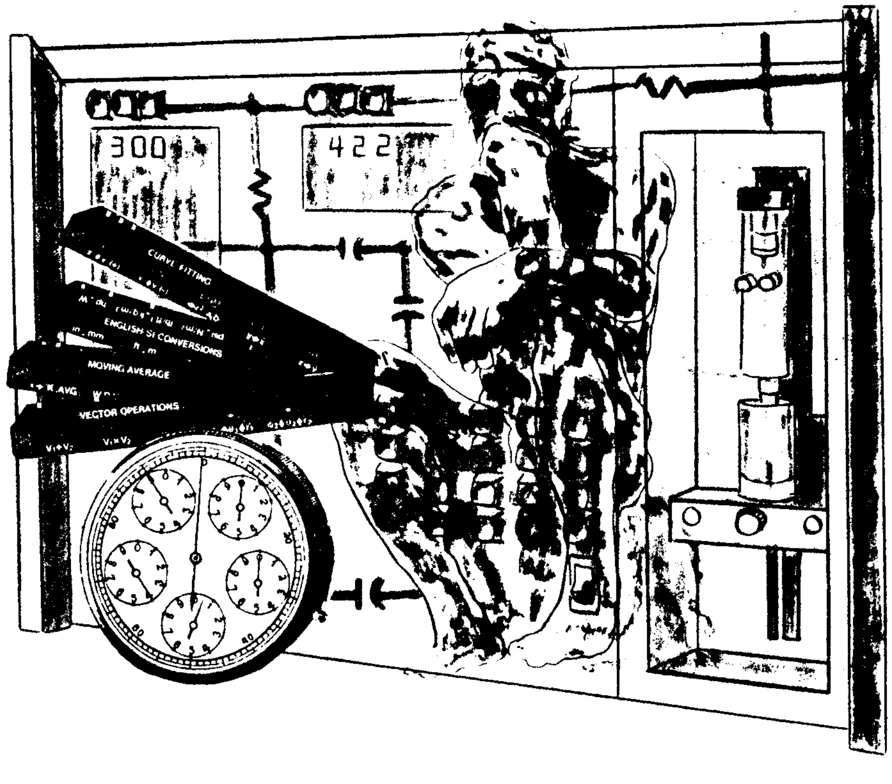
HEWLETT-PACKARD

HP-97S

Installation and Operation Guide



HEWLETT PACKARD

Part Number 5955-2816

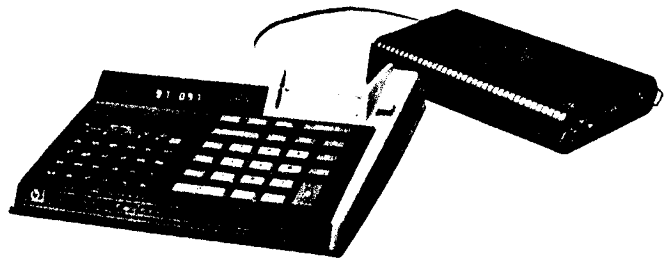
Printed in U.S.A.
December 1, 1977

HEWLETT hp PACKARD

The HP-97S I/O Calculator

Installation and Operation Guide

December 1, 1977



5955-2816

Contents

General Information

The HP-97S I/O Calculator represents a significant step forward in low-priced calculator performance and utility. The HP-97S provides all the power of the HP-97, including printer, magnetic card program storage, 224 program steps, and 26 data registers. In addition, the HP-97S will interface to a peripheral device via five output lines and a 10-digit BCD data port. The data port will accept integer, fixed-point, and floating-point data as well as certain selected instructions. Input control lines allow the peripheral to interact with the calculator during data acquisition, providing a variety of interface configurations. Data inputs are compatible with many logic families including TTL, DTL, CMOS, NMOS, and high threshold logic. The output lines will drive TTL, DTL, CMOS and NMOS. Protection against overvoltage and static discharge is provided on all inputs. To make the versatility picture complete, the HP-97S can be operated from 110 VAC, 220 VAC, or a self-contained, rechargeable battery pack.

> NOTE: Before using this manual, be sure you are familiar with the HP-97 Calculator and its programming language as described in the HP-97 Owner's Handbook and Programming Guide (HP Part No. 00097-90001).

Most of the operating and programming information for the HP-97S can be found in the HP-97 Owner's Handbook. The HP-97S Installation and Operation Guide explains the features which are not covered in the HP-97 Owner's Handbook and provides a more detailed view of the techniques which are most useful to the task of interfacing.

Part One Installation

Introduction

Part One of this manual provides the technical description of the interfacing lines available on the HP-97S. Basic functions of the inputs and outputs are presented along with detailed electrical parameters. The functional information will help you understand the basic actions of the interface, while the electrical information will provide specific checks on the compatibility of the HP-97S with your system. In most cases, intermediate buffering will not be necessary when installing the HP-97S, but the detailed drive and loading specifications can be used to identify applications requiring additional buffers.

Specific application examples and programming techniques for interface control are given in Part Two.

Section 1

Technical Specifications

Temperature Range

Operating or Charging	+10 to +40°C
Storage	−40 to +50°C

Power Requirements

AC Operation

Power Consumption	7 watts maximum
Frequency	50-400Hz
Supply Voltage	
with adapters 82059A and 82069A	90-125 VAC
with adapter 82066A, 82067A, and 82068A	200-250 VAC

Battery Operation

Continuous operation from a fully charged battery pack	3-6 hours
Typical recharge time of a fully discharged battery back	
with calculator OFF	6 hours
with calculator ON	17 hours

Shorter charging periods will reduce the operating time you can expect from a single battery charge. Whether the calculator is on or off, the HP-97S battery pack is never in danger of being overcharged.

DATA Input Lines

Data can be input to the HP-97S by using any or all of 40 DATA input lines. These lines are divided into ten 4-bit digits. Each digit can input any of the following numbers or instructions using binary code.

DATA Input Codes

Input Code (Binary)	Meaning	
0000	[0]	
0001	[1]	
0010	[2]	
0011	[3]	
0100	[4]	
0101	[5]	
0110	[6]	
0111	[7]	
1000	[8]	
1001	[9]	
1010	[•]	(decimal point)
1011	EEX	(enter exponent)
1100	ENTER↑	
1101	A	(continue at LABEL A)
1110	CHS	(change sign)
1111	NO-OP	(no operation)

The logical polarity of the DATA inputs is optional. Either positive-true or negative-true logic can be selected. An internal connection is provided on all DATA inputs so that unused digits automatically input a NO-OP instruction. Therefore, unused DATA inputs can be left unconnected.

The DATA inputs are directly compatible with TTL, DTL, 5-volt CMOS, 12-volt CMOS, HiNIL, HTL, NMOS or PMOS if V_{cc} = 5 volts, and reed relay logic. DATA input voltage during normal operation should be kept within the limits of −12 volts to +24 volts. However, overvoltage protection for the DATA inputs is ±50 volts and all inputs have full static discharge protection.

The following tables show the electrical requirements for the DATA inputs in both True and Complement mode. In these and all other electrical tables, conventional current flowing into the interface is shown as positive and conventional current flowing out of the interface is shown as negative.

Input Drive Requirements with T/C Line High (True Data)

Parameter		Test Condition	DATA Inputs	Units
V_{IH}	High Level Input Voltage (min)		2.8	Volts
V_{IL}	Low Level Input Voltage (max)		1.4	Volts
I_{IH}	High Level Input Current (max)	V_{IN} = 5.0V	25	μa
		V_{IN} = 12.0V	300	μa
I_{IL}	Low Level Input Current (max)	V_{IN} = 0.0V	−300	μa
I_{I}	Input Current at Maximum Input Voltage (max)	V_{IN} = 24V	800	μa
		V_{IN} = −12V	−800	μa

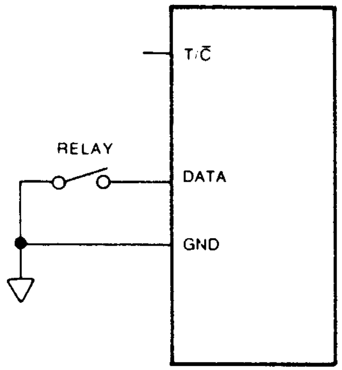
Input Drive Requirements with T/C Line Low (Complemented Data)

Parameter		Test Condition	DATA Inputs	Units
V_{IH}	High Level Input Voltage (min)		2.8	Volts
V_{IL}	Low Level Input Voltage (max)		1.4	Volts
I_{IH}	High Level Input Current (max)	V_{IN} = 5.0V	200	μa
		V_{IN} = 12.0V	500	μa
I_{IL}	Low Level Input Current (max)	V_{IN} = 0.0V	Near 0	μa
I_{I}	Input Current at Maximum Input Voltage (max)	V_{IN} = 24V	1000	μa
		V_{IN} = −12V	−500	μa

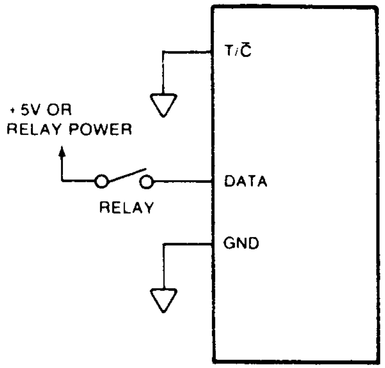
From these specifications you can see that the logic high threshold is a little above guaranteed TTL levels. However, the HP-97S interface will typically use only half of the high level drive of a low power gate and significantly smaller percentages for other TTL families. Additionally, pull-up resistors are provided in the True Data mode which will raise the high output voltage to almost five volts for a lightly loaded device. Compatibility with TTL families is assured if the devices driving the interface are lightly loaded. You should avoid connecting to heavily loaded TTL if the interface is in the Complemented Data mode. This configuration could provide poor noise immunity in the logic high state.

The DATA inputs can be driven by open-collector logic if the interface is in the True Data mode.

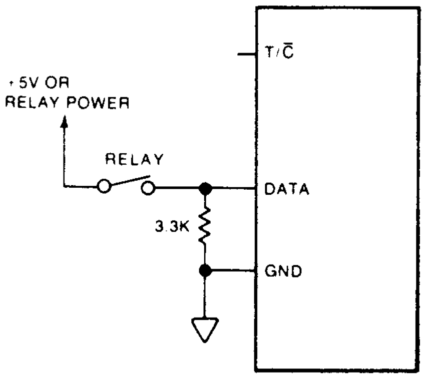
Relay logic can be interfaced to the DATA inputs in four ways. These are shown below. If the relay power supply voltage is connected to an input, it should not be greater than +24 volts.



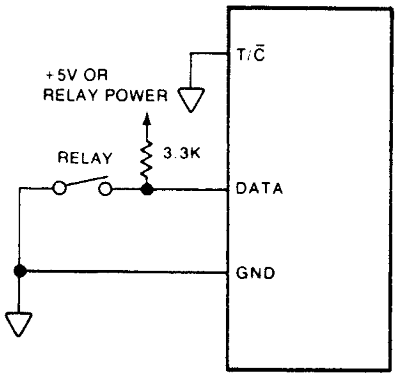
CONTACT OPEN = LOGIC "1"



CONTACT OPEN = LOGIC "1"



CONTACT OPEN = LOGIC "0"



CONTACT OPEN = LOGIC "0"

Control Lines

Control Outputs

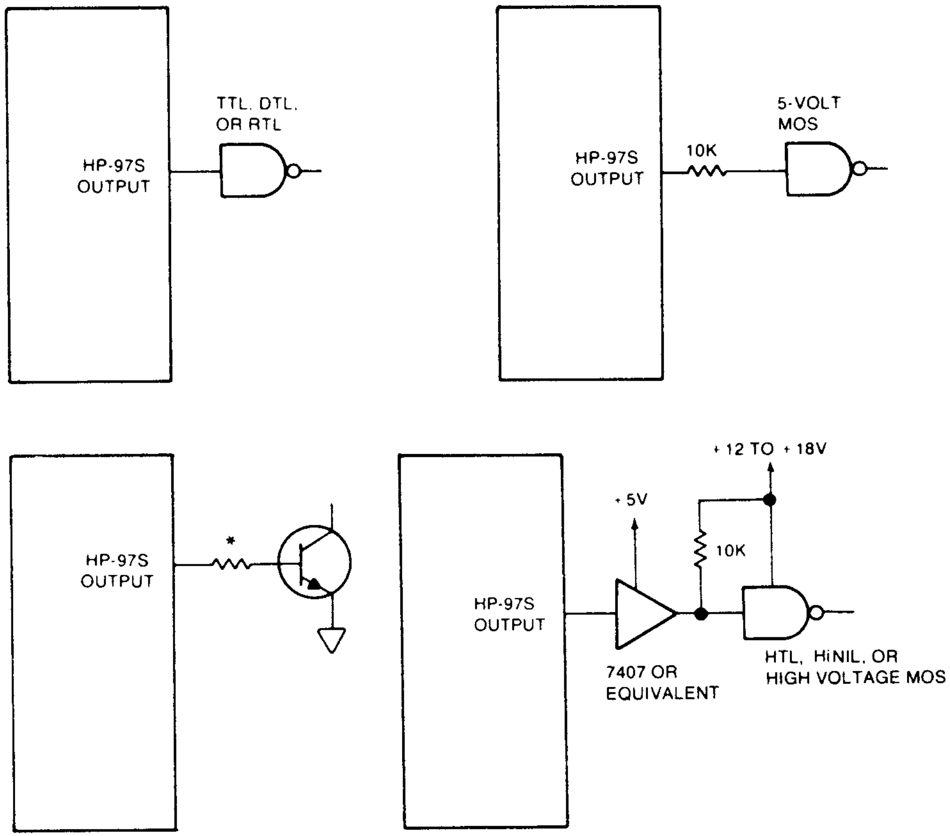
There are five control outputs provided by the HP-97S. Four of these are the flags controlled by the calculator. These are labeled $\overline{F\emptyset}$, $\overline{F1}$, $\overline{F2}$, and $\overline{F3}$. The other output is generated by the interface to control the loading of data through the interface. This line is labeled $\overline{LE}$. All of the outputs are active-low logic and have the following drive capability.

