Other ASCII characters such as ENQ/ESC may be used for these sequences. The following program lines how how to set Control Registers 11, 15, 16, and 19 to implement the ENQuire/ACKnowledge handshakes when the HP Series 80 Personal Computer is defined as the host.

```
10 CONTROL 10,11 ; 128 ! ALLOW XMIT FLAG ENABLE
20 CONTROL 10,15 ; 6 ! ACK ENABLES XMIT FLAG
30 CONTROL 10,16 ; 67 ! 3 CHAR EOL AND DISABLE XMIT
40 CONTROL 10,19 ; 5 ! ENQ IS 3rd EOL CHARACTER
```

### XON/XOFF

Another handshake protocol used by some peripherals and host computer systems is the XON/XOFF handshake (commonly referred to as the DC1/DC3 handshake). During data transfers, the receiver monitors its input buffers to ensure that sufficient space remains for at least one more line of data. When there is not sufficient space remaining in the input buffers, the receiver sends an XOFF (normally a DC3) to the transmitter. The transmitter then suspends further transmission until the receiver sends an XON (normally a DC1) to indicate that transmission may resume. The receiver sends the XON when sufficient buffer space becomes available for at least one more line of text. The following program lines show how to set Control Registers 11, 14, and 15 to implement XON/XOFF handshakes when the HP Series 80 Personal Computer is defined as the host.

```
10 CONTROL 10,11 ; 192 ! SET XMIT FLAG FOR XON/OFF
20 CONTROL 10,14 ; 19 ! DC3 IS XOFF
30 CONTROL 10,15 ; 17 ! DC1 IS XON
```

**Note:** When the HP Series 80 Personal Computer is configured as an input device (not as a host), you must implement the ENQ/ACK or XON/XOFF handshake sequences to prevent overflowing the computer buffers. The handshakes must be implemented from the BASIC program.

## Modems

The word **modem** is a contraction of the words MOdulator and DEModulator. A modem is a device that changes digital information into audio tones for transmission over the existing telephone system and converts received audio tones into digital information for input to the receiving device. A modem is used to establish a communication link using telephone lines, since digital data can only be transmitted over short distances with direct point-to-point wiring. Also, the bit format of digital data is incompatible with long distance communication.

Modems utilize handhake signals to communicate with the device to which they are connected. The serial I/O interface implements the following RS-232C modem control signals:

**Table 3-1. Control Signals**

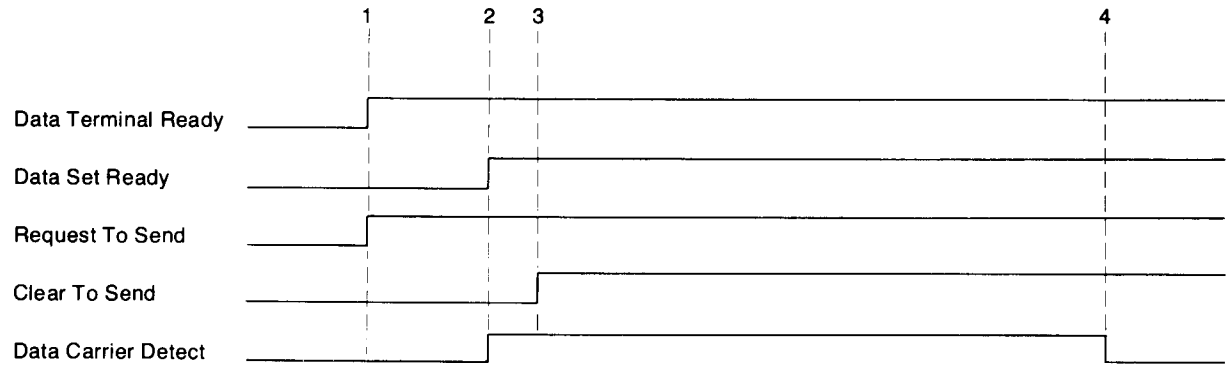
| Cable Pin | Signal Name | Function |
|---|---|---|
| 1 | Protective Ground | Frame and ac power ground |
| 2 | Transmitted Data | Output data sent by the DTE device to the DCE device. |
| 3 | Received Data | Input data detected by the DCE device. |
| 4 | Request to Send | Indicates to the DCE device that the DTE device is ready to accept data. |
| 5 | Clear to Send | Indicates to the DTE device that the DCE device is ready to accept data. |
| 6 | Data Set Ready | Indicates to the DTE device that the DCE device is not in test mode and power is on. |
| 7 | Signal Ground | Establishes reference point between the remote device and the terminal. |
| 8 | Data Carrier Detect | Indicates to the DTE that the DCE device is receiving carrier signals. |
| 20 | Data Terminal Ready | Indicates to the DCE that the DTE device is ready to transfer data. |
| 23 | Data Signal Rate Select | Selects one of two rates available on two speed modems. |

You can control and monitor these modem signals by accessing interface control and status registers. See appendix B.

The next drawings show typical handshake sequences that occur between a modem and a terminal. Half and full-duplex handshakes are shown.

### Establishing Full Duplex Connection

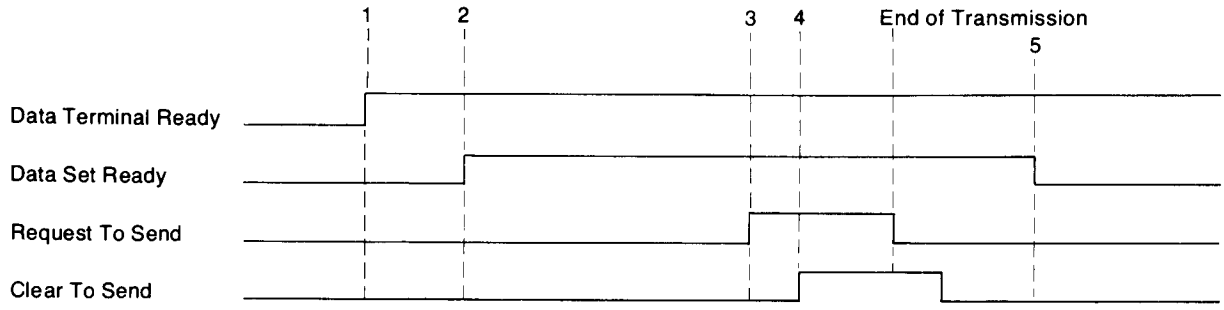
This drawing shows typical handshake sequences that occur when full duplex data transfers are implemented.




1. Interface sets Data Terminal Ready and Request To Send (Register 2, bits 0 and 1).
2. Interface monitors Data Set Ready and Data Carrier Detect responses. These signals are monitored from the program (Status Register 3, bits 1 and 3), or can be automatically monitored by the interface (Register 5, bits 1 and 3).
3. Interface monitors Clear To Send response. When Clear To Send becomes active (true), the interface transmits the output data to the remote. This signal is monitored from the program (Status Register 3, bit 0), or can be automatically monitored by the interface (Register 5, bits 0 and/or 4). The automatic monitoring feature enables you to suspend transmission or execute an auto-disconnect if Clear To Send, Data Set Ready, or Data Carrier Detect signals are lost (goes false).
4. Auto disconnect for lost carrier.

### Establishing Half Duplex Connection

This drawing shows typical handshake sequences that occur when half duplex data transfers are implemented:




1. Interface sets Data Terminal Ready (Register 2, bit 0).
2. Interface monitors Data Set Ready response (Status Register 3, bit 1).
3. Interface sets Request To Send. Automatic control of this signal can be specified (Control Register 16, bit 7).
4. Interface monitors Clear To Send response. When Clear To Send is active (true), the interface transmits the output data. Automatic control of this handshake sequence can be specified (Register 5, bit 4).
5. Auto disconnect when Data Set Ready signal is lost.

The modem handshake sequences described here are for the option 001 interface. Similar automatic signal monitoring, auto-handshake and auto-disconnect features are provided for the standard interface. See the appropriate registers for an explanation of these features.

> **Note:** Full duplex transfers can not be implemented using ENTER and OUTPUT statements. Use interrupt TRANSFERs instead. This allows the serial I/O interface to interrupt the computer when there is input data present, or to request data for transmission.

## Auto-originate and Auto-answer Routines

This discussion shows typical auto-originate and auto-answer programming routines that are used when connecting to a remote computer. These routines should be added to the main program used to communicate with the remote computer.

### Example 4 (Auto-originate)

```
10  ! ==========================
20  ! This routine controls a
30  ! modem for originating a
40  ! call.
50  !
60  ! Modem Lines:
70  !    DTR: Data Terminal Ready
80  !    RTS: Request to Send
90  !    DSR: Data Set Ready
100 !    DCD: Data Carrier Detect
110 !    CTS: Clear to Send
120 !
130 ! S=I/O Card Select Code
140 ! ==========================
150 !
160 ! Turn ON DTR & RTS.
170 !
180 ASSERT S;3
190 !
200 ! Establish Connection
210 !
220 DISP "Dial Number - then 'CONT'"
230 PAUSE
240 !
250 ! Enable autodisconnect.
260 ! If DSR, DCD or CTS drop
270 ! then autodisconnect will
280 ! turn OFF DTR & RTS and
290 ! generate an error.
300 !
310 CONTROL S,5 ; 11
320 !
330 ! TransferData
340 !
350 ! ************************
360 ! Add Data I/O routine here
370 ! ************************
380 END
```

**Example 5 (Auto-answer)**

```
10 ! ========================
20 ! This routine controls a
30 ! modem for autoanswer of an
40 ! incoming call.
50 !
60 ! Modem Lines:
70 !   DTR: Data Terminal Ready
80 !   RTS: Request to Send
90 !   DSR: Data Set Ready
100 !  DCD: Data Carrier Detect
110 !  CTS: Clear to Send
120 !
130 ! S=I/O Card Select Code
140 ! ========================
150 !
160 ! Turn ON DTR & RTS.
170 !
180 ASSERT S;3
190 !
200 ! Wait for incoming call
210 ! (DSR 'ON')
220 !
230 STATUS S,3 ; M
240 IF NOT BIT(M,1) THEN 230
250 !
260 ! Start 25 second connect
270 ! timer and enable auto-
280 ! disconnect. If DSR, DCD
290 ! or CTS drop then auto-
300 ! disconnect will turn OFF
310 ! DTR & RTS and branch to
320 ! line 590.
330 !
340 ON TIMER# 1,25000 GOTO 580
350 ON INTR S GOTO 590
360 ENABLE INTR S;4
370 CONTROL S,5 ; 11
380 !
390 ! Check DCD line.
400 !
410 STATUS S,3 ; M
420 IF NOT BIT(M,3) THEN 410
430 !
440 ! If DCD 'ON' THEN disable
450 ! timer and transfer data.
460 !
470 ! OFF TIMER# 1
480 ! **************************
490 ! Add Data I/O routine here
500 ! **************************
510 !
520 ! If timer times out before
530 ! DCD turns ON then turn
540 ! OFF DTR & RTS, disable
550 ! autodisconnect, wait 5
```

```
560 ! sec & setup for next call
570 !
580 ASSERT S;0
590 CONTROL S,5 ; 0
600 OFF TIMER# 1
610 STATUS S,1 ; M
620 WAIT 5000
630 GOTO 180
640 END
```

## BASIC (I/O ROM) Statements

The HP 82939A Serial Interface is controlled using the HP Series 80 BASIC statements listed below. A detailed discussion of each statement is contained in the I/O ROM Manual.

