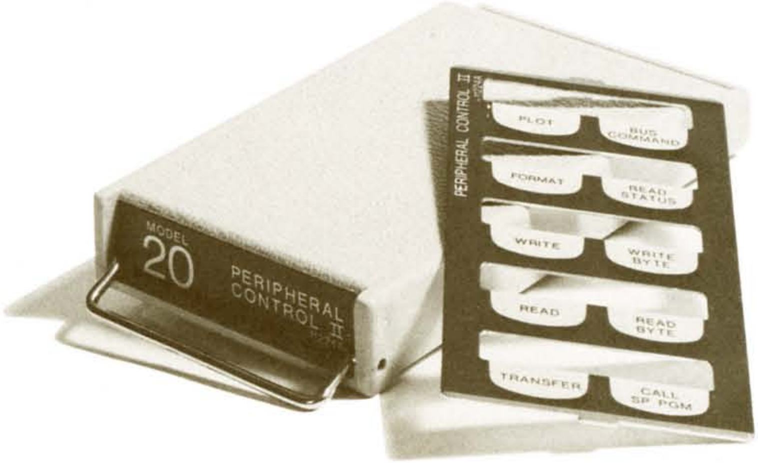
MODEL

20

HEWLETT PACKARD 9820A CALCULATOR
11224A PERIPHERAL CONTROL II

OPERATING MANUAL

OPERATING MANUAL



11224A PERIPHERAL CONTROL II BLOCK

HEWLETT–PACKARD CALCULATOR PRODUCTS DIVISION
P.O. Box 301, Loveland, Colorado 80537, Tel. (303) 667-5000
Rue du Bois-du-Lan 7, CH-1217 Meyrin 2, Geneva, Tel. (022) 41 54 00

TABLE OF CONTENTS

CHAPTER 1. GENERAL INFORMATION

INTRODUCTION TO PC II 1-1

SUPPLIED EQUIPMENT 1-1

INITIAL INSPECTION 1-1

INSTALLATION PROCEDURE 1-2
ROM Block Memory Usage 1-2

CHAPTER 2. PERIPHERAL CONTROL OPERATIONS

INTRODUCTION 2-1

INTERFACING THE MODEL 20 2-1
Interface Kits 2-2
Peripheral Select Code 2-2
Select Code Specification 2-2
Input-Output Codes 2-2

WRITE STATEMENTS 2-3
The WRITE Syntax 2-3
Delimiters 2-3
Free-Field Formatting 2-3
The WRITE BYTE Syntax 2-4

READ STATEMENTS 2-4
The READ Syntax 2-4
Data Items 2-4
Free-Field Format 2-4
The READ BYTE Syntax 2-5

FORMAT STATEMENTS 2-5
The FORMAT Syntax 2-5
Formatting Output - Conversion Specs. 2-6
Deleting Leading Spaces 2-7
Formatting Output - Edit Specs. 2-7
Keyboard Execution of Output Statements 2-9
Formatting Input - Conversion Specs. 2-9
Formatting Input - Edit Specs. 2-10

THE TRANSFER STATEMENT 2-11

ASCII BUS CONTROL 2-11

SPECIAL PROGRAMS 2-12

PERIPHERAL STATUS CHECK 2-13

APPENDIX

DIAGNOSTIC NOTES

KEY MNEMONICS

KEYBOARD OVERLAY

This manual is intended to contain all operating information for the calculator peripheral devices which are controlled with the Peripheral Control II Block. The chapter index to the right lists the devices that are now described in this manual; you may use the index to quickly find operating information on each peripheral device.

As you add other suitable peripherals to your Model 20 System, each will be supplied with operating information that should be added to this manual.

The instructions in this manual assume that the reader is familiar with operation of the basic Model 20 Calculator, as described in the Model 20 Operating and Programming Manual.

PERIPHERAL CONTROL OPERATIONS

PLOTTER CONTROL

TYPEWRITER CONTROL

DIGITIZER CONTROL

TAPE READER CONTROL

NOTES

Chapter 1

GENERAL INFORMATION

INTRODUCTION TO PC II

The -hp- Model 11224A Peripheral Control II Block (the PC II Block) consists of a read-only-memory (ROM Block) and a keyboard overlay. The PC II Block enables the Model 9820A Calculator to control and send data to, or receive data from, many -hp- 9800-Series calculator peripherals. Also, the PC II Block enables the Model 20 to control many other devices when they are properly interfaced to the calculator. Chapter 2 contains general interfacing and operating information which should help you determine if your specific device can be operated in a Model 20 System.

SUPPLIED EQUIPMENT

The items supplied with the PC II Block are listed below.