Drive Capability of Control Output Lines

Parameter	Test Condition	Typical @25°C	Minimum @40°C	UNITS
I_{OH} High Level Output Current	V_{out} = 0.6V	−6.0		ma
	V_{out} = 2.5V	−2.5	−1.0	ma
I_{OL} Low Level Output Current	V_{out} = 0.4V	6.0	2.0	ma

The outputs are compatible with TTL, DTL, 5-volt CMOS, low threshold NMOS, RTL and discrete transistors. The typical high level output voltage will be 5.0 to 5.5 volts into a light load. Some type of high voltage buffer, such as a 7407, is necessary if the outputs need to drive 12-volt or 15-volt logic. The drawings on the next page show some typical output interface connections.

Typical Output Interfaces



*Resistor may not be needed. With no resistor, drive current to transistor base will be 5 to 10 ma. A resistor is used of this amount of drive is excessive.

The flag outputs duplicate the actions of the four software flags described in Section 13 of the HP-97 Owner's Handbook. When a flag is set, the corresponding flag output will go to the active state (logic low). When a flag is cleared, the corresponding flag output will go to the inactive state (logic high). The following table is a summary of the flag action in the HP-97S. For further information, see "Output Control" in Section 3 of this manual.

Action of Flag Outputs

Flag #	Cleared By	Set By
0	Power-on Interface Inhibit (2) CLF 0	STF 0
1	Power-on Interface Inhibit (2) CLF 1	STF 1
2	Power-on Interface Inhibit (2) CLF 2 F?2	STF 2
3	Power-on Interface Inhibit (2) CLF 3 F?3	STF 3 Data Entry (1)

NOTES: 1. Data entry may originate from the keyboard or the interface. The data functions are the numbers 0-9, EEX, and . (decimal point). Data entry from the card reader is not detected by the interface.

2. Interface inhibit clears the flag outputs, but not the software flags in the calculator memory. All four flag outputs will be updated to match the software flags by the first flag-affecting instruction received after the interface is enabled.

The $\overline{LE}$ line is used to enable the input circuitry of the interface and signal the peripheral device that the calculator is ready to accept data. A logic low on the $\overline{LE}$ line indicates that the interface input circuitry is enabled. A logic high on the $\overline{LE}$ line indicates that the interface will not input data at that time. The conditions controlling the $\overline{LE}$ line are summarized by the next five statements.

1. $\overline{LE}$ will be high (disabled) while a program is executing.
2. $\overline{LE}$ will be high (disabled) while the interface contains a previously loaded item of data.
3. $\overline{LE}$ will be high (disabled) if flag 3 is set and a "continue at A" instruction is entered through the interface.
4. $\overline{LE}$ will be low (enabled) whenever none of the first three conditions exist. Basically, to enable $\overline{LE}$ the input register of the interface must be empty, flag 3

must be cleared, and program execution halted. More details will be covered in other sections.

5. $\overline{LE}$ will be low (enabled) immediately after power-on of the calculator.

Control Inputs

The HP-97S has four control input lines: LOAD, $\overline{LOAD}$, T/$\overline{C}$, and INHIBIT. Two of these lines are used to initiate the loading sequence of the interface. A third line is used to select positive-true or negative-true logic for the data, and the last is an interface enable.

All of the control inputs are compatible with TTL, DTL, HiNIL, and reed relay logic. Additionally, the $\overline{LOAD}$ input is compatible with 5-volt CMOS, 12-volt CMOS, and NMOS or PMOS if V_{cc} = 5 volts.

Control input voltage during normal operation should be kept within the limits of −12 volts to +24 volts. However, overvoltage protection for the control inputs is at least 50% greater than this range and all inputs have full static discharge protection. The following table shows the electrical requirements for the control inputs.

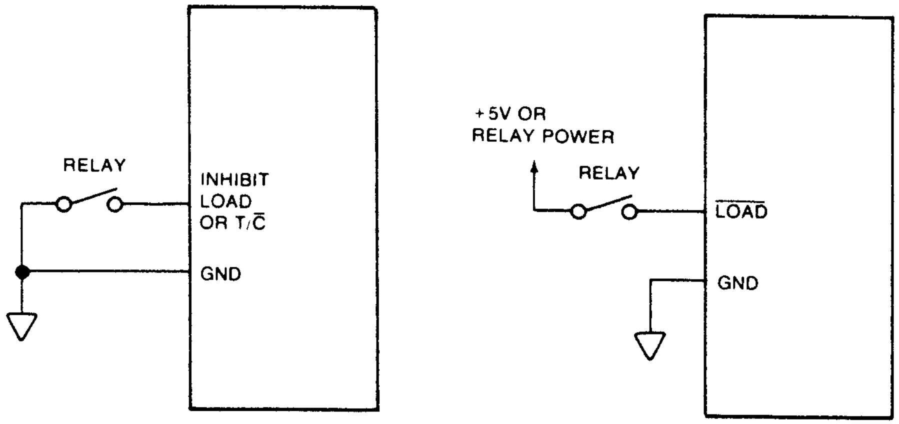
Input Drive Requirements of Control Input Lines

Parameter	Test Condition	INPUT				Units
		$\overline{LOAD}$	LOAD	T/$\overline{C}$	INHIBIT	
V_{IH} High Level Input Voltage (min)		2.8	2.8	3.4	3.4	Volts
I_{IL} Low Level Input Voltage (max)		1.4	1.4	0.8	2.0	Volts
I_{IH} High Level Input Current (max)	V_{IN} = 5.0V	60	150	200	120	μa
	V_{IN} = 12.0V	0.15	2.0	2.5	1.0	ma
I_{IL} Low Level Input Current (max)	V_{IN} = 0.0V	Near 0	−1.5	−1.5	−0.7	ma
I_I Input Current at Maximum Input Voltage (max)	V_{IN} = 24V	0.3	5.0	6.0	2.5	ma
	V_{IN} = −12V	−0.15	−5.0	−5.0	−2.0	ma

Internal pull-up resistors are provided on the T/$\overline{C}$, LOAD, and INHIBIT lines. As explained in the "DATA Input Lines" section, this provides better TTL compatibility by raising the high level voltage. This also guarantees certain states if the lines are left unconnected. Thus, with nothing plugged into the connector, the interface automatically enters the inhibited state. No connection to the T/$\overline{C}$ input establishes the True Data mode. If either $\overline{LOAD}$ or LOAD is unused, internal connections will hold that input in the enabled state. This allows control of the interface using only one control signal.

If the control inputs are driven with open-collector logic, the open collector device should have a voltage rating of at least 7.0 volts. If the control inputs are driven with relay logic, the following connections must be used. If the relay power supply is connected to an input, it should not be greater than +24 volts.

Connecting Relay Logic to the Control Inputs



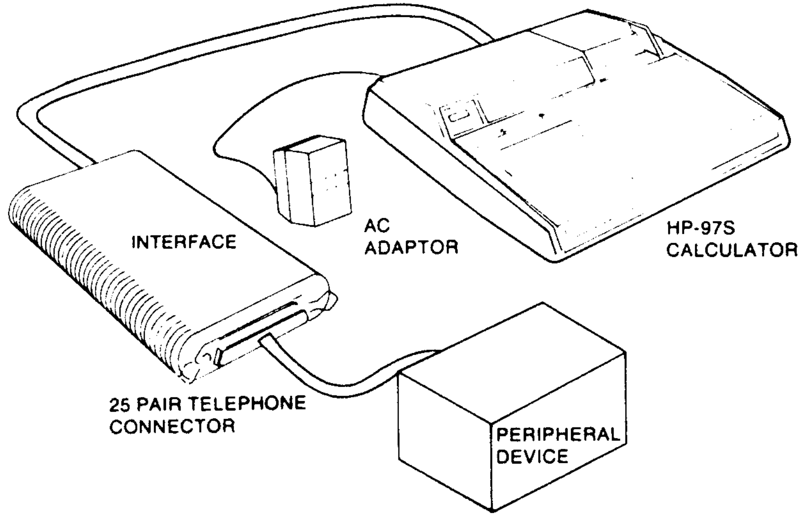
```
649.1523000   ***
        084   RCL5
656.6409000   ***
        085   X⇄Y
649.1523000   ***
        086   -
7.488600000   ***
        087   ÷
4.849371044   ***
        088   STOE
        089   ×
176.1049095   ***
        090   RCL6
94.32000000   ***
        091   X²
8896.262400   ***
        092   RCL9
3.000000000   ***
        093   ÷
2965.420800   ***
```

Section 2

External Connections

System Diagram

The general configuration of an HP-97S system is shown below.

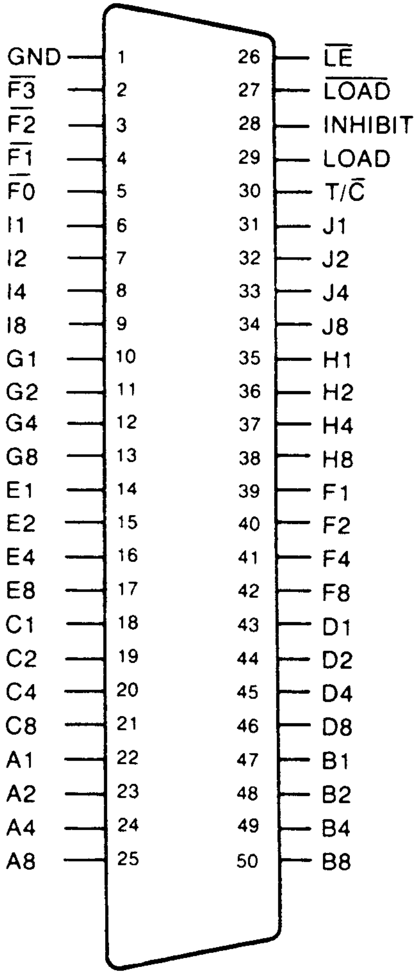


Peripheral Connector Wiring

Your peripheral device is connected to the HP-97S interface with a standard 25-pair telephone connector (Amphenol 57-30500 or equivalent). The diagram on the next page shows the rear view (wire side) of the connector which must be attached to the peripheral device. The ten digit inputs are labeled A through J. A is the most significant digit. The references 1, 2, 4, and 8 indicate the relative value of the four binary lines used for each digit. When wiring the connector, do not forget to connect the signal ground of the peripheral to pin #1. Erratic operation will result if the peripheral and the HP-97S do not have a common ground connection.

Interface Connector Plug

VIEWED FROM WIRE SIDE
OF MALE CONNECTOR



DATA INPUTS ARE A(MSD) TO J(LSD)

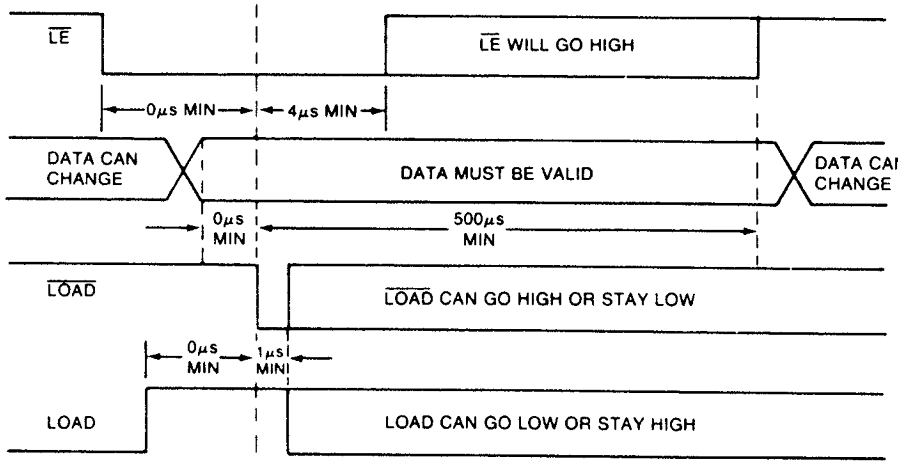
Using the Interface Control Lines

The basic sequence of operation for a data input is listed below.