**Table 3-2. Summary of Basic Statements**

| Statement | Function |
|---|---|
| ABORTIO *(isc)* | Abort transfers in prograss and drop all modem lines. |
| ASSERT *(isc)* ; *(exp)* | Interrupting write to modem control register. (Immediate—does not wait for Tx character.) |
| CONTROL *(isc)* , *XX** ; *(list)* | See CONTROL summary (in appendix B). |
| ENABLE INTR *(isc)* ; *(exp)* | Enable interrupts with (exp) as mask. |
| ENTER *(isc)* ; *(list)*<br>*(isc)* USING *(string)* ; *(list)*<br>*(isc)* USING *(line#)* ; *(list)* | Read Rx characters in default format.<br>Read Rx characters in (string) format.<br>Read Rx characters with format of (line#). |
| HALT *(isc)* | Abort transfers in progress and leave modem lines unchanged. |
| OUTPUT *(isc)* ; *(list)*<br>*(isc)* USING *(string)* ; *(list)*<br>*(isc)* USING *(line#)* ; *(list)* | Transmit data in default format.<br>Transmit data with (string) format.<br>Transmit data with format of (line#). |
| REQUEST *(isc)* ; *(exp)* | Sends a break of (exp) character-time duration ((exp) character-time spacing line followed by a 5 character-time marking idle line). |
| RESET *(isc)* | Reset card: disconnect modem, run self-test, initialize card to defaults. |
| RESUME *(isc)* | Enable transmitter. |
| SEND *(isc)* ; DATA *(list)*<br>[END] | Output data as bytes.<br>Execute EOL sequence after data. |
| SET I/O *(isc)* , *XX** ; *(numeric data)* | Modify control register values (see appendix B). |
| STATUS *(isc)* , *XX** ; *(list)* | See STATUS summary (in appendix B). |
| TRANSFER *(isc)* TO *(buffer)*<br>*(buffer)* TO *(isc)*<br>INTR<br>[COUNT *(exp)*] | Read receive data into buffer.<br>Send transmit data from buffer.<br>Interrupt transfers.<br>Byte count to terminate input transfer.<br>Input termination character. |

*The "XX" indicates the register number being addressed.

Section 4

# Maintenance, Service, and Warranty

## Maintenance

There are no customer serviceable parts inside the HP 82939A Serial Interface. It should not be necessary to clean the interface module or cable contacts. The action of installing the module in the port or plugging the cable into a peripheral is normally sufficient to clean contamination from the contacts.

## Service

If at any time you suspect that the interface may be malfunctioning, do the following:

1. Turn off the computer and all peripherals. After disconnecting all plug-in devices from the ports, turn on the computer. If the cursor appears and no error message is displayed, the computer is functioning properly.
2. Turn off the computer. After installing the interface module in question in any port, turn on the computer.
   - If `Error 110 : I/O CARD` appears, the interface module requires service.
   - If the cursor does not appear, the system is not operating properly. To help determine if the interface module is interfering with proper operation, repeat this step with the module installed in a different port.
3. If improper operation is indicated in either the interface module or the computer, repair service is required.

## Warranty and Repair Service Information

The warranty statement and procedures for obtaining repair service are contained on the Warranty and Service Information sheet shipped with your HP 82939A Serial Interface. If you need additional information, please contact your authorized HP dealer or the nearest Hewlett-Packard sales and service facility.

If you have any questions concerning the warranty, and you are unable to contact the authorized HP dealer or the HP sales office where you purchased your computer, please contact:

**In the U.S.:**

Hewlett-Packard
Corvallis Division Customer Support
1000 N.E. Circle Boulevard
Corvallis, OR 97330
Telephone: (503) 758-1010
Toll Free Number: (800) 547-3400 (except in Oregon, Hawaii, and Alaska).

**In Europe:**

Hewlett-Packard S.A.
7, rue du Bois-du-lan
P.O. Box
CH-1217 Meyrin 2
Geneva
Switzerland

**Other Countries:**

Hewlett-Packard Intercontinental
3495 Deer Creek Road
Palo Alto, California 94304
U.S.A.
Tel. (415) 856-1501

## Radio Frequency Interference Statement

The HP 82939A Serial Interface uses radio frequency energy and may cause interference to radio and television reception. The interface has been type-tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of the FCC Rules. These specifications provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If the interface does cause interference to radio or television, which can be determined by turning the computer on and off with the interface installed and with the interface removed, you can try to eliminate the interference problem by doing one or more of the following:

- Reorient the receiving antenna.
- Change the postition of the interface cable with respect to the receiver.
- Change the position of the computer with respect to the receiver.
- Move the computer away from the receiver.
- Plug the computer into a different outlet so that the computer and the receiver are on different branch circuits.

If necessary, consult an authorized HP dealer or an experienced radio/television technician for additional suggestions. You may find the following booklet, prepared by the Federal Communications Commission, helpful: *How to Identify and Resolve Radio-TV Interference Problems.* This booklet is available from the U.S. Government Printing Office, Washington, D.C. 20402, Stock No. 004-000-00345-4.

# RS-232C

## RS-232C Compatible

What is the meaning of "RS-232C Compatible"? Or, of more importance, what doesn't it mean?

To answer the latter question first, it does not mean that every piece of equipment bearing that label will work perfectly with every other piece of equipment so labelled. What it does mean is that the equipment does not exceed any of the specifications or characteristics set down in the standard known as EIA RS-232C. But within the scope of RS-232C there is enough latitude to permit minor imcompatibilities from one device to another, and these minor incompatibilities can cause unpleasant surprises for the unwary.

## What is RS-232C?

In 1963, the Electronic Industry Association (EIA) established a standard to govern the Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial Binary Interchange. The latest revision of this standard has been in effect since 1969 and is known colloquially as RS-232C. It specifies:

- Mechanical characteristics of the interface.
- Electrical characteristics of the interface.
- A number of interchange circuits with descriptions of their functions.
- The relationship of interchange circuits to standard interface types.

The Comite Consultatif International Telephonique et Telegraphique (CCITT) has established standards that correspond to RS-232C. While these standards, CCITT V.24 and CCITT V.28, are very similar to RS-232C, they are not identical. Because it does not make use of all the circuits defined in both RS-232C and CCITT V.24, the HP 82939A Serial Interface conforms to both RS-232C and CCITT V.24 without any modification of the interface. The circuits which are utilized vary with different applications and with different modems. The drivers and receivers used in the HP 82939A Serial Interface conform to voltage and other electrical specifications of both CCITT V.28 and RS-232C.

## Mechanical Characteristics

The standard gives definitions to 22 pins and designates three pins as unassigned, but does not specify a 25-pin connector. Although a particular 25-pin connector is not defined, the industry has accepted the connector shown in figure A-1 as a de facto standard. The male connector is used with data terminal equipment (the computer), and the female connector is used with data communications equipment (the modem).

The length of the cable used by data terminal equipment to connect data communications equipment should not be longer than 15.24 metres (50 feet). This is assuming that the load capacitance at the interface point is the worst case value of 2,500 picofarads. Longer cables are often used, especially in point-to-point configurations when the user knows that the total load capacitance will not exceed the 2,500 pf maximum.

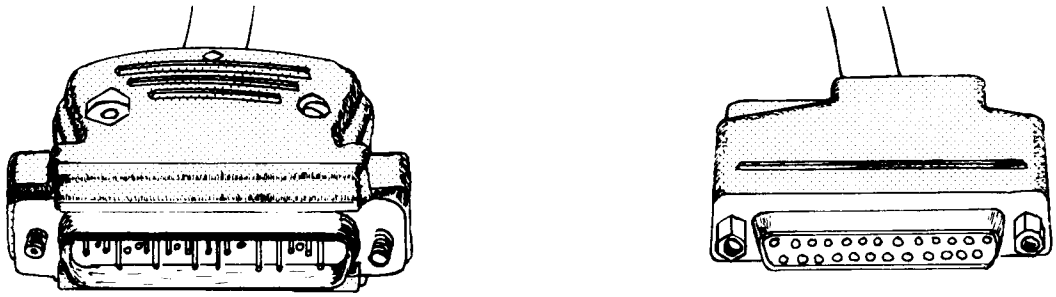


**Figure A-1. 25-Pin Connector**

## Electrical Characteristics

A number of electrical parameters and limitations are defined by RS-232C for each interchange circuit. They refer to the equivalent interchange circuit shown in figure A-2. All voltage measurments are made at the interface point and with reference to signal ground.

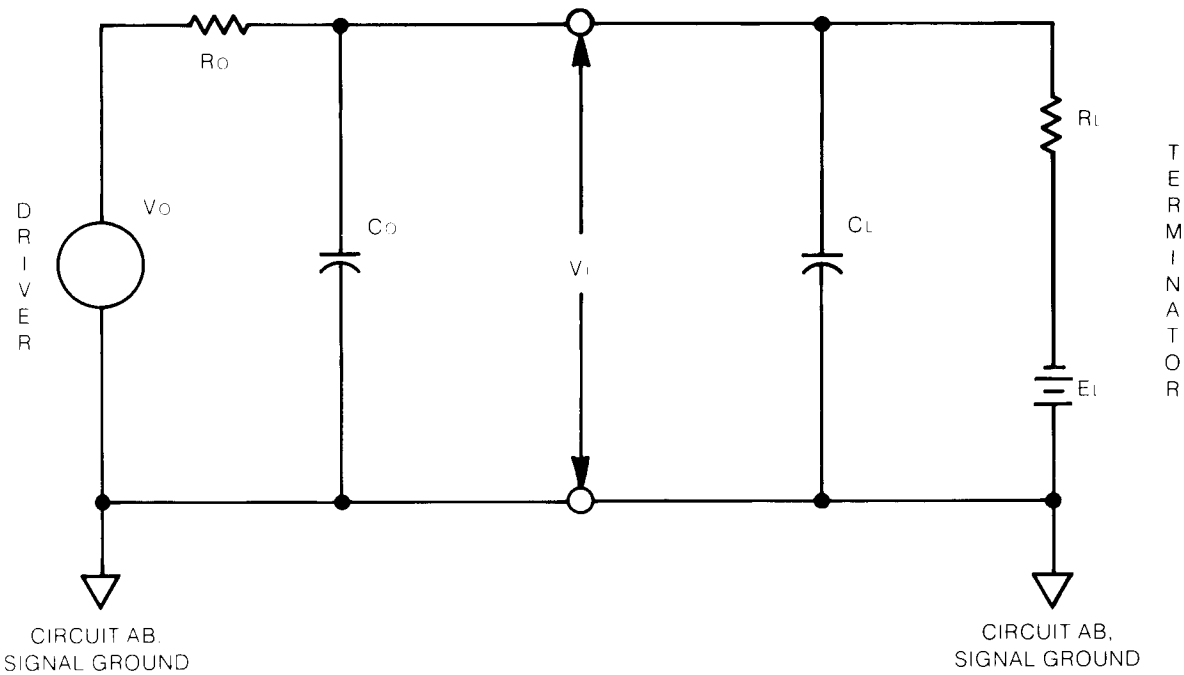



**Figure A-2. Interchange Equivalent Circuit**

- Open circuit voltage from the driver shall not be greater than ±25 volts.
- The open circuit voltage of the terminator shall not exceed ±2 volts.
- The total capacitance of the terminator shall not exceed 2,500 picofarads.
- The driver output voltage must be between 5 and 15 volts when the total terminator input resistance is between 3,000Ω and 7,000Ω.
- The output impedance of the driver circuit, when the driver power is off, shall not be less than 300Ω.
- The rate of change of the driver output voltage (slew rate) shall not exceed 30 volts per microsecond.