Table 1-1. Equipment Supplied.

DESCRIPTION	QUANTITY	-hp- PART NUMBER
Key Overlay	1	7120-3314
Operating Manual	2	09820-90024
	Supplements to the Model 20 Electrical Inspection Booklet:	
Supplement E	1	09820-90058
Supplement G	1	09820-90061
Supplement H	1	09820-90063
Supplement I	1	09820-90065

INITIAL INSPECTION

The PC II Block and the equipment listed in Table 1-1 were carefully inspected before they were shipped to you. Please verify that all the equipment listed is present, and inspect the ROM block for physical damage.

To check operation of the PC II Block, see the Model 20 Electrical Inspection Booklet, which is supplied with your calculator.

If any damage or electrical malfunction is found, contact the nearest -hp- Sales and Service Office; office locations are listed at the back of this manual.

INSTALLATION PROCEDURE

As with other ROM blocks, the PC II Block may be installed in any of the ROM Slots on top of the Model 20. Also, the PC II Block defines the keyblock directly in front of where it is installed.

Even though a ROM Block can be installed in any of the slots, programs recorded on magnetic cards or tape cassettes dictate that any required block be in a specific slot — namely, the slot that the block was in when the program was recorded. Before loading any recorded program, always check the instructions which accompany the program to determine which ROM's should be installed in which slots.

It is recommended that the PC II Block be installed in ROM Slot #3, since most programs published by -hp- will specify that the PC II Block be in that slot.

To install the PC II Block:

1. Switch the calculator OFF — if you leave the calculator ON, the installed block will not be 'accepted' until the MEMORY ERASE key is pressed.
2. Position the block over the desired slot (ROM Slot #3 is recommended), such that the 'PERIPHERAL CONTROL' label is readable from the front of the calculator (see Figure 1-1). Push the block straight down until it is firmly seated.
3. Install the key overlay by inserting the tab at the top of the overlay into the locking slot at the top of the keyblock; then press the overlay down over the keys.
4. Switch the calculator ON.

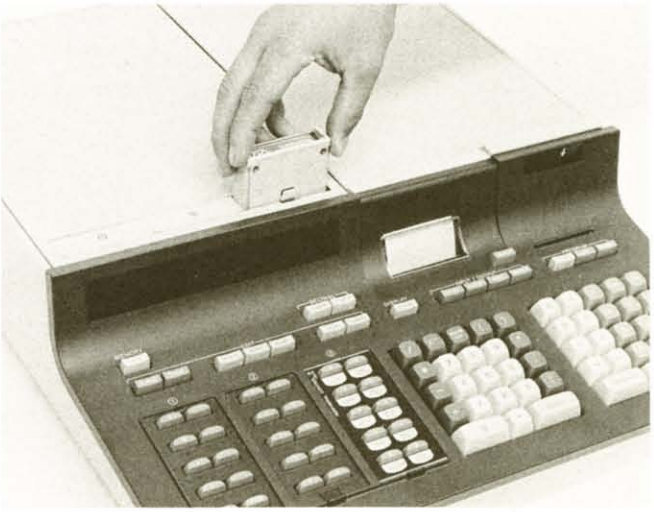


Figure 1-1. PC II Block Installation

ROM BLOCK MEMORY USAGE

When the PC II Block is installed, it requires the use of 12 words of User Read-Write Memory; this usage is indicated by the loss of 3 R-registers, as indicated at the end of any program listing.

Chapter 2

PERIPHERAL CONTROL OPERATIONS

INTRODUCTION

This chapter describes all general control operations which are available with the PC II Block. If you are to use the PC II Block to control only 9800-Series peripherals, you need not read this chapter, since this manual has a separate chapter on each calculator peripheral which requires the PC II Block — see the index at the front of the manual.