1. Clear flag 3 and halt program execution with a R/S, RTN, or PAUSE instruction.
2. If the input register is empty, $\overline{LE}$ will go low.
3. Be sure that the DATA lines are stable before activating the LOAD inputs.
4. Apply an active transition to one of the LOAD inputs while the other is in the active state.
5. Be sure that the DATA lines remain stable for at least 500 μs after activating the LOAD inputs.

The following diagram illustrates the timing relationships of the control lines during the loading of data. An active transition of the $\overline{LOAD}$ line is used as an example, but the timing requirements are identical for loading initiated by an active transition of the LOAD line.

Control Line Timing



Summary of Loading Control Lines

$\overline{LE}$	$\overline{LOAD}$	LOAD	Loading Action
H	X	X	None
L	X	L	None
L	H	X	None
L	↓	H	Load
L	L	↑	Load

H = Logic high level
L = Logic low level
X = Level doesn't matter
↑ = Active transition from low to high ($t_r < 2\mu s$)
↓ = Active transition from high to low ($t_f < 2\mu s$)

The timing diagram does not show the action of the $\overline{LE}$ output beyond a very short time period. The state of the $\overline{LE}$ line is dependent upon the input provided through the interface. The shortest time that $\overline{LE}$ will stay high is about 6ms. if only NO-OP instructions are input. Loading one digit takes about 120ms. and loading ten digits takes about 1.2 seconds. If no "continue at A" instruction is included, $\overline{LE}$ will go low again after the digits are loaded. If the data entry is followed by a "continue at A" instruction, $\overline{LE}$ will stay high until flag 3 is cleared and program execution halted.

The other interface control lines are $T/\overline{C}$ and INHIBIT. The function of these lines is very straightforward. Although they can be driven by external logic, for most applications they will be preset by the use of jumpers.

When the INHIBIT line is high, interface action is stopped. A convenient way to use the INHIBIT line is to install a jumper on the peripheral connector which pulls INHIBIT to ground. Thus, whenever the peripheral is attached, the interface is active. Whenever the peripheral is not attached, the internal pull-up resistor will cause the INHIBIT line to go high. This automatically isolates the interface from the calculator, allowing the HP-97S to be used as a stand-alone desktop calculator.

When the $T/\overline{C}$ line is left unconnected, the DATA inputs will be positive true. That is, a logic high will be a "one" and a logic low will be a "zero". With the $T/\overline{C}$ jumpered to ground, the DATA inputs accept inverted data. In this mode, a logic low will be a "one" and a logic high will be a "zero". Both modes, True and Complement, input a NO-OP instruction from a digit with all four bit lines unconnected.

Part Two
Operating and Programming

Introduction

This part describes the interaction between hardware and software in the HP-97S. It is not practical to provide a detailed description of every specific application. However, many typical configurations are discussed. You can use the information presented in these examples to create more specific or more complex systems.

The instruction set and program operation are the same in the HP-97 and the HP-97S. All available instructions and functions are described in the HP-97 Owner's Handbook. However, the interface requires that flag 3 and LABEL A be used in specific manners. Within this framework, any analysis which can be performed using data from the keyboard can be performed using data from the interface.

Basic Applications

Power-On Initialization

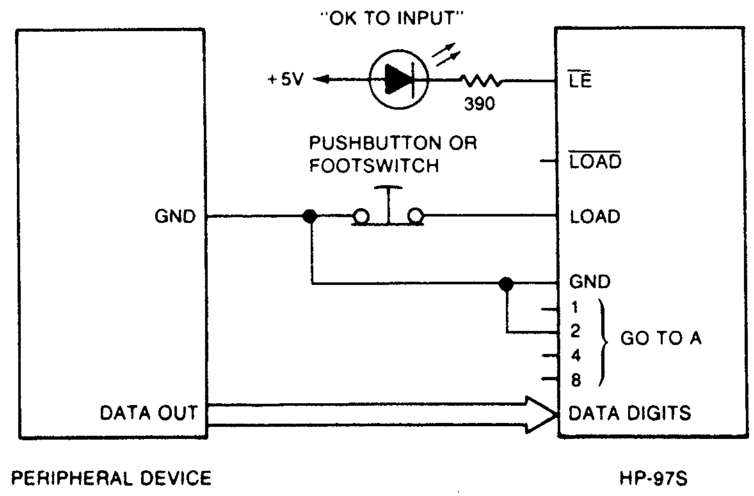
When power is first applied to the HP-97S, $\overline{LE}$ is low and the interface is ready to receive data. If the LOAD inputs are activated by the peripheral before a program is loaded, an error will be generated when the "continue at A" instruction is input. There is no LABEL A if there is no program.

If this happens, press any key to clear the error and load the desired program. To prevent this error, load and initialize the program before turning on the peripheral.

Manual Input

The manual mode is an alternative to machine controlled interface configurations. It may be the simplest approach when the HP-97S is incorporated into a work station which already requires an operator or attendant. Manual input control might be necessary if the peripheral device does not have the signals necessary to control a data logging device. The following diagram shows the connections used to input data manually. The "continue at A" code can be included by grounding the 2's place of any digit following the data. Up to nine data digits can be used and all unused DATA lines are left unconnected. An LED indicator can be used if desired.

Interface Connections



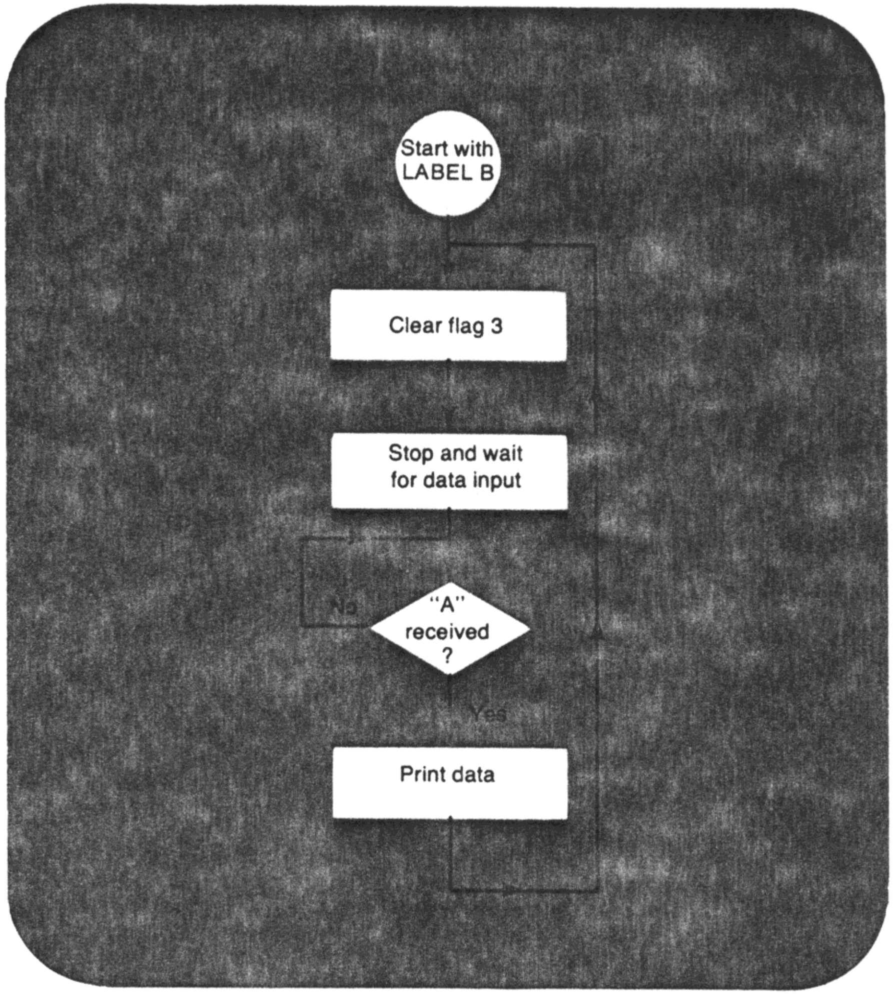
Manual input can be illustrated using a work station containing a digital scale and an HP-97S which is being used to log all of the package weights during a shift. The program loaded into the HP-97S at the beginning of the day is shown below.

```
001  *LBLB       21 12
002     CF3   16 22 03
003     R/S         51
004  *LBLA       21 11
005    PRTX        -14
006    GTOB      22 12
007     R/S         51
```

The data logging sequence is initiated by pressing the B key. This causes program execution to begin at LABEL B. The interface input register will be empty at this point, which is one of the conditions necessary to enable data loading. The CLF 3 is a second condition. The last condition is that program execution be halted. The R/S instruction is used to halt execution.

Next, the object to be weighed is placed on the scale. When the scale reading has stabilized, the attendant presses the pushbutton or footswitch. The data from the scale is loaded through the interface, followed by a "continue at A" instruction. This enters the data into the X-register and directs the program to the instructions at LABEL A.

At LABEL A, the PRINT X instruction logs the data. The program then goes back to LABEL B, executes the CLF 3, and stops to wait for the next input. If desired, an LED can be placed on the $\overline{LE}$ output to give the attendant a visual indication when the interface is enabled and ready to accept data. The program sequence is illustrated by the flowchart on the following page.



Programming Techniques

Processing the Data

Most of the examples in this manual will use a single PRINT X instruction as the only operation performed on the incoming data. This is to emphasize the I/O controlling software and to make the examples easier to understand. In actual applications, this simple data logging step would probably be augmented with many calculations. The next example will assume the same hardware and order of operations as the example shown in the "Manual Input" section. However, the program has been modified to record the weights in groups of ten, give the mean and standard deviation of each group, and format the resulting printout. The program listing is shown below.

```
001  *LBLC      21 13
002   CLRG      16-53
003    P⇄S      16-51
004    FIX        -11
005   DSP0     -63 00
006  *LBLB      21 12
007    CF3   16 22 03
008    R/S         51
009  *LBLA      21 11
010   PRTX        -14
011     Σ+         56
012   ENT↑        -21
013      1         01
014      0         00
015   X≠Y?      16-32
016   GTOB      22 12
017    SPC      16-11
018   DSP2     -63 02
019      x̄      16 53
020   PRTX        -14
021      S      16 54
022   PRTX        -14
023    SPC      16-11
024    SPC      16-11
025   GTOC      22 13
026    R/S         51
```


When the [C] key is pressed, the program begins at LABEL C. Initialization consists of clearing the statistics registers and selecting the format for printing the data. The steps at LABEL B prepare the system for data input from the interface. After the data is input with a "continue at A" instruction, execution resumes at LABEL A. First, the data is printed. Then the Σ+ calculations are performed, leaving the item count in the X-register. The ENTER instruction moves this value to Y so it can be compared to the maximum count of 10 which is placed in X. If the count has not yet reached 10, the program goes back to LABEL B to get the next weight input.