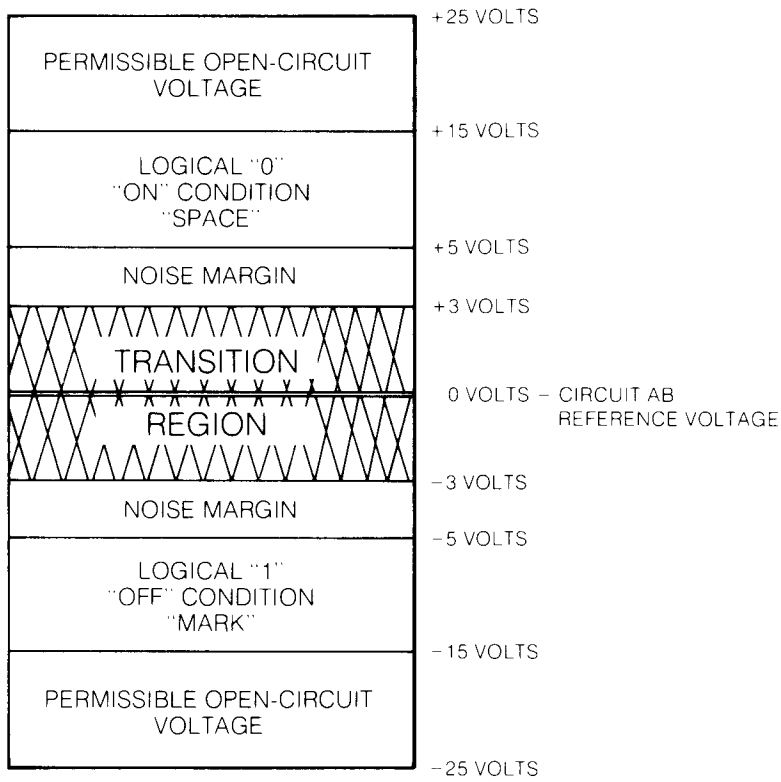



**Figure A-3. Circuit Voltage Levels**

In addition, several rules define the logic state indicated by voltage levels on the circuit.

- A logic 1 (MARK) is indicated when the voltage at the interface point is more negative than −3 volts.
- A logic 0 (SPACE) is indicated when the voltage at the interface point is more positive than +3 volts.
- To indicate a "1" signal condition (MARK), the driver shall assert a voltage between −5 volts and −15 volts.
- To indicate a "0" signal condition (SPACE), the driver shall assert a voltage between +5 volts and +15 volts.

Note that these standards allow for a 2-volt noise margin between the minimum driver voltage of 5 volts and the maximum undefined voltage of 3 volts. Other specifications that govern the transition region:

- All interchange signals entering the transition region shall proceed to the opposite valid signal state. It shall not re-renter the transition region until the next significant change in signal state.
- While in the transition region, the direction of the voltage change must not reverse.
- The time required for a control signal to cross the transition region shall not exceed one millisecond.
- The time required for a data or timing signal to cross the transition region shall not exceed one millisecond or four percent of the nominal signal period, whichever is the lesser.

## RS-232C Function Table

This table contains the RS-232C functions, listed by pin number in the connector. Those functions marked with an asterisk are implemented in the HP 82939A Serial Interface.

Pin 1 ***Protective Ground.** Electrical equipment frame and ac power ground.

Pin 2 ***Transmitted Data.** Data originated by the terminal to be transmitted via the sending modem.

Pin 3 ***Received Data.** Data from the receiving modem in response to analog signals transmitted from the sending modem.

Pin 4 ***Request to Send.** Indicates to the sending modem that the terminal is ready to transmit data.

Pin 5 ***Clear to Send.** Indicates to the terminal that its modem is ready to transmit data.

Pin 6 ***Data Set Ready.** Indicates to the terminal that its modem is not in a test mode and that modem power is ON.

Pin 7 ***Signal Ground.** Establishes common reference between the modem and the terminal.

Pin 8 ***Received Line Signal Detector.** Indicates to the terminal that its modem is receiving carrier signals from the sending modem.

Pin 9 Reserved for test.

Pin 10 Reserved for test.

Pin 11 Unassigned.

Pin 12 **Secondary Received Line Signal Detector.** Indicates to the terminal that its modem is recieving secondary carrier signals from the sending modem.

Pin 13 **Secondary Clear to Send.** Indicates to the terminal that its modem is ready to transmit signals via the secondary channel.

Pin 14 **Secondary Transmitted Data.** Data from the terminal to be transmitted by the sending modem's channel.

Pin 15 **Transmitter Signal Element Timing.** Signal from the modem to the transmitting terminal to provide signal element timing information.

Pin 16 **Secondary Received Data.** Data from the modem's secondary channel in response to analog signals transmitted from the sending modem.

Pin 17 **Receiver Signal Element Timing.** Signal to the receiving terminal to provide signal element timing information.

Pin 18 Unassigned.

Pin 19 **Secondary Request to Send.** Indicates to the modem that the sending terminal is ready to transmit data via the secondary channel.

Pin 20 ***Data Terminal Ready.** Indicates to the modem that the associated terminal is ready to receive and transmit data.

Pin 21 **Signal Quality Detector.** Signal from the modem telling whether a defined error rate in the received data has been exceeded.

Pin 22 Ring Indicator. Signal from the modem indicating that a ringing signal is being received over the line.

Pin 23 ***Data Signal Rate Selector.** Selects one of two signaling rates in modems having two rates.

Pin 24 **Transmit Signal Element Timing.** Transmit clock provided by the terminal.

Pin 25 Unassigned.

*Implemented in the HP 82939A Serial Interface.

Appendix B

# Registers

## Introduction

The serial I/O interface contains 24 distinct registers. These registers are accessed with STATUS or CONTROL statements. This discussion explains the contents of each register and shows the access procedures. If your system does not included an I/O ROM, you will need to substitute SET I/O for CONTROL statements. Status registers will not be able to be read.

## Register 0

To access this register execute:

```
STATUS 10,0;S
```

Register 0 is a read-only register. The value returned (always 2) indicates that this is the serial I/O interface.

## Register 1

**Interrupt Mask**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Break Received | Framing Error | Parity Error | Received Data Available | DCD (Opt. 001)<br>RTS (Standard) | Auto-disconnect | DSR (Opt. 001)<br>DRS (Standard) | CTS (Opt. 001)<br>DTR Standard) |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | **Value = 1** |

To access this register execute:

```
STATUS 10,1;S1
CONTROL 10,1;<value>
```

Register 1 is a read/write register. The value contained in this register determines the conditions that cause an interrupt to the computer. To specify interrupt conditions, execute the above CONTROL statement and specify the bit value (or values) for the desired interrupts. The value remains in this register until the interface is reset or until a new value is written to this register.

Reset default: 0 (all bits clear)

The meanings of the various bits are described next.

- Bit 7 set indicates than an interrupt is generated when a break is detected in the input data. A break is used to signal an interrupt to the remote device. A break appears to the serial interface as a null character, CHR$(0), and may cause parity and framing errors. Detecting a break may indicate that an erroneous null character has been input along with other valid data.
- Bit 6 set indicates that an interrupt is generated when a framing error is detected in the input data. A framing error may be caused by line transients or incorrectly specified stop bits or character length. The character that caused the framing error can be converted to a specific character if you desire (see Register 9).
- Bit 5 set indicates that an interrupt is generated when a parity error is detected in the input data. A parity error may be caused by line transients or when parity is incorrectly specified (see Register 4). The character that caused the parity error can be converted to a specific character if you so desire (see Register 9).
- Bit 4 set indicates that an interrupt is generated when there is received data in the interface input buffer. The data is then entered with an ENTER or TRANSFER statement.
- Bit 3 set indicates that the Data Carrier Detect (option 001 cable) or Request To Send (standard cable) modem signal has changed state. Note that the interrupt is generated when the modem signal changes from false to true or from true to false.
- Bit 2 set indicates that an interrupt is generated when an auto-disconnect is activated. An auto-disconnect is generated when specified modem signals are lost (see Register 5). Detecting an auto-disconnect when bit 2 is clear (0) generates error 115.
- Bit 1 set indicates that the Data Set Ready (option 001 cable) or Data Rate Select (standard cable) modem signal has changed state. Note that the interrupt is generated when the modem signal changes from false to true or from true to false.
- Bit 0 set indicates that the Clear To Send (option 001 cable) or Data Terminal Ready (standard cable) modem signal has changed state. Note that the interrupt is generated when the modem signal changes from false to true or from true to false.

## Register 2

**Modem Control Signals**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Not Used | Not Used | Not Used | Not Used | Not Used | DRS (Opt. 001)<br>DSR (Standard) | RTS (Opt. 001)<br>DCD (Standard) | DTR (Opt. 001)<br>CTS (Standard) |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
STATUS 10,2;S2
CONTROL 10,2;<value>
ASSERT 10;<value>
```

Register 2 is a read/write register. The value contained in this register controls the state of the interface control signals shown here. The state of these control signals is returned when the STATUS statement is executed. To change these signals, execute the above CONTROL or ASSERT statement and specify the bit value (or values) to set the desired lines. Executing the ASSERT statement immediately sets the modem signals as specified, regardless of any I/O operations that may be occuring. The value specified remains in this register until the interface is reset or until a new value is written to this register.

Reset default: 0 (all bits clear)

The meanings of the various bits are described next.

- Bits 7 through 3 are not used.
- Bit 2 set activates the Data Rate Select (option 001 cable) or Data Set Ready (standard cable) modem signal. To deactivate this signal, clear this bit (set to 0).
- Bit 1 set activates the Request To Send (option 001 cable) or Data Carrier Detect (standard cable) modem signal. To deactivate this signal, clear this bit (set to 0).
- Bit 0 set activates the Data Terminal Ready (option 001 cable) or Clear To Send (standard cable) modem signal. To deactivate this signal clear this bit (set to 0).

## Control Register 3

To access this register execute:

```
CONTROL 10,3; <value>
```

Control Register 3 is a write-only register. The value contained in this register determines the baud rate for transmitted and received data. This register selects standard baud rates only. Non-standard baud rates are specified by registers 6 and 7. To specify a standard baud rate, execute the above CONTROL statement and specify the desired value as shown in the table below. The value remains in this register until the interface is reset or until a new value is written to this register.

Reset default: 6 (300 baud)

| Value | Rate Specified |
|---|---|
| 0 | 50 |
| 1 | 75 |
| 2 | 110 |
| 3 | 134.5 |
| 4 | 150 |
| 5 | 200 |
| 6 | 300 |
| 7 | 600 |

| Value | Rate Specified |
|---|---|
| 8 | 1200 |
| 9 | 1800 |
| 10 | 2000 |
| 11 | 2400 |
| 12 | 2600 |
| 13 | 4800 |
| 14 | 7200 |
| 15 | 9600 |

## Status Register 3

**Modem Status and Cable Option**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Not Used | Not Used | Not Used | Not Used | DCD (Opt. 001)<br>RTS (Standard) | Cable Type | DSR (Opt. 001)<br>DRS (Standard) | CTS (Opt. 001)<br>DTR (Standard) |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
STATUS 10,3;S3
```

Status Register 3 is a read-only register. The value returned from this register indicates the status of the specified modem signals and type of cable installed on the interface.

Reset Default: None

The meanings of the various bits are described next.

- Bits 7 through 4 are not used.
- Bit 3 set indicates that the Data Carrier Detect (option 001 cable) or Request To Send (standard cable) modem signal is active (true). Bit 3 clear (0) indicates an inactive modem signal.
- Bit 2 set indicates that the interface has the standard cable installed. Bit 2 (0) clear indicates an option 001 cable.
- Bit 1 set indicates that the Data Set Ready (option 001 cable) or Data Rate Select (standard cable) modem signal is active (true). Bit 1 clear (0) indicates an inactive modem signal.
- Bit 0 set indicates that the Clear To Send (option 001 cable) or Data Terminal Ready (standard cable) modem signal is active (true). Bit 0 clear (0) indicates an inactive modem signal.

## Register 4

**Line Characteristics**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Not Used | Set Break | Force Parity | Odd/Even Parity | Enable Parity | Stop Bits | Character Length | |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
STATUS 10,4;S4
CONTROL 10,4;<value>
```

Register 4 is a read/write register. The value contained in this register specifies character length, stop bits, parity, and forces a **break** to be transmitted. To change these character definitions, execute the above CONTROL statement and specify the bit value (or values) as desired. The value remains in this register until the interface is reset or until a new value is written to this register. A break is sent by setting bit 6 with a CONTROL statement or by executing the REQUEST statement. The REQUEST statement causes the transmit line to be held in the space condition for the amount of character times specified for the value parameter, followed by holding the transmit line in the mark condition for at least 5 character times. When using the CONTROL statement to set a break, the line remains in the space condition until bit 6 is cleared.

Reset default: 10 (seven bits, odd parity, one stop bit)

The meanings of the various bits are described next.

- Bit 7 is not used.
- Bit 6 set causes a break signal to be output to the remote device. A break forces the transmit line to the space condition. The transmit line remains in the space condition until bit 6 is cleared.
- Bits 5, 4, and 3 are defined in the following table.

| Value | Bits | | | Parity Specified |
|---|---|---|---|---|
| | 5 | 4 | 3 | |
| 0 | 0 | 0 | 0 | No parity bit |
| 8 | 0 | 0 | 1 | Odd parity |
| 24 | 0 | 1 | 1 | Even parity |
| 40 | 1 | 0 | 1 | Always 1 |
| 56 | 1 | 1 | 1 | Always 0 |

- Bit 2 set specifies two stop bits. Bit two clear (0) specifies one stop bit.
- Bits 1 and 0 specify character length as shown in the following table:

| Value | Bits | | Character Length |
|---|---|---|---|
| | 1 | 0 | |
| 0 | 0 | 0 | 5 |
| 1 | 0 | 1 | 6 |
| 2 | 1 | 0 | 7 |
| 3 | 1 | 1 | 8 |

The next table shows how to select the value for the CONTROL statement to specify combinations of character length, parity, and stop bits.

| Bits/Character | Parity Specifier | | | | |
|---|---|---|---|---|---|
| | None | Odd | Even | 1 | 0 |
| 5 | 0 | 8 | 24 | 40 | 56 |
| 6 | 1 | 9 | 25 | 41 | 57 |
| 7 | 2 | 10 | 26 | 42 | 58 |
| 8 | 3 | 11 | 27 | 43 | 59 |

For example, you want to select 7 bits per character, odd parity. From the table, you find the value 10 specifies this configuration. The statement required to specify this configuration is:

```
CONTROL 10,4;10
```

**Note:** The table shown above specifies 1 stop bit. To specify 2 stop bits, add 4 to the values found in the table.

## Register 5

**Modem Features**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Not Used | Not Used | Receive Handshake | Transmit Handshake | DCD (Opt. 001)<br>RTS (Standard) | Not Used | DSR (Opt. 001)<br>DRS (Standard) | CTS (Opt. 001)<br>DTR (Standard) |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
STATUS 10,5;S5
CONTROL 10,5;<value>
```

Register 5 is a read/write register. The value contained in this register determines the auto-handshake and auto-disconnect features that are specified. To change the specifications, execute the above CONTROL statement and specify the bit value (or values) as desired. The value remains in this register until the interface is reset or until a new value is written to this register. Note that bits 5 and 4 can also be controlled by the reset defaults (see the Installation section of this manual).

Reset default: 0 (all bits clear)

The meanings of the various bits are described next.

- Bits 7 and 6 are not used.

- Bit 5 set enables the receive auto-handshake feature. This feature discards received data and character status information if the Data Carrier Detect (option 001 cable) or Request To Send (standard cable) modem signals are inactive. Bit 5 clear (0) disables this feature:
- Bit 4 set enables the transmit auto-handshake feature. This feature suspends data transmission when the Clear to Send (option 001 cable) or Data Terminal Ready (standard cable) modem signals are inactive. Data transmission resumes when this bit is clear (0). See Printer Interfacing for an example use of this feature.
- Bit 3 set enables the auto-disconnect feature. An auto-disconnect is generated when the Data Carrier Detect (option 001 cable) or Request To Send (standard cable) modem signal becomes inactive. Auto-disconnect generates error 115 unless an interrupt is specified (see Register 2, bit 2). Bit 3 clear (0) disables this feature.
- Bit 2 is not used.
- Bit 1 set enables the auto-disconnect feature. An auto-disconnect is generated when the Data Set Ready (option 001 cable) or Data Rate Select (standard cable) modem signal becomes inactive. Auto-disconnect generates error 115 unless an interrupt is specified (see Register 2, bit 2). Bit 1 clear (0) disables this feature.
- Bit 0 set enables the auto-disconnect feature. An auto-disconnect is generated when the Clear To Send (option 001 cable) or Data Terminal Ready (standard cable) modem signal becomes inactive. Auto-disconnect generates error 115 unless an interrupt is specified (see register 2, bit 2). Bit 0 clear (0) disables this feature.

## Register 6

To access this register execute:

```
STATUS 10,6;S6
CONTROL 10,6;<value>
```

## Register 7

To access this register execute:

```
STATUS 10,7;S7
CONTROL 10,7;<value>
```

Registers 6 and 7 are read/write registers. The value contained in these registers indicates the value specified for the transmit and receive data transfer rates. These registers are used to specify non-standard baud rates (see Register 3 for standard baud rate specifiers). The values for these registers are determined as follows:

1. Rate = 115 200/Divisor
2. Divisor = ((Register 6 value × 256) + (Register 7 value))

The default values for these two registers specify 300 baud (see register 3). These default values are 1 for Register 6 and 128 for Register 7. Entering these values into step 2 we obtain the following results:

Divisor = ((1 × 256) + (128)) = 384

Rate = 115 200/384 = 300

To specify a non-standard baud rate, the values for Registers 6 and 7 are determined as follows:

1. Divisor = 115 200/Rate
2. Register 7 value = (Divisor) MOD 256
3. Register 6 value = (Divisor) DIV 256

For example, you want to specify a baud rate of 275 bits per second. Calculate the register values as follows:

1. Divisor = 115 200/275 = 418
2. Register 7 value = (418) MOD 256 = 162
3. Register 6 value = (418) DIV 256 = 1

To specify a baud rate of 275 execute the following CONTROL statement to Registers 6 and 7:

```
CONTROL 10,6;1,162
```

Reset defaults:

Register 6 = 1
Register 7 = 128

## Register 8

To access this register execute:

```
STATUS 10,8;S8
CONTROL 10,8;<value>
```

Register 8 is a read/write register. The value contained in this register is the decimal value of the ASCII character that is specified as the parity and framing error replacement character. When a parity or framing error is detected in the input data, the character that contained the error can be replaced with the character in this register. This feature is used to flag errors and simplify troubleshooting. This feature is enabled by bit 4, Register 9.

Reset default: 0 (all bits clear)

## Register 9

**Transmitter/Receiver Control**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Enable Transmitter | Strip Received Rubouts | Strip Received Nulls | Change Character if Error | Set Bit 7 of Character if Error | Reset Receive Queue | Auto-echo Enable | Enable Receiver |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
STATUS 10,9;S9
CONTROL 10,9;<value>
```

Register 9 is a read/write register. This register is the primary control register for interface transmitter and receiver. The value contained in this register indicates the state of the various control signals. To change these control signals, execute the above CONTROL statement and specify the bit value (or values) to activate the desired control signals. The value remains in this register until the interface is reset or until a new value is written to this register. Executing the RESUME statement immediately enables the transmitter (sets bit 7). The RESUME statement may be executed concurrent with active transfers.

Reset default: 137 (bits 0, 3, and 7 set)

The meanings of the various bits are described next.

- Bit 7 set enables the data transmitter. When bit 7 is clear, only echo data is transmitted (see bit 1). Bit 7 may be controlled by the XON/XOFF feature provided by Control Register 11.
- Bit 6 set allows received rubout characters, CHR$(127), to be stripped from incoming data. This feature is used when entering data from a device that sends extra rubout characters to avoid buffer overruns. Bit 6 clear (0) disables this feature.
- Bit 5 set allows received null characters, CHR$(0), to be stripped from incoming data. This feature is used when entering data from a device that sends extra null characters to avoid buffer overruns. Bit 5 clear (0) disables this feature.
- Bit 4 set enables replacement of characters received with parity or framing errors. The replacement character is defined by Register 8. Bit 4 clear (0) disables this feature.
- Bit 3 set enables the underline feature for parity and framing errors. This feature sets bit 7 of Register 8. This bit set causes all characters with parity or framing errors to be underlined when they are displayed on the internal CRT or are printed on the internal printer. Bit 3 clear (0) disables this feature.
- Bit 2 set clears the receive data queue. When the receive data queue is cleared, this bit is automatically reset by the interface. All data in the receive queue is lost when this feature is activated. Ensure that values written to this register are correct before you execute the CONTROL statement.
- Bit 1 set enables the auto-echo feature. This feature causes all received characters to be retransmitted to the sending device. Bit 1 clear (0) disables this feature.
- Bit 0 set enables the receiver. All input data, including status, is entered into the interface input buffer. Bit 0 clear (0) disables the receiver. All input data, including status and break is ignored.

## Control Register 10

Control register 10 is not implemented. Executing a CONTROL statement to this register generates error 111.

## Status Register 10

**Line Status**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Not Used | Not Used | Transmit Register Empty | Break Received | Framing Error | Parity Error | Not Used | Received Data Available |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
STATUS 10,10;S0
```

Status Register 10 is a read-only register. The value contained in this register provides information about each character as it is received. This status information is entered into the receive queue in synchronization with each character.