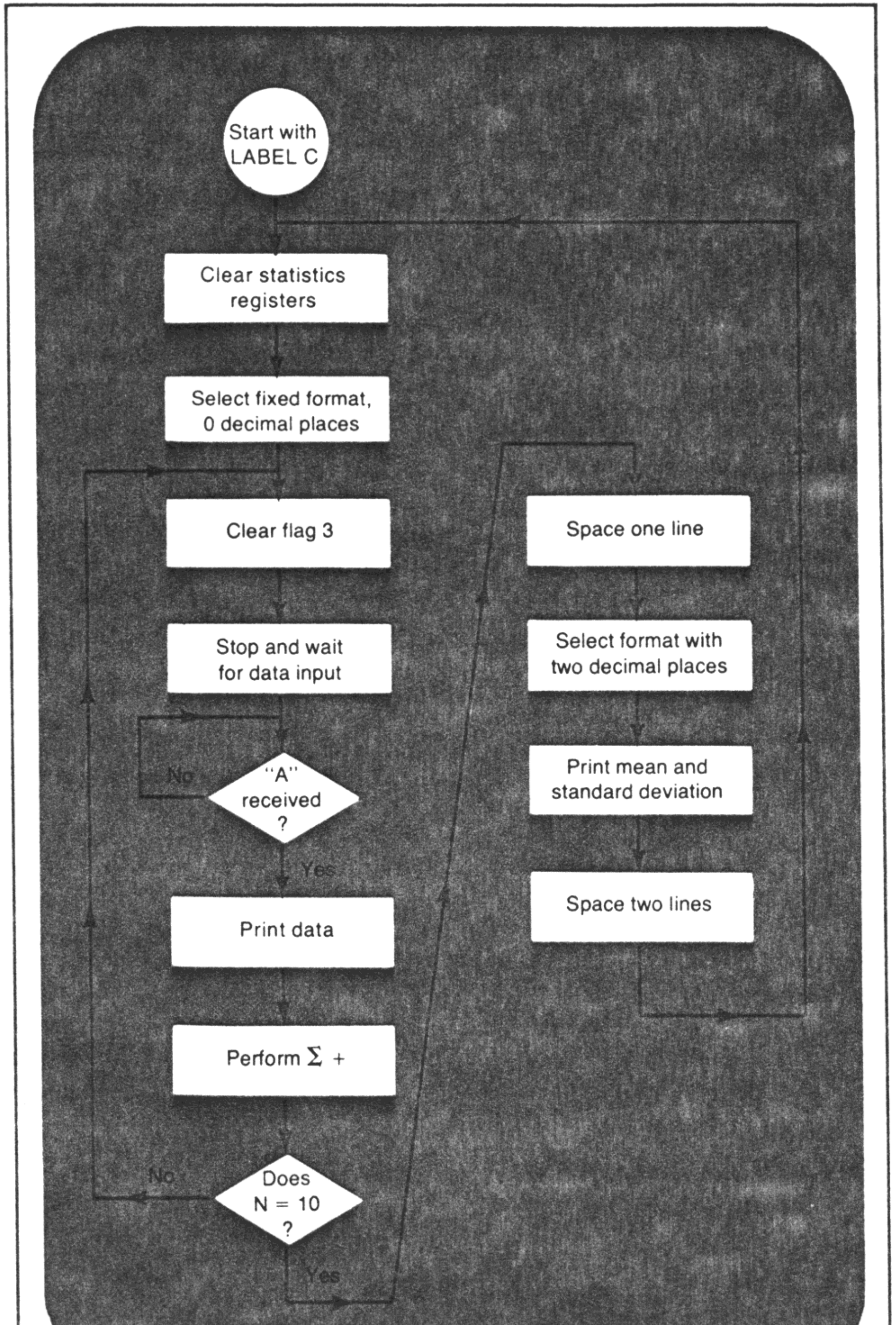
When ten weights have been input, the printer adds a space for clarity and changes formats for more resolution on the statistical printout. Then the mean and standard deviation are computed and printed. Two more spaces are added to separate groups and the program goes back to LABEL C to process the next group of weighings.

A typical printout looks like this.

```
  101.  ***
  100.  ***
  102.  ***
   99.  ***
  105.  ***
  102.  ***
  104.  ***
   98.  ***
  102.  ***
  101.  ***

101.40  ***
  2.12  ***
```

The sequence of this program is illustrated by the flowchart on the next page. This is only a sample of how a data input can be combined with calculations. The HP-97S provides a broad range of operations, including scaling, formatting, linearizing, curve fitting, comparing, unit conversions, statistical and geometric analysis, and many mathematical functions. The HP-97 Owner's Handbook explains the functions which are available and provides a base of information from which you can build your own programs.



Use of LABEL A

The only instruction provided to cause the HP-97S to resume program execution after an interface input is "continue at A" (binary code 1101). If this instruction is executed with the printer in TRACE mode, the listing will show "GSBA". However, this instruction does not store a pending return address nor does it require a RTN statement to end the routine. Therefore, this manual refers to the instruction as "continue at A", "go to A", and "A", since the "GSB" term is somewhat misleading.

Whatever you choose to call it, the code 1101 at the interface produces the same calculator action as the pressing of key **A**. This action is explained in "Running a Program", Section 6 of the HP-97 Owner's Handbook. The "A" or "continue at A" instruction will cause the program to resume execution at the **next** LABEL A it finds. If there is no LABEL A between the current line and the end of the program, the calculator will wrap around to line 001 and resume the search. A program can contain any number of LABEL A lines.

The following example program shows the use of two LABEL A lines. This program accepts seven pairs of readings, such as X and Y coordinates. In order to use the convenience of the Σ+ function, the first data input must cause a different program action than the second. This would be cumbersome with only one LABEL A. A description of the program follows the printout.

```
001  *LBLC        21 13
002   CLRG        16-53
003    P⇄S        16-51
004      7           07
005   STOI        35 46
006  *LBLB        21 12
007    CF3     16 22 03
008    R/S           51
009  *LBLA        21 11
010    CF3     16 22 03
011    R/S           51
012  *LBLA        21 11
013     Σ+           56
014   DSZI     16 25 46
015   GTOB        22 12
016      x̄        16 53
017    X⇄Y          -41
018   PRTX          -14
019     R↓          -31
```

The program starts by clearing the statistics registers and storing seven in the I-register. The I-register will be used to count the data pairs. At LABEL B, flag 3 is cleared and execution halted so that the Y coordinate can be input. When the "continue at A" instruction is received, program execution resumes at the first LABEL A (line 009). Here another input operation is performed which places the X coordinate in the X-register and moves the Y coordinate to the Y-register. When the "continue at A" instruction from this input is received, program execution resumes at the LABEL A on line 012. The $\Sigma+$ function is performed and I is decremented. If the number in I has reached 0, the means are calculated and printed. If the number in I was not yet 0, the program will go back to LABEL B to input another pair of coordinates.

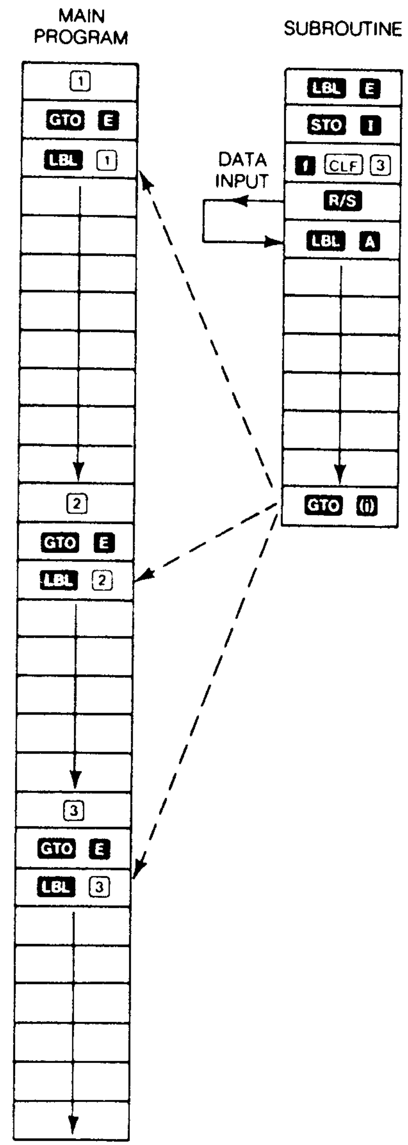
Data Input in Subroutines

As explained in the previous section, the "continue at A" instruction which comes through the interface produces the same action as pressing key **A** . This causes the calculator to lose all pending subroutine returns. (See "Subroutine Limits", Section 10 of the HP-97 Owner's Handbook). Therefore, data cannot be input through the interface while the calculator is executing a conventional subroutine.

It takes only a couple of extra program steps to avoid this problem. If your application makes it desirable to input data from a subroutine, the last subroutine instruction must use an explicit return address. The subroutine is called with a GTO, rather than a GSB, and the return to the main program is accomplished with a GTO (i), rather than a RTN. The extra steps needed are those required to store and label the return address.

A detailed explanation of the GTO (i) instruction can be found in "Indirect Control of Branches and Subroutines", Section 12 of the HP-97 Owner's Handbook. The following diagram shows an example of a data input subroutine. The program block at LABEL E represents a subroutine which inputs data and formats or adjusts that input for use in the main program. This subroutine is called three times by the main program with the GTO E instructions. Immediately before each GTO E, the X-register is loaded with the return address for that particular subroutine call. This return address is stored in the I-register and used by the GTO (i) instruction to resume program execution at the desired label when the subroutine is finished.

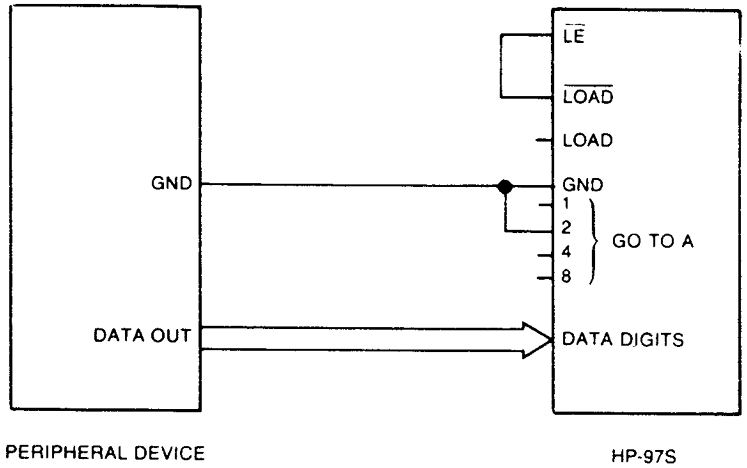
Simulating Subroutine Calls



Continuous Sampling

This mode can be used when the data from the peripheral is always valid and the data acquisition is to be software controlled. This can be illustrated by a data logging task which requires a reading approximately every ten seconds. The PAUSE instruction can be used to keep time if accuracy is not important. With $\overline{LE}$ connected to $\overline{LOAD}$, the interface will input data everytime it is enabled. A diagram and sample program listing follow. Longer time intervals are best generated by placing the PAUSE in a loop.

Interface Connections



Program Listing

```
001  *LBLA      21 11
002   PRTX        -14
003    PSE      16 51
004    PSE      16 51
005    PSE      16 51
006    PSE      16 51
007    PSE      16 51
008    PSE      16 51
009    PSE      16 51
010    PSE      16 51
011    CF3   16 22 03
012    R/S         51
```


Strobed Input

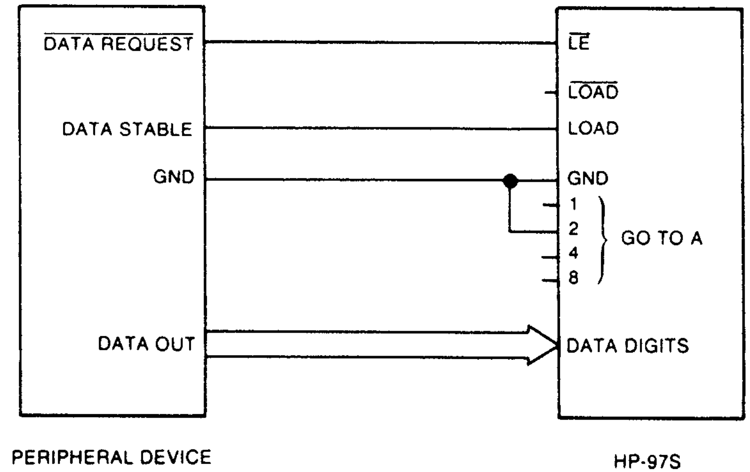
A strobed data input is an easy way to load data from equipment which has a "data stable" signal. A device like a scale will have an unstable output while an object placed on it is coming to rest. The manual input mode deals with that problem by having an attendant wait until the reading looks stable. However, many instruments have an output signal which activates when the data from the instrument has stabilized. When connected to this type of instrument, the HP-97S can request a reading and be signaled when the reading is stable.

There are two basic approaches to controlling a strobed data transfer. In one case, a button on the calculator is pressed to initiate the transfer and in the other case, a button on the peripheral is pressed to initiate the transfer. The method chosen depends upon your preference and whether or not your peripheral is capable of intiating the transfer.

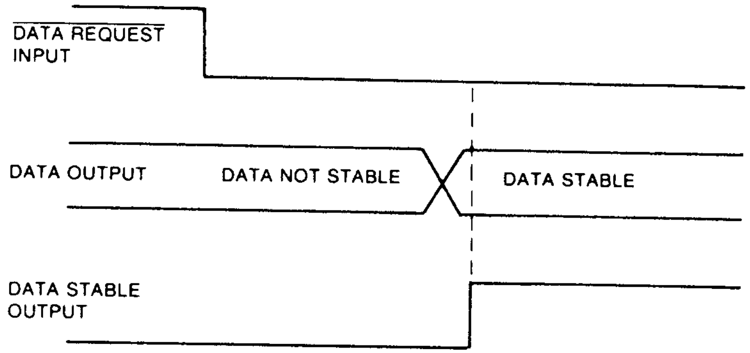
The first example shows the wiring and programming necessary for the HP-97S to initiate the transfer. Using a digital scale as an example, the configuration works as follows:

1. The object to be weighed is placed on the scale.
2. The **R/S** key on the HP-97S is pressed.
3. When the output from the scale is stable, the reading is input and printed.
4. The next object is placed on the scale and the process is repeated.