Reset default: 0 (all bits clear)
The meanings of the various bits are described next.

- Bits 7 and 6 are not used.
- Bit 5 set indicates that the transmitter is ready to accept another character from the computer. This does NOT mean that all data has been transmitted to the remote device. The data may still be in the interface serial output register.
- Bit 4 set indicates that a break from the remote has been detected.
- Bit 3 set indicates that a framing error has been detected for at least one input character. Framing errors usually indicate a line transient, an incorrect number of stop bits, or an incorrect number of bits per character.
- Bit 2 set indicates that a parity error has been detected for at least one input character. Parity errors indicate a line transient, an incorrect number of bits per character, or an incorrect parity specifier.
- Bit 1 is not used.
- Bit 0 set indicates that received data is available in the input queue. The received data should be entered into the computer.

## Control Register 11

**Input Data Control**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Enable Transmit Flag XON | Disable Transmit Flag XOFF | Not Used | Terminate if CR15 (see CR15) | Terminate if CR14 (see CR14) | Terminate if CR13 (see CR13) | Terminate if CR12 (see CR12) | Not Used |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
CONTROL 10,11;<value>
```

This register specifies the input terminator character location. This register also provides control of the transmitter enable flag when input terminator characters are detected. To specify the input terminator character location, or to control the transmitter enable flag, execute the above CONTROL statement and specify the bit values for the desired characters or flag state. The value remains in this register until the interface is reset or until a new value is written to this register.

Reset default: 0 (all bits clear)

The meanings of the various bits are described next.

- Bit 7 set specifies that the transmitter enable flag is set when the character defined by Control Register 15 is detected in the input data stream (XON).
- Bit 6 set clears the transmitter enable when the character defined by Control Register 14 is detected in the input data stream (XOFF).
- Bit 5 is not used.

- Bit 4 set specifies that input operations are terminated when the character defined by Control Register 15 is detected in the input data stream.*
- Bit 3 set specifies that input operations are terminated when the character defined by Control Register 14 is detected in the input data stream.*
- Bit 2 set specifies that input operations are terminated when the character defined by Control Register 13 is detected in the input data stream.*
- Bit 1 set specifies that input operations are terminated when the character defined by Control Register 12 is detected in the input data stream.*
- Bit 0 is not used.

*Input operations can ONLY be terminated by these characters if the % image specifier is used.

## Status Register 11

**I/O Termination Cause**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| End of Output Data List | End of Input Data List | Transfer Count Expired | CR15 Character Received | CR14 Character Received | CR13 Character Received | CR12 Character Received | DELIM Character Received |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
STATUS 10,11;R1
```

Status Register 11 is a read-only register. The value returned from this register indicates the reason for terminating an I/O operation. When multiple termination conditions are specified, this register is read to determine the I/O termination cause.

Reset default: 0 (all bits clear)

The meanings of the various bits are explained next.

- Bit 7 set indicates the the I/O operation was terminated by the computer at the end of the output list.
- Bit 6 set indicates that the I/O operation was terminated by the computer when the input list was filled.
- Bit 5 set indicates that the input TRANSFER operation was terminated by satisfying the COUNT parameter.
- Bit 4 set indicates that the input operation was terminated when the character defined by Control Register 15 was detected in the input data stream.
- Bit 3 set indicates that the input operation was terminated when the character defined by Control Register 14 was detected in the input data stream.
- Bit 2 set indicates that the input operation was terminated when the character defined by Control Register 13 was detected in the input data stream.
- Bit 1 set indicates that the input operation was terminated when the character defined by Control Register 12 was detected in the input data stream.
- Bit 0 set indicates than an input TRANSFER operation was terminated when the DELIM character was detected in the input data stream.

## Control Registers 12 through 15

Control Registers 12, 13, 14, and 15 contain the termination characters that are specified by Control Register 11. The transmit enable flag is also controlled by the characters specified by Control Registers 14 and 15. To define termination characters for these registers, execute a CONTROL statement to the desired register and specify the decimal value of the ASCII character defined as the termination character.

To access these registers execute:

```
CONTROL 10,12;<value>
CONTROL 10,13;<value>
CONTROL 10,14;<value>
CONTROL 10,15;<value>
```

## Control Register 16

**Output EOL Sequence**

Most Significant Bit — Least Significant Bit

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| Auto RTS Enable | EOL Transmit Disable | Six Bit EOL Character Count | | | | | |
| Value = 128 | Value = 64 | Value = 32 | Value = 16 | Value = 8 | Value = 4 | Value = 2 | Value = 1 |

To access this register execute:

```
CONTROL 10,16;<value>
```

Control Register 16 is a write-only register. This register contains the EOL sequence character count, the EOL transmitter control, and the auto Request To Send enable. The EOL is a character or sequence of characters that is output at the end of each line of data (e.g., CR/LF). To change the EOL character count or to enable the auto Request To Send or EOL transmit disable, execute the above CONTROL and specify the bit value as desired. The new value remains in this register until the interface is reset or until a new value is written to this register.

Reset default: 2 (bit 1 set)

The meanings of the various bits are described next.

- Bit 7 set enables the auto Request To Send feature. This feature activates the Request To Send modem signal at the beginning of each transmission. When the EOL sequence is transmitted (end of TRANSFER or OUTPUT statement) the Request To Send signal is set to the inactive state. Bit 7 clear disables this feature.
- Bit 6 set clears the transmitter enable flag after all output data (including EOL sequences) has been transmitted. This flag must be reenabled from the program or by detecting a special received character (see Control Register 11) before transmission can resume. This feature is used with ENQuire/ACKnowledge handshakes. Bit 6 clear (0) disables this feature.
- Bits 5 through 0 contain the EOL character count value. This value specifies how many characters are defined for the EOL sequence. This value ranges from 0 through 63. Long EOL sequences are normally used with slow printing devices such as teletypes to allow the print mechanism sufficient time to execute a carriage return/line feed before sending more data from the interface. The characters that make up the actual End-Of-Line sequence are contained in Control Registers 17 through 23. The EOL character count parameter and the actual character sequences are explained following Control Registers 17 through 23.

## Control Registers 17 through 23

Control Registers 17 through 23 contain the user defined output EOL sequence. You define the EOL sequence by specifying a maximum of seven separate characters. The EOL characters are defined by executing a CONTROL statement to the desired register and specifying the decimal value of the desired ASCII character.

To access these registers execute:

```
CONTROL 10,17;<value>
CONTROL 10,18;<value>
CONTROL 10,19;<value>
CONTROL 10,20;<value>
CONTROL 10,21;<value>
CONTROL 10,22;<value>
CONTROL 10,23;<value>
```

Reset defaults:

Register 17—CR (CHR$(13))
Register 18—LF (CHR$(10))
Registers 19 through 23—Null (CHR$(0))

The default value for Control Register 16 is 2. This specifies a two character EOL sequence. This default and the default for Control Registers 17 and 18 mean that the default EOL sequence is a CR/LF. The EOL sequence definition is changed as shown next.

| Sequence | Control Reg 16 | Control Regs 17 through 23 |
|---|---|---|
| CR-LF-Null | 3 | Reg 17 = 13<br>Reg 18 = 10<br>Reg 19 = 0 |
| CR-LF-NULL<br>Null-RO-RO | 6 | Reg 17 = 13<br>Reg 18 = 10<br>Reg 19 = 0<br>Reg 20 = 0<br>Reg 21 = 255<br>Reg 22 = 255 |
| CR-LF-Null-Null<br>RO-RO-followed<br>by 15 Nulls | 21* | Reg 17 = 13<br>Reg 18 = 10<br>Reg 19 = 0<br>Reg 20 = 0<br>Reg 21 = 255<br>Reg 22 = 255<br>Reg 23 = 0 |

*When the value for Register 16 exceeds 7, the characters defined by Registers 17 through 23 are output first, followed by continuous Register 23 characters until the count value is satisfied.

## Table B-1. Status Register Contents

| Status Register Number | Bit Number 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Default Value (Decimal) | Register Function |
|---|---|---|---|---|---|---|---|---|---|---|
| SR0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 2 | Interface identification |
| SR1 | Break received | Framing error | Parity error | Receive data available | DCD (RTS) | Auto-disconnect | DSR (DRS) | CTS (DTR) | 0 | Interrupt cause |
| | | | | | | DRS (DSR) | RTS (DCD) | DTR (CTS) | 0 | Modem control lines |
| SR3 | | | | | DCD (RTS) | Cable type | DSR (DRS) | CTS (DTR) | Variable | Modem status lines |
| SR4 | | Set break | Force parity | Odd/even parity | Enable parity | Stop bits | Character length | | Variable | Line characteristics |
| SR5 | | | Receive handshake | Transmit handshake | DCD (RTS) | | DSR (DRS) | CTS (DTR) | Variable | Modem features |
| SR6 | Most significant byte | | | | | | | | Variable | Baud rate |
| SR7 | Least significant byte | | | | | | | | Variable | Divisor latches |
| SR8 | Parity/framing error replacement character | | | | | | | | 0 | Error replacement character |
| SR9 | Enable transmitter | Strip received rubouts | Strip received nulls | Change char if error | Set bit 7 or char if error | Reset receive queue | Auto-echo enable | Enable receiver | 137 (10001001) | Transmitter/ receiver control |
| SR10 | | | Transmit register empty | Break received | Framing error | Parity error | | Received data available | 32 | Line status |
| SR11 | END of output data list | End of input data list | Transfer count expired | CR15 character received | CR14 character received | CR13 character received | CR12 character received | DELIM character received | 0 | I/O Termination cause |