Interface Connections



Peripheral Timing



Program Listing

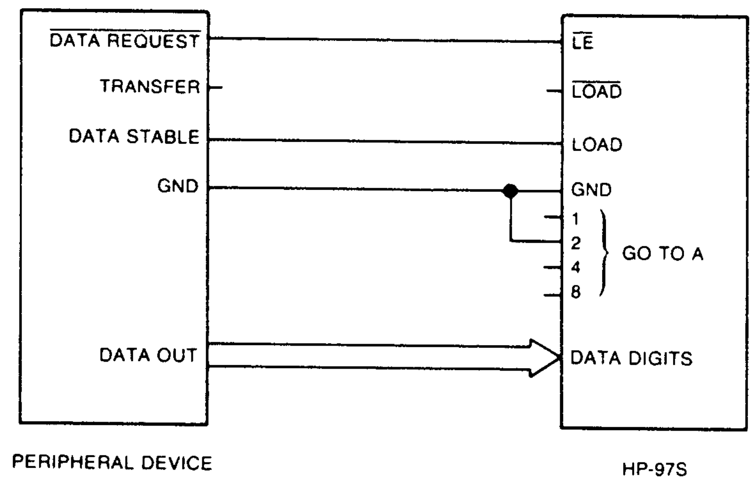
```
001  *LBLB      21 12
002    CF3   16 22 03
003    R/S         51
004  *LBLA      21 11
005   PRTX        -14
006    R/S         51
007   GTOB      22 12
008    R/S         51
```

The second example shows the wiring and programming necessary if the peripheral is capable of initiating the data transfer. This configuration operates the same as the first example, except a "transfer" button on the peripheral is pressed instead of the **R/S** key on the calculator.

Although this system is similar to the manual input mode, it has some advantages:

- No extra switches or indicators are needed.
- The attendant does not need to watch the $\overline{\text{LE}}$ state of the calculator or the stability of the peripheral data.
- Since the data input requires no operator attention, this configuration can be used on automated machinery.

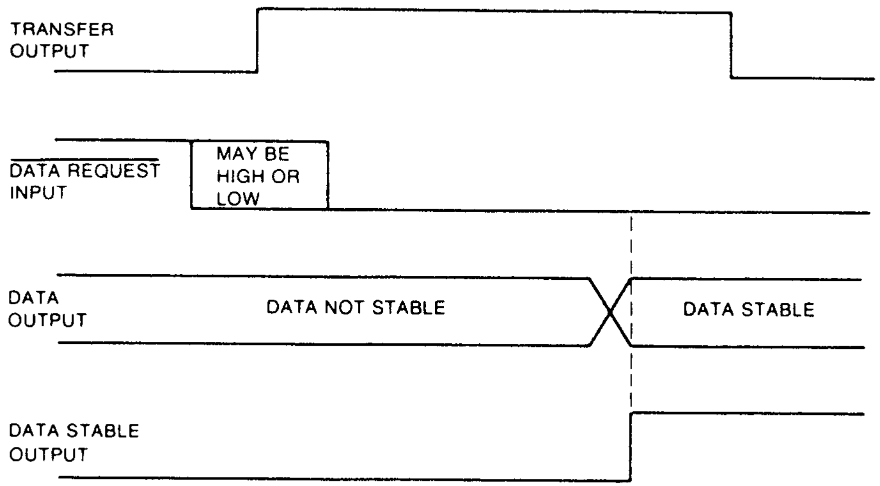
Interface Connections



Program Listing

```
001  *LBLB      21 12
002    CF3   16 22 03
003    R/S         51
004  *LBLA      21 11
005   PRTX        -14
006   GTOB      22 12
007    R/S         51
```

Peripheral Timing



Output Control

One of the important features of the HP-97S is the availability of output signals. Four flag output lines are provided, but there are some limitations on the use of flag 3 because of its role in controlling data transfers through the interface. Flags 0, 1 and 2 can be used freely to enhance the controlling power of the HP-97S. These outputs can be used to drive LED indicators directly, providing more information to operators. Commands such as peripheral range and function selections can be program controlled by using the flag outputs. When connected to suitable drivers, the flags can operate hydraulic and pneumatic solenoids, motor starters, or other electro-mechanical devices.

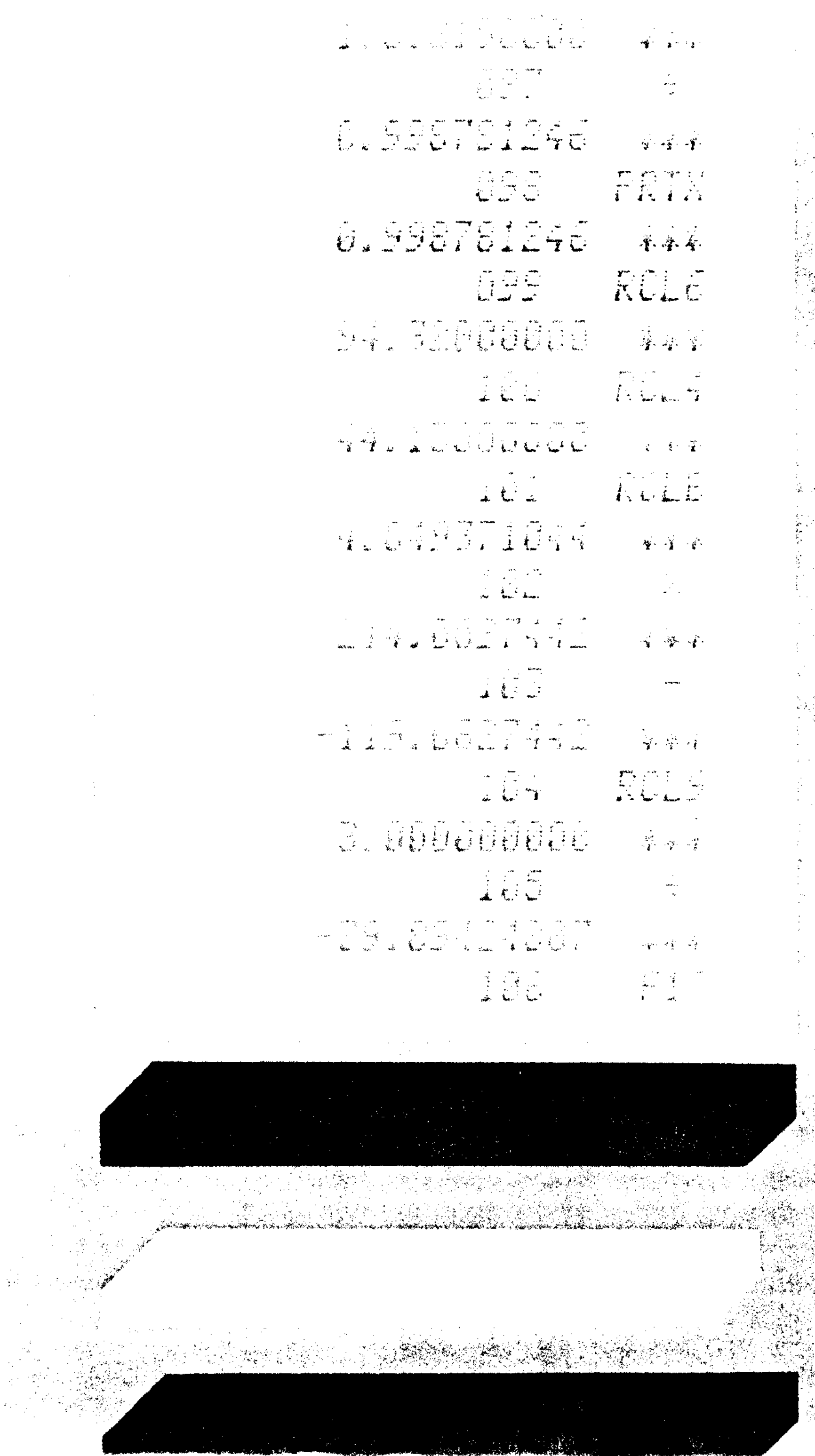
The flag outputs of the interface are controlled by the software flags in the HP-97S memory. However, there are certain situations when the flag outputs may not correspond to the states of the software flags. These are listed below.

1. When a program is loaded from a magnetic card, the software flags are placed in the states indicated by the status information on the program card. At this time, the flag outputs will remain in their previous states.
2. When data is loaded from a magnetic card, the software flag 3 will be set, but the $\overline{F3}$ output will remain unchanged.
3. When f CLPRGM is pressed in PRGM mode, the software flags will be cleared, but the flag outputs will remain unchanged.

4. When the interface has been deactivated by use of the INHIBIT input, the flag outputs will assume an inactive (logic high) state, but the software flags will remain unchanged.

Any discrepancies between the flag outputs and the software flags will be eliminated by a flag-affecting instruction. If any instruction listed below is received when the interface is enabled, the HP-97S will place all flag outputs in the states indicated by the software flags.

1. Any STF or CLF instruction
2. Any F? instruction
3. A data entry from the keyboard or interface.



Extended Applications

The preceding section discussed some of the simpler interfacing tasks. This section outlines some solutions to more complex interfacing problems. The use of the control and DATA lines is discussed in greater detail. There are also hints on how to augment the interfacing power of the HP-97S with a minimum of external circuitry.

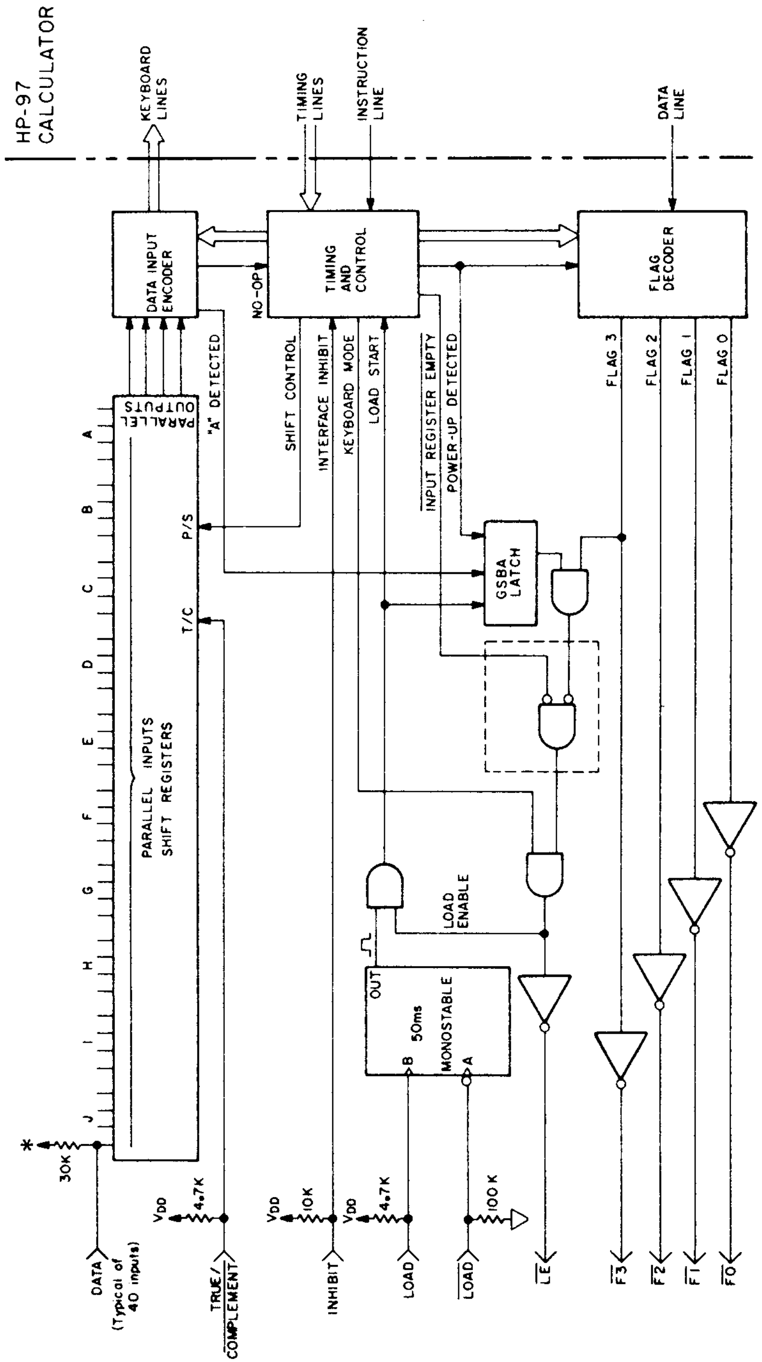
Since it is important to understand the action of all input and output lines when designing a specialized interface, a detailed block diagram is shown on the next page. This is a diagram of the calculator interface module, with special emphasis on the external input and output lines. When used with the electrical parameters presented in Part One, this diagram will provide the detailed information necessary to the custom system designer.