## Table B-2. Control Register Contents

| Control Register Number | Bit Number 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Default Value (Decimal) | Register Function |
|---|---|---|---|---|---|---|---|---|---|---|
| CR1 | Break received | Framing error | Parity error | Receive data available | DCD (RTS) | Auto-disconnect | DSR (DRS) | CTS (DTR | 0 | Interrupt mask |
| CR2 | | | | | | DRS (DSR) | RTS (DCD) | DTR (CTS) | 0 | Modem control lines |
| CR3 | Baud rate number | | | | | | | | Variable | Baud rate select |
| CR4 | | Set break | Force parity | Odd-even parity | Enable parity | Stop bits | Character length | | Variable | Line characteristics |
| CR5 | | | Receive handshake | Transmit handshake | DCD (RTS) | | DSR (DRS) | CTS (DTR) | Variable | Modem features |
| CR6 | Most significant byte | | | | | | | | Variable | Baud rate |
| CR7 | Least significant byte | | | | | | | | Variable | Divisor latches |
| CR8 | Parity/framing error replacement character | | | | | | | | 0 | Error replacement character |
| CR9 | Enable transmitter | Strip received rubouts | Strip received nulls | Change char if if error | Set bit 7 of char if error | Reset receive queue | Auto-echo enable | Enable receiver | 137 (10001001) | Transmitter/ receiver control |
| CR11 | Transmit ON | Transmit OFF | | Terminate if CR15 | Terminate if CR14 | Terminate if CR13 | Terminate if CR12 | | 0 | Input data control |

| Control Register Number | Bit Number 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Default Value (Decimal) | Register Function |
|---|---|---|---|---|---|---|---|---|---|---|
| CR12 | Input termination character | | | | | | | | 0 | Input termination character #1 |
| CR13 | Input termination character | | | | | | | | 0 | Input termination character #2 |
| CR14 | Input termination character | | | | | | | | 0 | Input termination character #3 |
| CR15 | Input termination character | | | | | | | | 0 | Input termination character #4 |
| CR16 | Auto-RTS enable | EOL transmit disable | Six bit EOL character count | | | | | | 2 | Output EOL sequence |
| CR17 | EOL character | | | | | | | | 13 (CR) | EOL character #1 |
| CR18 | EOL character | | | | | | | | 10 (LF) | EOL character #2 |
| CR19 | EOL character | | | | | | | | 0 | EOL character #3 |
| CR20 | EOL character | | | | | | | | 0 | EOL character #4 |
| CR21 | EOL character | | | | | | | | 0 | EOL character #5 |
| CR22 | EOL character | | | | | | | | 0 | EOL character #6 |
| CR23 | EOL character | | | | | | | | 0 | EOL character #7 |

Appendix C

# Functional Description

## Introduction

This appendix describes the HP 82939A Serial Interface hardware. Most of the interface functions are performed by a ROM program internal to a microcomputer. Therefore, the interface is almost entirely a software device. Refer to the operation and programming section for the software theory of operation.

The interface consists of three major components: the translator, the microcomputer, and the UART. The translator functions as an interface between the mainframe's internal bus structure and the interface's internal bus structure; the UART converts parallel data to serial, and vice versa, for communication with the remote device; and the microcomputer controls and synchronizes the operation of both, on instructions from the mainframe and from its own internal ROM program.

The operation of these three components will be examined in greater detail, then data will be followed as it passes through the receiving and transmitting circuits to see how each component handles it.

In the discussions that follow, **CPU** will refer to the HP Series 80 Personal Computer whereas **microcomputer** will refer to the microprocessor internal to the interface itself.

## Translator Function

The translator functions as an interface, or translator, between the computer's internal bus structure and the interface's internal bus structure. It contains two registers which can be written to by the computer CPU and read by the interface microcomputer (one for data and one for status), and two which can be written to by the interface microcomputer and read by the HP computer CPU (again one for data and one for status). All four registers are eight bits wide. It also contains control circuitry for manipulating these registers to produce the desired results.

These registers can be described as follows:

### Output Buffer (OB)

The output buffer is used by the HP Series 80 Personal Computer CPU to send data to the interface microcomputer. It is written to by the CPU and read by the microcomputer. When the CPU writes to the OB, it sets the Output Buffer Full (OBF) flag. The microcomputer clears the OBF flag when it reads the OB.

### Input Buffer (IB)

The input buffer is identical to the OB except that the data flows in the opposite direction. It is written to by the microcomputer and read by the CPU. The microcomputer sets the Input Buffer Full (IBF) flag when it writes to the IB, and the CPU clears the IBF flag when it reads the IB.

### Control Register (CR)

The control register is used to send control information from the CPU to the microcomputer. The middle six bits (bits 1 to 6) are similar to the OB in that they are written by the CPU and read by the

microcomputer. Bits 0 and 7 are two-function bits. When the CPU writes the CR, the middle six bits go straight through to be read by the microcomputer, but the outer two are diverted. Bit 0 becomes the INT signal, which is used to interrupt the microcomputer. Bit 7 becomes RESET, which is used to re-set the microprocessor. When the microcomputer reads the CR, it reads the IBF flag as bit 0 and the OBF flag as bit 7.

### Status Register (SR)

The status register is similar to the CR except that status information is sent from the microcomputer to the CPU. The middle six bits are written by the microcomputer and read by the CPU. Bits 0 and 7 are two-function bits. When written by the microcomputer, bit 0 becomes SERVICE, which is used to interrupt the CPU. Bit 7 becomes HLTEN, which is used to halt the CPU. When the CPU reads the SR, it reads the IBF flag as bit 0 and the OBF flag as bit 7.

These four registers all share the same bus lines; that is, there is only one 8-bit bus connecting the HP Series 80 Personal Computer internal bus lines and the microcomputer. Control and status information and data all flow along this bus and are directed to the proper register by address logic.

## Microcomputer Function

The controlling entity of the HP 82939A Serial Interface is an 8049 single-chip microcomputer. A major portion of this microcomputer is a ROM which contains a microcode which determines the entire behavior of the interface. This microcode is not changeable by the user, however, many variables can be changed, either by switches (see Section 2) or by software (see operation and programming sections of this manual). The microcomputer also contains a 5.5296 MHz clock for internal timing, which is divided by three and passed to the UART for internal timing. The microcomputer has one 8-bit-wide data port which functions for both input and output.

Internal to the microcomputer are 12, 8-bit status registers and 22, 8-bit control registers. The status registers contain information indicating the current state of the microcomputer. Table B-1 shows the location of this information. The control registers are written to whenever a change in the microcomputer state is desired. Table B-2 lists the functions of the control registers.

## UART Function

The Universal Asynchrounous Receiver/Transmitter (UART) is the device which does the actual serial-to-parallel conversion. Under control of the microcomputer, it converts incoming serial data streams into parallel data for transmission to the mainframe. It also converts parallel data into serial data streams suitable for RS-232C transmission.

The UART can be programmed by the microcomputer to cause an interrupt on various conditions.

## Data Flow Through the Interface

Having looked at the operation of the major components of the interface, we will now trace the path which data takes as it passes through the interface. Refer to figure C-1 while reading this description.

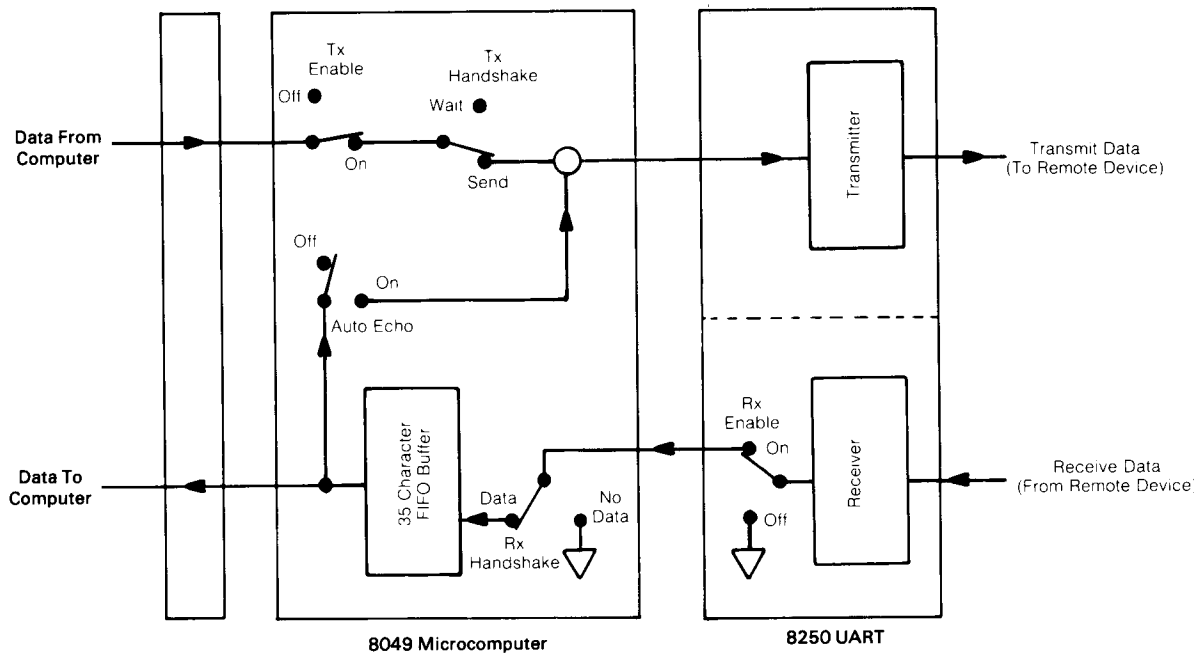



**Figure C-1. Data Flow Diagram**

As data is received from a remote device, the receiver checks it for errors and converts it to parallel form. If the user has enabled the receiver via the receiver control register (Control Register 9, bit 0), the UART will interrupt the microcomputer to indicate that it has a data byte ready. If the user has not enabled the receiver, the data byte will be discarded.

When interrupted for received data, the microcomputer checks the receive autohandshake bit (Control and Status Register 5, bit 5) and the data carrier detect (Modem Status Register 3, bit 0) modem status line to determine if the received data is valid. If the user has specified the receive handshake feature and DCD is "OFF", the received data will be discarded. Otherwise, the data will be pushed into the queue buffer. If the data was received with a parity or framing error, this information will also be pushed into the queue buffer.

A received BREAK follows essentially the same path as received data. If a BREAK is received, an indicator will be pushed into the queue buffer if the receiver is enabled and the autohandshake criteria are met.

All received data and status are passed through the queue buffer, as described here, to keep correct time sequence.

**Note:** Data and status will be saved in the buffer whether or not there is an input data statement active in the computer.

The background idle loop in the interface microcomputer pulls data and status information from the queue buffer and takes appropriate action. This idle loop handles auto echo, data transfer, status and interrupts. If the auto echo feature is enabled, the receive data is sent back to the UART for re-transmission. Since data will not be sent to the mainframe until it has been re-transmitted, the auto echo feature may affect performance.

Received data is input to computer from the queue buffer using either the ENTER or TRANSFER I/O statements. The termination conditions, such as count and character match, are checked as the data is transferred to the mainframe.

Receive status information, such as BREAK or data errors, can be read via the Line Status Register (#10) and, depending on the state of the interrupt mask (Control Register 1), may cause a user interrupt.

Due to the nature of the queue buffer, if a data byte is in the queue, status information which follows it will not be seen until this data has been sent to the mainframe. For example, if an input is not active, a break will not be reported if a data byte was received before the break. (The idle loop will not look ahead beyond the received character to perceive the break.)

This is the procedure which the interface uses for receiving data. The data transmit path will be presented next.

When the UART transmit buffer is empty, the microcomputer checks the transmit autohandshake bit (Control Register 5, bit 4) and the Clear to Send signal (Modem Status Register 3, bit 0) to determine if the remote device is ready to accept data. If the user has enabled the transmit autohandshake feature and CTS is "OFF", no data will be sent. Otherwise, the microcomputer will look for data to transmit. Characters to be echoed get the highest priority for transmission. If auto echo is enabled, the queue buffer will be scanned for echo data to be sent to the UART transmitter. If there is no echo data, data from the mainframe will be sent if there is an OUTPUT or TRANSFER statement active.

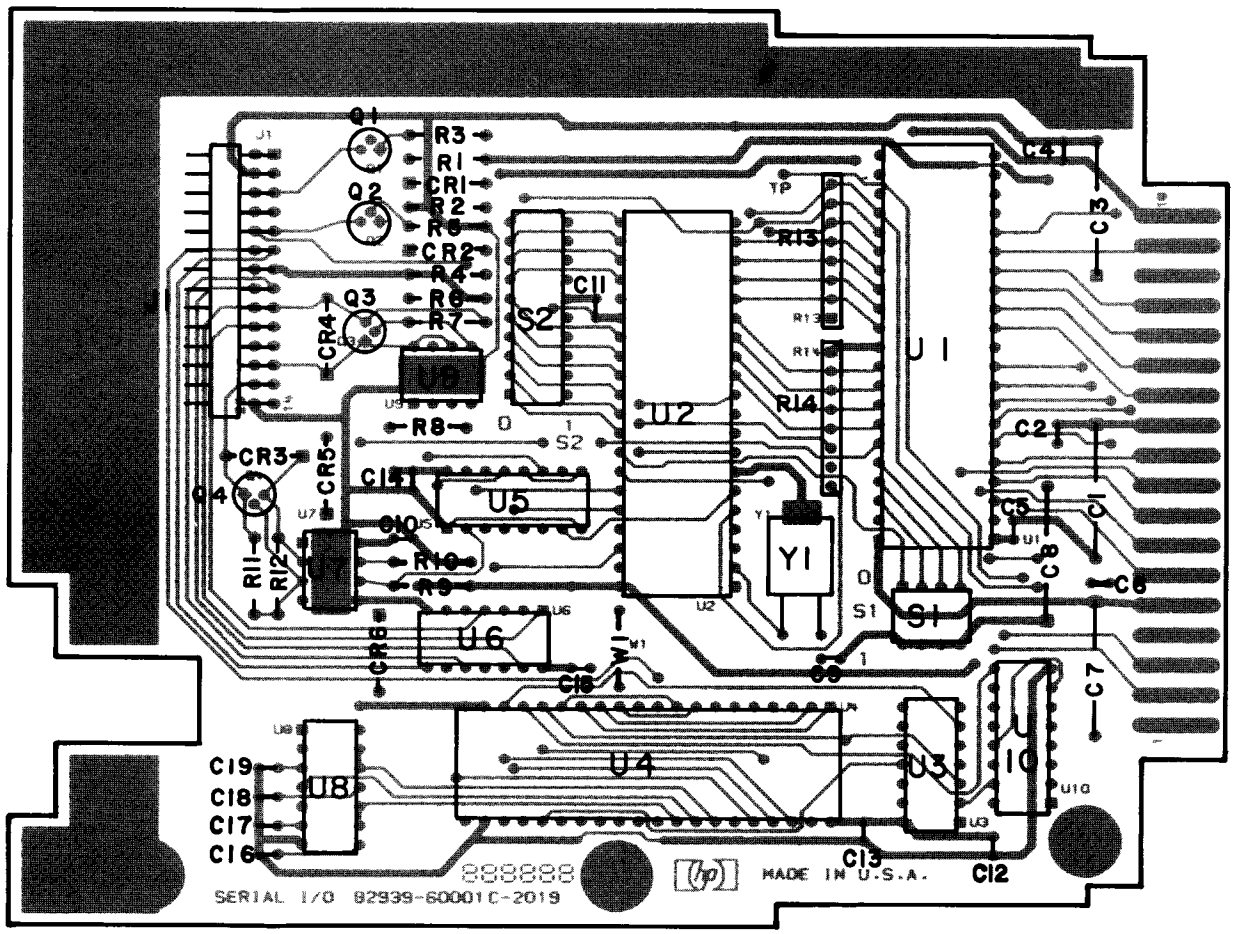
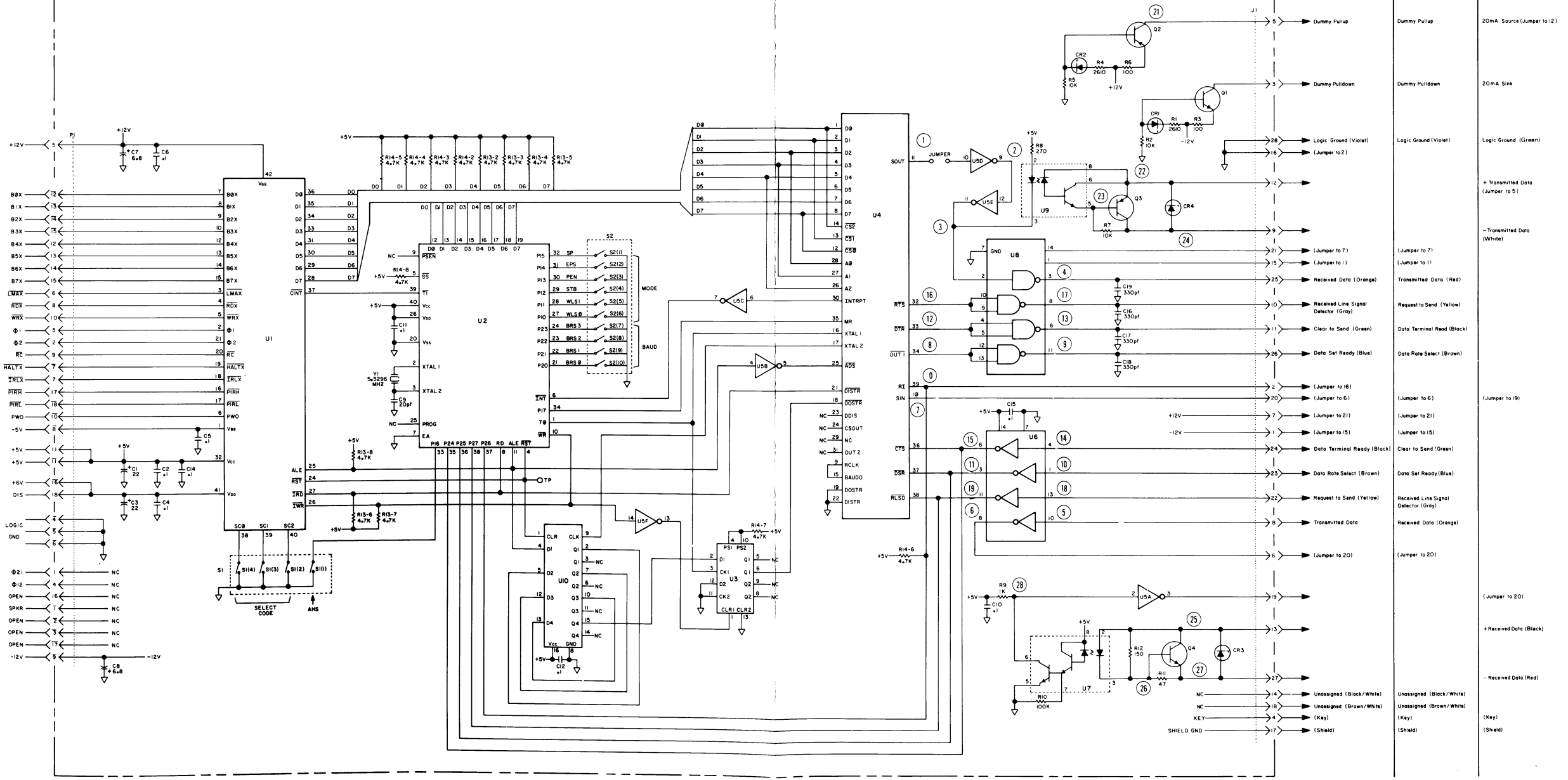

Q1
Q2
Q3
Q4
R3
R1
CR1
R2
R6
CR2
R4
R5
R7
R8
CR4
CR3
CR5
R11
R12
R10
R9
CR6
C14
C10
C11
C15
C16
S2
U2
U5
U6
U7
U8
U4
U3
U1
U10
TP
R13
R14
Y1
S1
C19
C18
C17
C16
C13
C12
C4
C3
C2
C5
C1
C8
C6
C7
888888
SERIAL I/O 82939-60001C-2019
MADE IN U.S.A.