Handshake Interface

The term "handshake" is used to describe a popular form of asynchronous data transfer. The method has these characteristics:

- The sender can make data available at any time, but it must hold the data until the receiver takes it.
- The receiver can request data at any time, but it cannot perform an input operation until the sender makes the data available.
- Communication between the sender and receiver is accomplished on two control lines. These signals are often referred to as "Data Available" and "Data Accepted".
- The Data Available signal is generated by the sender when its data output is stable and valid. The Data Accepted signal is generated by the receiver when the Data Available signal has been received and the data input operation is complete. The Data Accepted signal tells the sender that the last data item can be forgotten and the next data item can be provided.

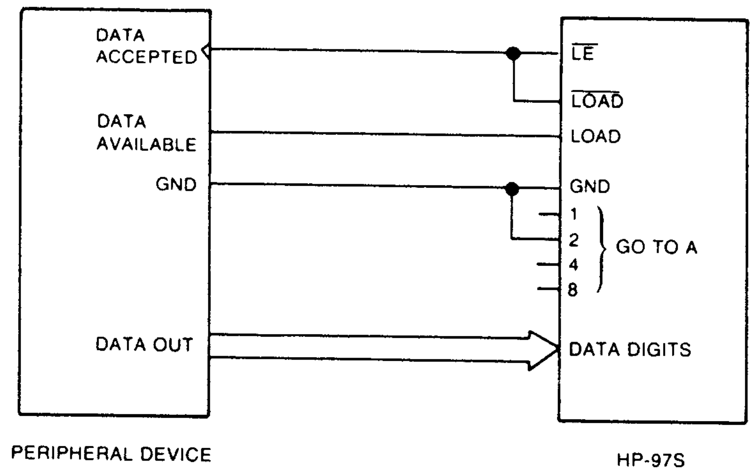
Detailed Block Diagram



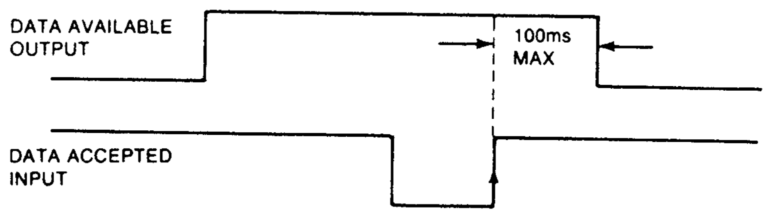
* PULLED TO V_{DD} FOR TRUE DATA; PULLED TO GND FOR COMPLEMENTED DATA

There are two connections which can be used to implement a handshake interface with the HP-97S. The method chosen depends upon the characteristics of the peripheral device. The first configuration is used if the handshake circuitry in the peripheral is only sensitive to an active transition of the Data Accepted line.

Interface Connections

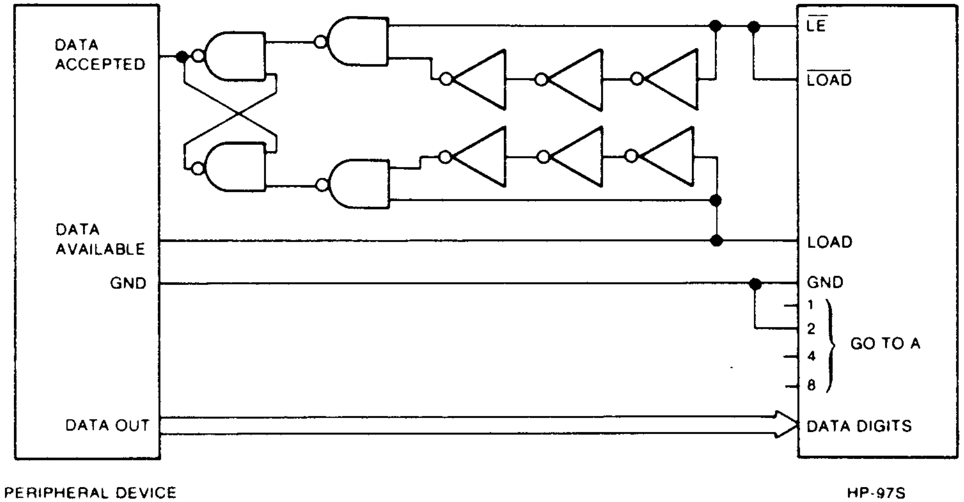


Peripheral Timing

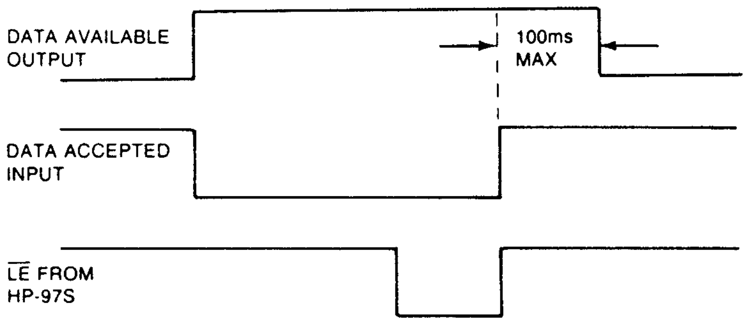


The connection just shown is unacceptable if the peripheral device is sensitive to the **level** of the Data Accepted line instead of an **edge**. For a peripheral of this type, the Data Accepted line must be reset as soon as new data becomes available. Otherwise, the peripheral will think that the data was accepted as soon as it became available. The circuit shown below uses a few simple gates to make an edge-triggered latch. This latch will maintain proper discipline on level sensitive handshake lines.

Interface Connections



Peripheral Timing



The handshake interface is controlled by hardware and takes no special program structure. The basic program used in the "Manual Input" and "Strobed Input" sections will also work for a handshake input. For applications requiring only one LABEL A, the simple program structure shown below can be used.

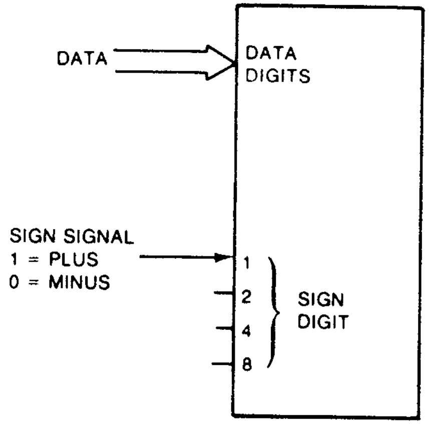
```
001  *LBLA      21 11
002    CF3   16 22 03
003   PRTX        -14
004    R/S         51
```

Negative Numbers and Decimal Point Scaling

Many peripherals output something other than positive integers. The HP-97S is capable of handling positive, negative, fixed-point, floating-point, scientific, and engineering data input formats. The interface accepts these various number types by the proper combining of data digits and preset instructions. The great variety of peripheral devices make it impractical to cover every data format in detail. The many examples in this chapter represent a cross section of the various data formats. Most applications will use some combination or variation of the examples shown here.

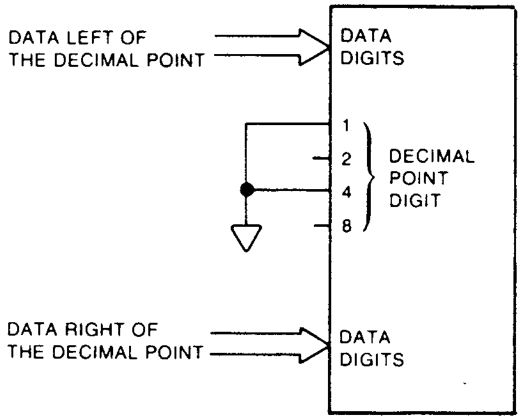
Negative Numbers

Negative numbers are entered by using the sign signal to control a CHS instruction. The CHS digit should immediately follow the data digits. By controlling the least significant bit of that digit, the instruction is a NO-OP for a positive number and a CHS for a negative number. A typical diagram is shown below.



Fixed-Point Numbers

For the purposes of this manual, fixed-point data is data containing a decimal point, and the position of the decimal point is known before the data is available. Two fixed-point techniques will be shown. The first is used when the position of the decimal point never changes. To configure this type of input, simply wire a decimal point instruction at the appropriate position. The following diagram shows an example:



In the second example, the position of the decimal point is not fixed, but the calculator knows the position before the data is input. This method can be used when the HP-97S sets the range of a remotely controlled device. One or more of the HP-97S flag outputs can set the range of the peripheral and a scale factor is stored in the calculator memory. The data is then input as an integer and adjusted by program according to the stored factor.

As an example, consider a digital voltmeter with a X.XXX data format. The meter can be remotely set to read volts or millivolts. Flag 0 is used to select the range and R_0 is used to store the scaling factor. The sample program listing is shown on the next page.


```
001  *LBLE        21 15
002    CF0     16 22 00
003    EEX          -23
004      3           03
005   STO0        35 00
006   GTOB        22 12
007  *LBLe     21 16 15
008    SF0     16 21 00
009    EEX          -23
010      6           06
011   STO0        35 00
012  *LBLB        21 12
013    CF3     16 22 03
014    R/S           51
015  *LBLA        21 11
016   RCL0        36 00
017      ÷          -24
018   PRTX          -14
019    R/S           51
```

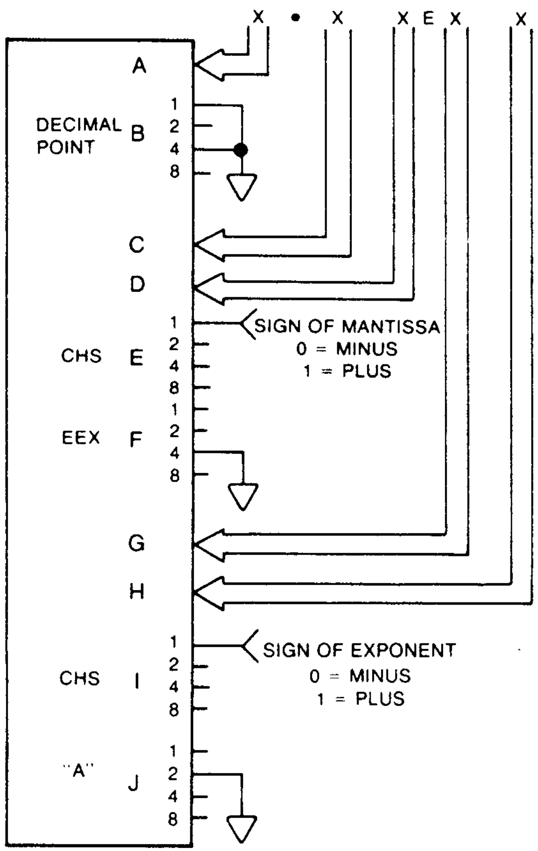
To enter a reading in volts, begin program execution at LABEL E. This example assumes that clearing flag 0 directs the voltmeter hardware to the volts range. Then 10^3 is stored in R_0 and the program branches to LABEL B. Here the data is input, followed by a "continue at A". At LABEL A, the data is divided by 10^3 to convert the integer **XXXX** to the fixed-point number **X.XXX** representing volts.

If a millivolt reading were desired, program execution is started at LABEL e. Setting flag 0 directs the voltmeter hardware to the millivolt range. Then 10^6 is stored in R_0 and the data is input. This time when the steps at LABEL A are executed, the data is divided by 10^6 to convert the integer **XXXX** to the fixed-point number **.00XXXX** representing millivolts.