**Table C-1. HP 82939 A Serial Interface Replaceable Parts List**

| Reference Designator | CD | HP Part. NO. | TQ | Description |
|---|---|---|---|---|
| A1 | 8 | 82939-60901 | 1 | PC Assembly |
| | 9 | 82939-60902 | 1 | Case, Complete, Standard |
| | 0 | 82939-60903 | 1 | Case Complete, Option 1 |
| | 1 | 82939-60904 | 1 | Case, Complete, Option 2 |
| | 0 | 8120-3248 | 1 | Cable, Complete, Standard |
| | 1 | 8120-3247 | 1 | Cable, Complete, Option 1 |
| | 2 | 8120-3191 | 1 | Cable, Complete, Option 2 |
| | 3 | 07221-60157 | 1 | Cable, Male to Male, RS232 |
| C1 | 6 | 0180-0228 | 2 | C-F: 22μF, 15V |
| C2 | 5 | 0160-4571 | 10 | C-F: .1μF, 50 V |
| C3 | 6 | 0180-0228 | — | C-F: 22μF, 15V |
| C4 – C6 | 5 | 0160-4571 | — | C-F: .1μF, 50V |
| C7 – C8 | 1 | 0180-0116 | 2 | C-F: 6.8μF, 35V |
| C9 | 4 | 0160-4767 | 1 | C-F: 20pF, 200V |
| C10 – C15 | 5 | 0160-4571 | — | C-F: .1μF, 50V |
| C16 – C19 | 4 | 0160-3694 | 4 | C-F: 330pF, 100V |
| CR1 – CR14 | 1 | 1901-1098 | 4 | Diode: Switching |
| MP1 | 9 | 0340-0883 | 4 | Transistor Insulator |
| Q1 | 7 | 1854-0477 | 2 | Transistor: NPN, 2N2222A |
| Q2 – Q3 | 9 | 1853-0281 | 2 | Transistor: PNP, 2N2907A |
| Q4 | 7 | 1854-0477 | — | Transistor: NPN, 2N2222A |
| R1 | 0 | 0698-0085 | 2 | F-F: 2610Ω, 1%, .125W |
| R2 | 9 | 0757-0442 | 2 | R-F; 10kΩ, 1%, .125W |
| R3 | 0 | 0757-0401 | 2 | R-F: 100Ω, 1%, .125W |
| R4 | 0 | 0698-0085 | — | R-F: 2610Ω, 1%, .125W |
| R5 | 9 | 0757-0442 | — | R-F: 10kΩ, 1%, .125W |
| R6 | 0 | 0757-0401 | — | R-F: 100Ω, 1%, .125W |
| R7 | 1 | 0683-1035 | 1 | R-F: 10kΩ, 5%, .25W |
| R8 | 6 | 0683-2715 | 1 | R-F: 270Ω, 5%, .25W |
| R9 | 9 | 0683-1025 | 1 | R-F: 1kΩ, 5%, .25W |
| R10 | 3 | 0683-1045 | 1 | R- F; 100kΩ, 5%, .25W |
| R11 | 8 | 0683-4705 | 1 | R-F: 47Ω, 5%, .25W |
| R12 | 2 | 0683-1515 | 1 | R-F: 150Ω, 5%, .25W |
| R13, R14 | 7 | 1810-0205 | 2 | Resistor Network, 4.7kΩ |
| S1 | 8 | 3101-2533 | 1 | Switch, 4 Segment SPST |
| S2 | 2 | 3101-2532 | 1 | Switch, 10 Segment SPST |
| U1 | 3 | 1MB5-0101 | 1 | IC: Translator |
| U2 | 3 | 1820-2438 | 1 | IC: MCU, 11MHz |
| U3 | 8 | 1820-1112 | 1 | IC: SN74LS74N |
| U4 | 0 | 1820-2443 | 1 | IC: 8250 |
| U5 | 7 | 1820-1492 | 1 | IC: SN74LS368AN |
| U6 | 8 | 1820-0990 | 1 | IC: MC1489AL |
| U7 | 6 | 1990-0494 | 1 | Opto-Isolator |
| U8 | 6 | 1820-0509 | 1 | IC: MC1488L |
| U9 | 6 | 1990-0444 | 1 | Opto-Isolator |
| U10 | 7 | 1820-1195 | 1 | IC: SN74LS175N |
| Y1 | 0 | 0410-1221 | 1 | XTAL: 5.5296 MHz |
| | 9 | 0590-0199 | 2 | Hex Nut with Lock Washer |
| | 0 | 2200-0143 | 6 | Screws: 4-40 Machine |
| | 9 | 0363-0174 | 1 | Contact, Ground |
| | 7 | 1400-1062 | 1 | Clamp: Cable |
| | 8 | 1400-1063 | 1 | Clamp: Cable |

A1 82939-60901 RS-232 INTERFACE ASSEMBLY



Appendix D

# Error Messages

The Serial Interface generates error messages when operation of the card violates certain conditions. The table below contains a list of these error messages.

### Summary of Error Messages

**System Errors (common to all I/O cards)**

| Error Number | Error |
|---|---|
| 110 | Hardware self-test failed (RAM, ROM, ALU, UART). |
| 111 | Illegal operation. |

**Interface Dependent Errors (unique to Serial Interface)**

| Error Number | Error |
|---|---|
| 113* | UART receiver overrun—data lost. |
| 114* | Rx queue buffer overrun—data lost. |
| 115* | Automatic disconnect forced. |

*Aborts TRANSFERs in progress.

## Troubleshooting Hints

| Problem | Probable Cause |
|---|---|
| OUTPUT and ENTER operations hang. | Ensure auto-handshake feature is disabled if not required. |
| Random unintelligible data. OUTPUT completes, but ENTER does not. | Incorrect Baud rate specified, or mismatched data codes. The internal data code of the computer is ASCII. Use conversion tables if other codes are used. May also be caused by incorrect stop bits or bits per character specified. |
| Input data has underlines when printed or displayed. | Underlining is the default method used to indicate parity or framing errors. |
| All input characters are underlined. | Incorrect parity specified. |
| Approximately half of input is underlined. | Check for correct bits per character and stop bits specified. May also be caused when parity is specified for the interface and is not used by the remote. |
| Overwritten data on slow printers. | Increase the number of EOL "Null" characters that are output. This allows the print mechanism time to execute a CR/LF sequence. |
| Remote loses data. | Remote may require handshakes to prevent buffer overrun. |
| Single character OK, but multiple characters in succession produce garbage. | Incorrect stop bits or bits per character specified. |

Appendix E

# ASCII Character Sets

## HP Series 80 and ASCII Character Sets

| EQUIVALENT FORMS | | | | EQUIVALENT FORMS | | | | EQUIVALENT FORMS | | | | EQUIVALENT FORMS | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Series 80 Key | Binary | ASCII | Dec | Series 80 Key | Binary | ASCII | Dec | Series 80 Key | Binary | ASCII | Dec | Series 80 Key | Binary | ASCII | Dec |
| @ᶜ | 00000000 | NULL | 0 | SPACE | 00100000 | space | 32 | | 01000000 | @ | 64 | KEY LABEL$ | 01100000 | ` | 96 |
| Aᶜ | 00000001 | SOH | 1 | ! | 00100001 | ! | 33 | | 01000001 | A | 65 | | 01100001 | a | 97 |
| Bᶜ | 00000010 | STX | 2 | " | 00100010 | " | 34 | | 01000010 | B | 66 | | 01100010 | b | 98 |
| Cᶜ | 00000011 | ETX | 3 | # | 00100011 | # | 35 | | 01000011 | C | 67 | | 01100011 | c | 99 |
| Dᶜ | 00000100 | EOT | 4 | ‡ | 00100100 | $ | 36 | | 01000100 | D | 68 | | 01100100 | d | 100 |
| Eᶜ | 00000101 | ENQ | 5 | ⁚ | 00100101 | % | 37 | | 01000101 | E | 69 | | 01100101 | e | 101 |
| Fᶜ | 00000110 | ACK | 6 | & | 00100110 | & | 38 | | 01000110 | F | 70 | | 01100110 | f | 102 |
| Gᶜ | 00000111 | BEL | 7 | ' | 00100111 | ' | 39 | | 01000111 | G | 71 | | 01100111 | g | 103 |
| Hᶜ | 00001000 | BS | 8 | ( | 00101000 | ( | 40 | | 01001000 | H | 72 | | 01101000 | h | 104 |
| Iᶜ | 00001001 | HT | 9 | ) | 00101001 | ) | 41 | | 01001001 | I | 73 | | 01101001 | i | 105 |
| Jᶜ | 00001010 | LF | 10 | ‡ | 00101010 | * | 42 | | 01001010 | J | 74 | | 01101010 | j | 106 |
| Kᶜ | 00001011 | VT | 11 | + | 00101011 | + | 43 | | 01001011 | K | 75 | | 01101011 | k | 107 |
| Lᶜ | 00001100 | FF | 12 | , | 00101100 | , | 44 | | 01001100 | L | 76 | | 01101100 | l | 108 |
| Mᶜ | 00001101 | CR | 13 | − | 00101101 | - | 45 | | 01001101 | M | 77 | | 01101101 | m | 109 |
| Nᶜ | 00001110 | SO | 14 | | 00101110 | . | 46 | | 01001110 | N | 78 | | 01101110 | n | 110 |
| Oᶜ | 00001111 | SI | 15 | | 00101111 | / | 47 | | 01001111 | O | 79 | | 01101111 | o | 111 |
| Pᶜ | 00010000 | DLE | 16 | 0 | 00110000 | 0 | 48 | | 01010000 | P | 80 | | 01110000 | p | 112 |
| Qᶜ | 00010001 | DC1 | 17 | 1 | 00110001 | 1 | 49 | | 01010001 | Q | 81 | | 01110001 | q | 113 |
| Rᶜ | 00010010 | DC2 | 18 | 2 | 00110010 | 2 | 50 | | 01010010 | R | 82 | | 01110010 | r | 114 |
| Sᶜ | 00010011 | DC3 | 19 | 3 | 00110011 | 3 | 51 | | 01010011 | S | 83 | | 01110011 | s | 115 |
| Tᶜ | 00010100 | DC4 | 20 | 4 | 00110100 | 4 | 52 | | 01010100 | T | 84 | | 01110100 | t | 116 |
| Uᶜ | 00010101 | NAK | 21 | 5 | 00110101 | 5 | 53 | | 01010101 | U | 85 | | 01110101 | u | 117 |
| Vᶜ | 00010110 | SYNC | 22 | 6 | 00110110 | 6 | 54 | | 01010110 | V | 86 | | 01110110 | v | 118 |
| Ẇᶜ | 00010111 | ETB | 23 | 7 | 00110111 | 7 | 55 | | 01010111 | W | 87 | | 01110111 | w | 119 |
| Xᶜ | 00011000 | CAN | 24 | 8 | 00111000 | 8 | 56 | | 01011000 | X | 88 | | 01111000 | x | 120 |
| Yᶜ | 00011001 | EM | 25 | 9 | 00111001 | 9 | 57 | | 01011001 | Y | 89 | | 01111001 | y | 121 |
| Zᶜ | 00011010 | SUB | 26 | | 00111010 | : | 58 | | 01011010 | Z | 90 | | 01111010 | z | 122 |
| [ᶜ | 00011011 | ESC | 27 | ; | 00111011 | ; | 59 | | 01011011 | [ | 91 | /$ | 01111011 | { | 123 |
| \ᶜ | 00011100 | FS | 28 | < | 00111100 | < | 60 | | 01011100 | / | 92 | | 01111100 | \| | 124 |
| ]ᶜ | 00011101 | GS | 29 | = | 00111101 | = | 61 | | 01011101 | ] | 93 | −$ | 01111101 | } | 125 |
| ^ᶜ | 00011110 | RS | 30 | > | 00111110 | > | 62 | | 01011110 | < | 94 | *$ | 01111110 | ~ | 126 |
| _ᶜ | 00011111 | US | 31 | ? | 00111111 | ? | 63 | | 01011111 | — | 95 | +$ | 01111111 | DEL | 127 |


**Personal Computer Division**
**1010 N.E. Circle Blvd., Corvallis, OR 97330 U.S.A.**

Reorder Number
82939-90012

Printed in U.S.A. 3/82
82939-90016