Scientific Notation and Floating-Point Data

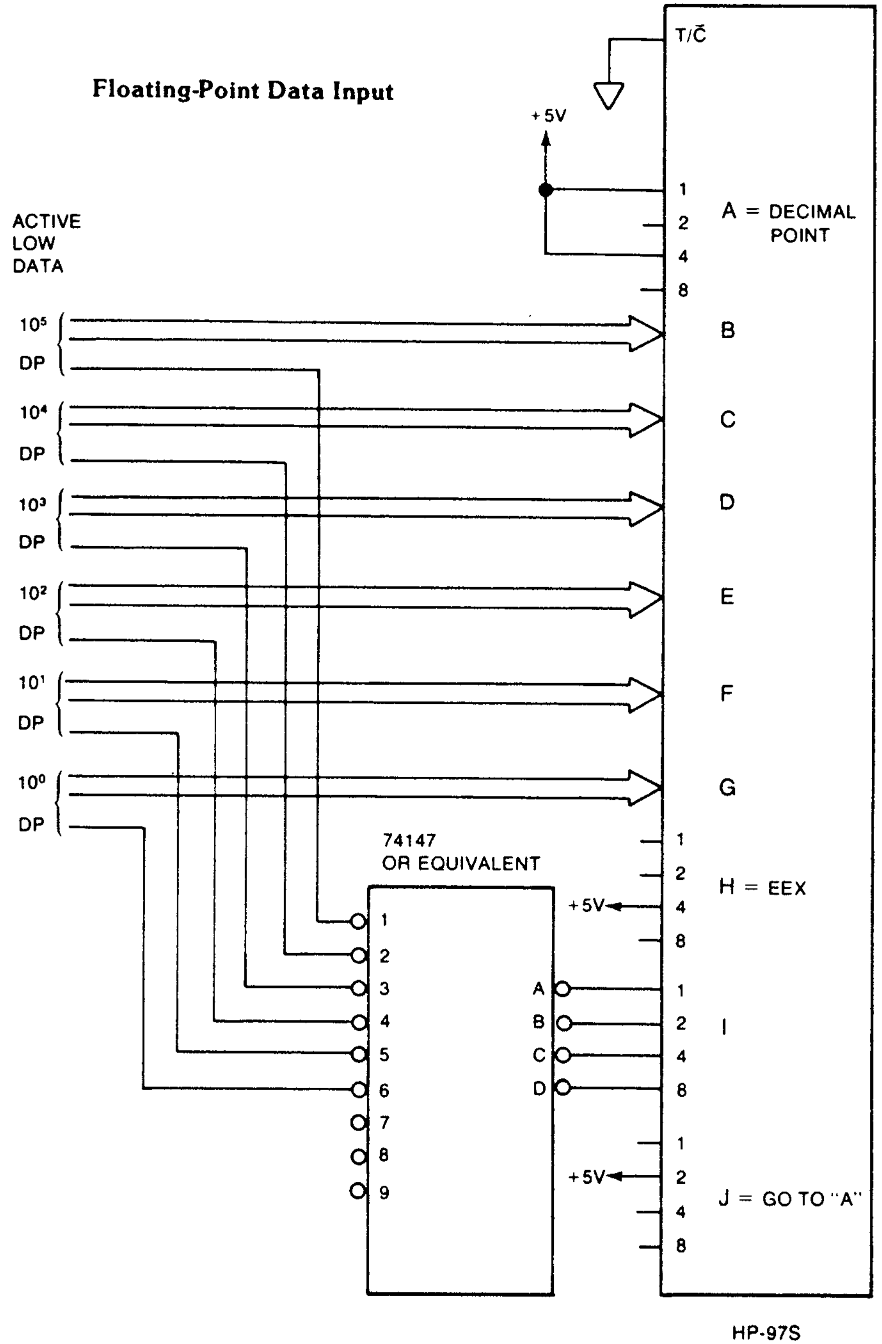
Data in scientific or engineering notation is input by proper use of the EEX, CHS, and decimal point instructions. The HP-97S will accept data with a positive or negative mantissa and a positive or negative exponent of one or two digits. The example shows three significant figures with all sign and exponent digits utilized. More significant figures can be accommodated if less exponent or sign digits are needed.

Inputting Data in Scientific Notation



Floating-point data is most easily handled by first converting it to exponential format. The exact circuit used will depend upon the logic level of the data, the number of significant digits, the range of the data, and whether the decimal point signal represents a left-hand or right-hand decimal point. The sample circuit on the next page is used for six significant digits of negative-true data. The "DP" lines represent a decimal point signal which goes low whenever the point is to the right of the corresponding digit. Therefore, the range of this floating-point format is **XXXXXX.** to **X.XXXXX**.

Floating-Point Data Input

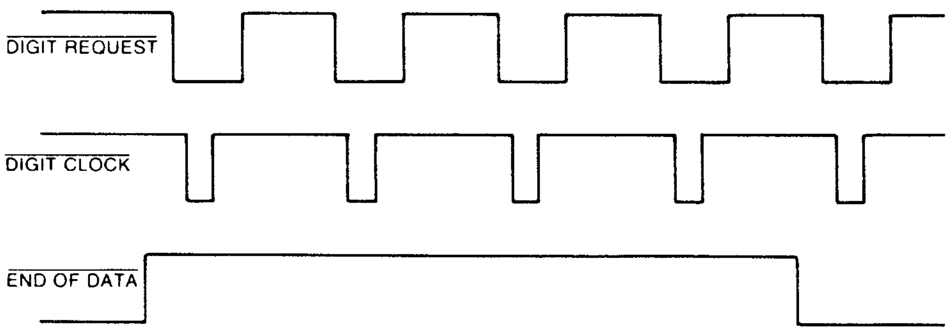


Serial Interface

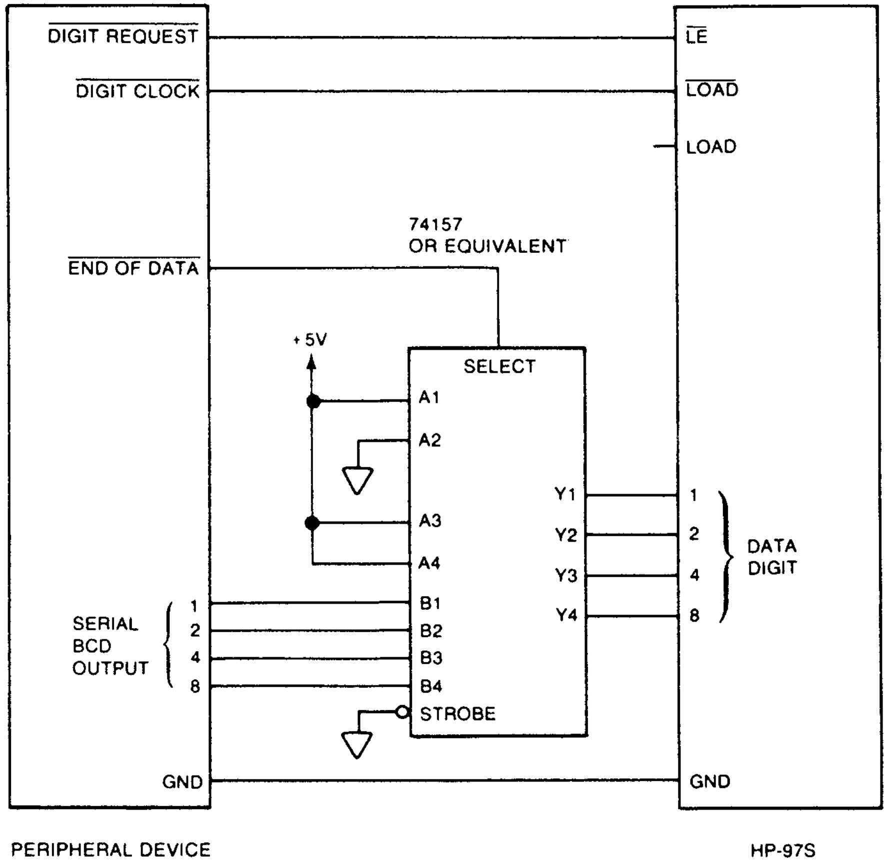
The HP-97S can accept data from a BCD character-serial device. Basically, the data lines are connected to one input with all other inputs left unconnected. Each digit is requested by the calculator with the $\overline{\text{LE}}$ output. Upon receiving the request, the peripheral provides the next digit and a digit clock pulse. An "end of data" line is used to select a "continue at A" instruction when all the digits in the serial string have been input. The following diagrams show the typical connections and timing.

This type of interface is controlled by hardware and does not require any special program structure. The programming techniques provided in the other examples are also applicable to a serial interface.

Typical Peripheral Timing to Input For Serial Characters



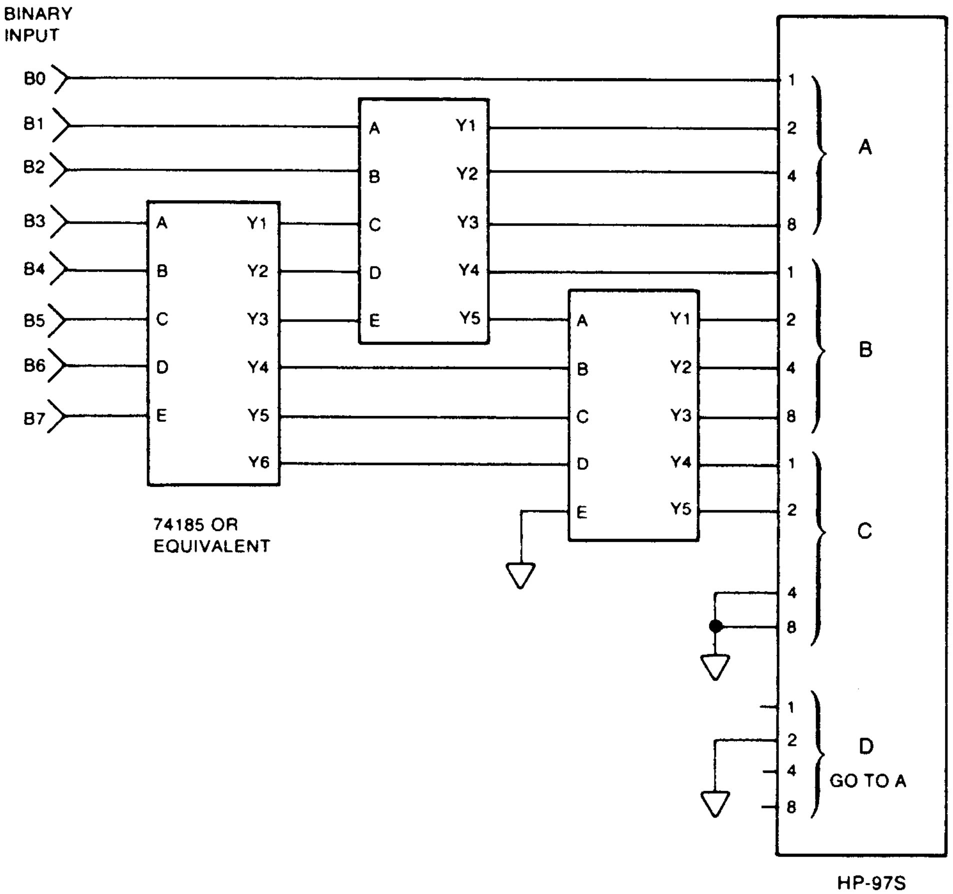
Interface Connections



Binary Interface

The HP-97S is intended to input BCD (binary code decimal) data only. If it is used without extra decoding to input straight binary data, the 4-bit combinations greater than 1001 will be interpreted as instructions, rather than numbers. If your application requires the HP-97S to monitor binary characters, those characters must be converted to their BCD equivalents before passing through the interface. The following circuit is an example of the decoding used. This circuit converts an 8-bit binary number to its decimal equivalent from 0 to 255.

8-Bit Binary Interface

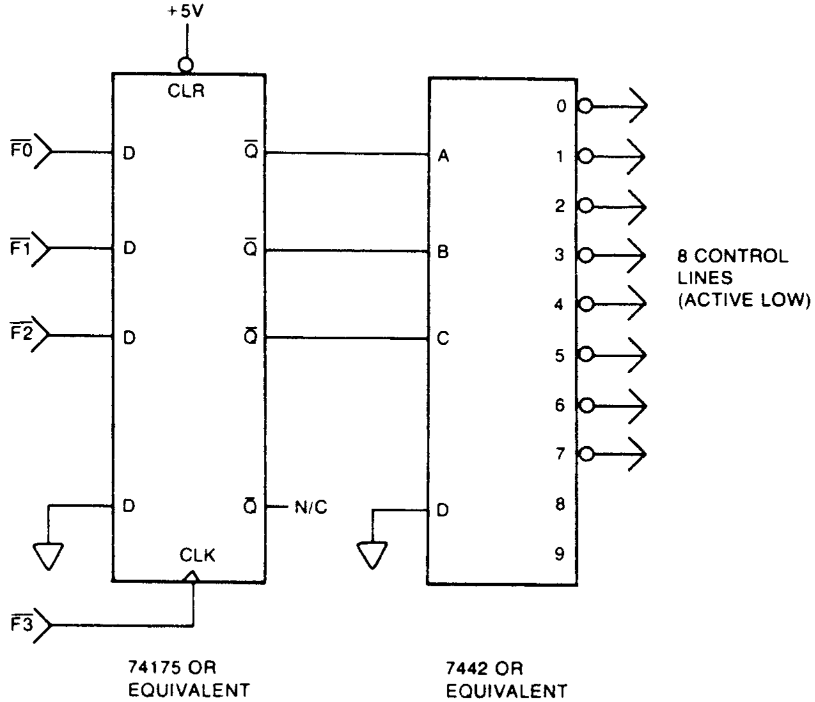


Expanded Output Control

Some applications may require more than four lines of output control. Four flag outputs (with the data entry restrictions on flag 3) is the maximum number of mutually independent control signals available. However, these can be decoded into eight individual control lines for operations like sequencing where only one line at a time needs to be active. In the following circuit, flags 0 through 2 are set as desired. Then flag 3 is cleared, and the state of flags 0 through 2 is latched and decoded to one of eight outputs. The use of the latch avoids any unwanted outputs while the flags are being changed one-by-one.

The sample program listing shows the steps that would be used to activate output line #1, input data, activate output line #4, and take another data input.

Generating 8 Output Lines



Program Listing

```
001  *LBLB        21 12
002    SF0    16 21 00
003    CF1    16 22 01
004    CF2    16 22 02
005    CF3    16 22 03
006    R/S          51
007  *LBLA        21 11
008   PRTX         -14
009    CF0    16 22 00
010    SF2    16 21 02
011    CF3    16 22 03
012    R/S          51
013  *LBLA        21 11
014   PRTX         -14
015    R/S          51
```

```
        086      −
 7.488000000   ***
        087      ÷
 4.849371044   ***
        088   STO8
        089      ×
 176.1049095   ***
        090   RCL6
 94.32000000   ***
        091     x²
 8896.262400   ***
        092   RCL9
 3.000000000   ***
        093      ÷
 2965.420800   ***
        094    CHS
-2965.420800   ***
        095   RCL7
 3141.740000   ***
        096      +
```

Part Three
Maintenance

Introduction

The HP-97S is a quality instrument which is thoroughly inspected before shipment. With proper care it will provide reliable service throughout its long life. The calculator has a full one-year warranty, and Hewlett-Packard repair service will always be available for your HP-97S should the unit malfunction.

This final part of the manual explains the use of the diagnostic aids supplied with the HP-97S, provides information on the proper care of your calculator, and explains Hewlett-Packard's repair policy.

Section 5

System Testing

Interface Diagnostic Program

This program is used with the interface test connector to verify the operation of the HP-97S interface and calculator. With the calculator turned off, plug the test connector into the interface. Then turn the calculator on, insert the program card, and press **B** . The calculator should load numbers through the interface and print them. The program takes about nine seconds to run. A properly functioning interface and calculator will produce the following printout.

```
 1.93000000+01  ***
 1.93000000+01  ***
 4.70000000+01  ***
-5.06000000+02  ***
 2.00000000+00  ***
```

If the program does not run or the printout does not match the one shown above, carefully repeat the procedure. If there are still errors, a malfunction is indicated. This test will not determine which portion of the system is causing the malfunction. The calculator portion can be tested separately using the calculator diagnostic program explained in the next section.

The interface test program will not run unless the interface test connector is properly installed. The program cannot be run a second time simply by pressing **B** again. The program card must be loaded before each running of this test.

Calculator Diagnostic Program

This program is used to test the calculator and diagnose calculator malfunctions. There should be nothing plugged into the interface while this program is being used. Leaving the interface disconnected will cause the INHIBIT line to go high, which should prevent the interface from interfering with the calculator during the test. Load both sides of the program card and press **A** . After a couple of seconds, the calculator should pause displaying:

−7.777777770 −77

Following this one second pause, the calculator should continue to run for about 50 seconds. At the end of the program, the following printout should be generated.

```
     1.+07  ***
 10.000+00  ***
 1.0000+07  ***
```

If the program will not load, does not run, stops prematurely, or produces an incorrect display or printout, carefully repeat the procedure. If there are still errors, a malfunction of the calculator portion of the HP-97S is indicated. If the calculator stops before the printing begins, a code number relating to the malfunction will be displayed. It will be helpful to mention the displayed number along with any other symptoms when sending the unit for repair. A description of the code numbers is contained in the "Diagnostic Program" section of the HP-97 Standard Pack Manual. If the calculator diagnostic program runs properly, but the interface diagnostic program fails, the malfunction is probably in the interface. This too would be helpful information to give to the Service Center.

Section 6

Service and Repair

Routine Care and Service

The HP-97S is designed to give trouble-free operation with a minimum of maintenance. Routine maintenance of the HP-97S involves little more than treating the unit properly and charging the battery when it is low. Details on the care of the calculator can be found in the "Service and Maintenance" appendix of the HP-97 Owner's Handbook. Summarized here are some important highlights of that section.

CAUTION

It is very important that you use only the proper Hewlett-Packard battery pack, battery charger, and thermal printer paper with your HP-97S. Improper substitutions may result in damage to the calculator.

NOTE: The HP-97S will not operate properly from the AC line if the battery pack is removed. The battery pack must be installed and making proper contact for portable or line operation.

Storage of Printer Paper

The printer in the HP-97S uses a special heat-sensitive paper which is available from your nearest HP distributor. Any paper rolls or printouts which are stored should be stored in a cool, dark place away from the fumes of ammonia or organic compounds.

Care of Magnetic Cards

The magnetic cards used with the HP-97S should be treated carefully. Oil, grease, or dirt on the cards can cause them to be misread and may degrade the performance of the card reader. Cards may be cleaned with alcohol and a soft cloth. Do not scratch the oxide coating on the cards and do not expose programmed cards to strong magnetic fields.

Care of Recording Head

An accumulation of dirt on the recording head can cause improper reading and recording of cards. If you suspect that the head is dirty, or have been having trouble reading or recording cards, use the head cleaning card supplied in the HP-97 Standard Pack. However, keep in mind that the cleaning card is abrasive and can cause abnormal wear on the recording head if used excessively. It should not be used frequently or in a routine "preventive maintenance" fashion. If a few passes of the cleaning card do not fix the card reader, the problem is probably not a dirty recording head. Other problems might be a discharged battery, a dirty or scratched card, or a calculator which is too cold.

Warranty

Full One-Year Warranty

The HP-97S is warranted against defects in materials and workmanship for one (1) year from the date of delivery.During the warranty period, Hewlett-Packard will, at its option, repair or replace equipment which proves to be defective. All equipment needing repair or replacement must be returned, shipping prepaid, to an authorized Hewlett-Packard Customer Service Facility. (Refer to Shipping Instructions.)

Repairs necessitated by misuse of the equipment, acts of God, or by software or interfacing not provided by Hewlett-Packard are not covered under this warranty. **No other warranty is expressed or implied. Hewlett-Packard specifically disclaims the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard is not liable for consequential damages.**

Obligation to Make Changes

Products are sold on the basis of specifications applicable at the time of sale. Hewlett-Packard shall have no obligation to modify or update products once sold.

Repair Policy

Repair Time

Hewlett-Packard calculators are normally repaired and reshipped within five (5) working days of receipt at any Customer Service Facility. This is an average time and could possibly vary depending upon time of year and work load at the Customer Service Facility.

Shipping Instructions

The calculator should be returned, along with completed Service Card, in its shipping case (or other protective package) to avoid in-transit damage. Such damage is not covered by warranty, and Hewlett-Packard suggests that the customer insure shipments to the Customer Service Facility. A calculator returned for repair should include the ac adapter/recharger and the battery pack. Send these items to the address shown on the Service Card.

Shipping Charges

Whether the calculator is in-warranty or out-of-warranty, the customer should prepay shipment to the Hewlett-Packard Customer Service Facility. During warranty, Hewlett-Packard will prepay shipment back to the customer.

Further Information

Service contracts are not available. Calculator circuitry and design are proprietary to Hewlett-Packard, and Service Manuals are not available to customers.

Should other problems or questions arise regarding repairs, please call your nearest Hewlett-Packard Sales Office or Customer Service Facility.

Interface Diagnostic Program
Annotated Listing

001	*LBLE	I = 0; select pattern #1 on test connector
002	CLRG	
003	SF1	
004	RTN	
005	*LBLA	Wait for interface input
006	ISZI	I = I+1; Flag 3 = 0; go to selected routine
007	F3?	
008	GTOI	
009	RTN	
010	*LBL1	Print "1.93 +81"; select pattern #2 on test connector
011	PRTX	
012	SF2	
013	CF1	
014	RTN	
015	*LBL2	Print "1.93 +81"; select pattern #3 on test connector
016	PRTX	
017	SF3	
018	CF2	
019	SF1	
020	RTN	
021	*LBL3	Print "47"; Print "−506"
022	PRTX	
023	R↓	
024	PRTX	
025	PSE	
026	PRTX	Input "2" and print; Flag 3 = 1
027	SF3	
028	RTN	
029	R/S	

Calculator Diagnostic Program
Annotated Listing

001	*LBLA	Clear registers
002	CLRG	
003	P⇄S	
004	CLRG	
005	CF3	Test digit entry
006	7	
007	.	
008	7	
009	7	
010	7	
011	7	
012	7	
013	7	
014	7	
015	7	
016	CHS	
017	EEX	
018	CHS	
019	7	
020	7	
021	X⇄Y	Test stack manipulations and stack registers
022	R↑	
023	R↑	
024	R↑	
025	R↑	
026	R↓	
027	PSE	Test display
028	*LBL0	Test registers
029	STOI	
030	RCLI	
031	X≠Y?	
032	GTO1	
033	ISZI	
034	RCLE	
035	RCL0	
036	X=Y?	
037	GTO2	
038	GTO0	
039	*LBL1	Error routine for registers test
040	RCLI	
041	RTN	
042	*LBL2	Start function checks
043	2	
044	5	
045	STOI	
046	SIN	Test sin, $\sin^{-1}$
047	SIN^{-1}	
048	GSB3	
049	COS	Test cos, $\cos^{-1}$
050	COS^{-1}	
051	GSB3	
052	TAN	Test tan, $\tan^{-1}$
053	TAN^{-1}	
054	GSB3	

055	→P	Test rectangular and polar functions
056	→R	
057	GSB3	
058	SIN	Test hours, minutes, seconds conversion
059	→HMS	
060	HMS→	
061	SIN^{-1}	
062	GSB3	
063	LOG	Test Log and 10^x
064	10^x	
065	GSB3	
066	LN	Test Ln and e^x
067	e^x	
068	GSB3	
069	X^2	Test X^2 and X
070	√X	
071	GSB3	
072	ENT↑	Test Y^x, 1/X and LST X
073	Y^x	
074	LSTX	
075	1/X	
076	Y^x	
077	GSB3	
078	ENT↑	Test + and −
079	+	
080	LSTX	
081	-	
082	GSB3	
083	ENT↑	Test X and ÷
084	×	
085	LSTX	
086	÷	
087	GSB3	
088	1/X	Test INT and FRAC
089	1	
090	+	
091	FRC	
092	1/X	
093	LSTX	
094	+	
095	INT	
096	GSB3	
097	D→R	Test degree and radian conversions
098	R→D	
099	GSB3	
100	EEX	Test %
101	2	
102	X⇄Y	
103	%	
104	GSB3	
105	GTO4	

106	*LBL3	Function check and error code routine
107	RND	
108	RCLI	
109	X≠Y?	
110	R/S	
111	ISZI	
112	RCLI	
113	RTN	
114	*LBL4	Check X−Y comparsons
115	1	
116	-	
117	RCLI	
118	X≤Y?	
119	RTN	
120	ISZI	
121	2	
122	+	
123	RCLI	
124	X>Y?	
125	RTN	
126	ISZI	Check X−0 comparisons
127	RCLI	
128	X=0?	
129	RTN	
130	ISZI	
131	RCLI	
132	X≠0?	
133	GTO5	
134	RTN	
135	*LBL5	
136	ISZI	
137	RCLI	
138	X<0?	
139	RTN	
140	ISZI	
141	RCLI	
142	X>0?	
143	GTO5	
144	RTN	
145	*LBL5	Flag off tests
146	ISZI	
147	RCLI	
148	F0?	
149	RTN	
150	ISZI	
151	RCLI	
152	F1?	
153	RTN	
154	ISZI	
155	F2?	
156	RTN	
157	ISZI	
158	RCLI	
159	F3?	
160	RTN	

161	ISZI	
162	SF0	
163	SF1	
164	SF2	
165	SF3	
166	F0?	
167	GTO6	
168	RTN	
169	*LBL6	
170	ISZI	
171	RCLI	
172	F1?	
173	GTO6	Flag on tests
174	RTN	
175	*LBL6	
176	ISZI	
177	RCLI	
178	F2?	
179	GTO6	
180	RTN	
181	*LBL6	
182	ISZI	
183	RCLI	
184	F3?	
185	GTO6	
186	RTN	
187	*LBL6	
188	EEX	
189	7	
190	PRTX	
191	ENG	
192	DSP4	Formatting check
193	PRTX	
194	SCI	
195	PRTX	
196	CF0	Clear flags for next run
197	CF1	
198	FIX	
199	DSP2	
200	RTN	Return to standard display format
201	R/S	

Index

A

B

C

D

E

F

G

H

I

L

M

N

O

P

R

Notes