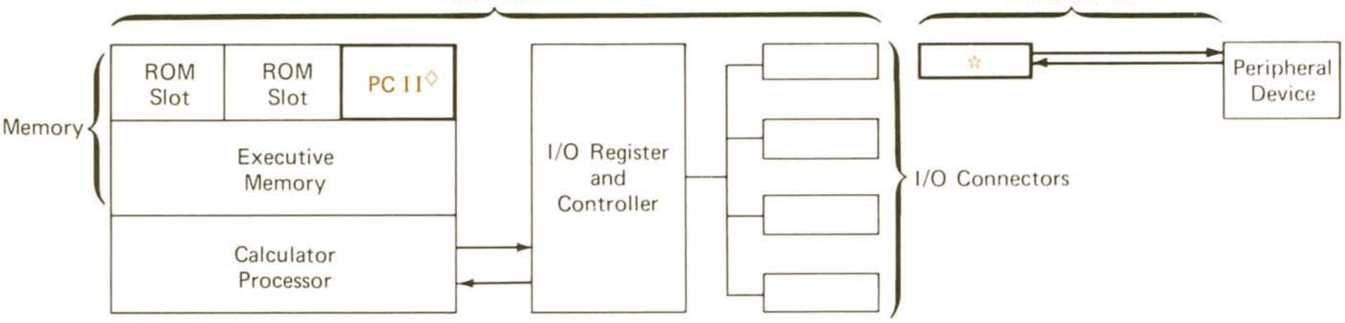
Interfacing a peripheral device with the Model 20 requires both hardware and software; this chapter is primarily concerned with describing the software (i.e., types of peripheral control operations and how to perform them) available with the block. A brief discussion of the Model 20 input-output (I/O) scheme is also provided.

INTERFACING THE MODEL 20

The general I/O scheme for interfacing devices to the Model 20 is shown in Figure 2-1. The four I/O connectors on the calculator are connected in a 'party-line' fashion, thus permitting four* peripheral devices to be connected to the calculator at one time.

Each device must be connected to the calculator through the appropriate interface card and cable. This card provides necessary electrical interface (signal conditioning, buffering, etc.) between the device and the calculator. 9800-Series peripherals are supplied with all required interfacing hardware. Also, interface kits are available for other interfacing applications.

*Use of the Model 9868A I/O Expander permits up to thirteen devices to be connected to the calculator at one time.



☆Select code determined here.

◇Adds I/O routines to the calculator

Figure 2-1. Model 20 I/O Scheme

INTERFACING THE MODEL 20

INTERFACE KITS

Interface kits are available to permit many devices which are not 9800-Series peripherals to be interfaced with the Model 20. The following is a brief description of two such interface kits.

- The -hp- 11202A TTL I/ O Interface enables the calculator to input and output information using standard ASCII codes. For example, the I/O Interface can be used to connect the -hp- Model 2748A Tape Reader or the -hp- Model 2895A Tape Punch to the Model 20 Calculator.

- The -hp- 11203A BCD Input Interface enables the calculator to send a data request (sample) signal to, and receive data from, a wide variety of devices which can output information in a 'binary coded decimal' (BCD) format (e.g., digital voltmeters, electronic counters, etc.).

Each of these interface kits is supplied with detailed 'hardware' related information (electrical specifications, recommended interface circuits, etc.) which should enable you to either interface your device directly to the Model 20 or to build the necessary additional hardware required in order to interface your device. With the proper use of one of these interface kits, the operations to be described may be utilized to control or transfer data to or from your peripheral device.

The nearest -hp- Sales and Service Office can furnish you with data sheets which list complete specifications for each available interface kit.

PERIPHERAL SELECT CODE

Since all peripheral devices are connected in a party-line fashion, each device must have a unique 'address', so that the calculator can specify which device should respond to each operation. This address (or select code) consists of a one or two-digit number and is determined by the interface card. Most 9800-Series peripherals contain a switch that permits the user to set any one of nine select codes, while a few peripherals have a fixed select code (e.g., the plotter select code is 14). Most I/O operations must specify a select code, thereby causing the correct interface card to respond to the operation, while all other cards ignore it.

SELECT CODE SPECIFICATION

The select code specified in I/O statements can be in the form of an integer number, a data register, or an expression. For example, any of the following WRITE statements will be addressed to the device set to select code 9.

```
WRT 9⊢
WRT A⊢     (when A=9)
WRT 3R0⊢ (when R0=3)
```

INPUT-OUTPUT CODES

The PC II Block enables the Model 20 to send (write) and receive (read) data and to send coded commands in the form of standard ASCII* codes. A table of ASCII equivalent codes in on Page 2-8.

Information exchange between the calculator and a peripheral device is handled on a 'handshake' basis, with information being transferred, character-by-character, in an 8-bit parallel, character-series fashion. For example, when sending data, the calculator sends one 8-bit ASCII-coded character at a time; if another character is to be sent, the calculator will wait until the peripheral is ready to accept it.

Also, the READ BYTE and WRITE BYTE statements enable the calculator to receive (or send) information in standard binary form. As each 8-bit code is received (or sent), the calculator automatically converts the information to (or from) its decimal equivalent code. The READ BYTE and WRITE BYTE statements are described later in this chapter.

There is no provision for system 'interrupt' operation when using the PC II Block, thus, a peripheral device cannot initiate or call for an input or output operation. The calculator must be in complete control of each peripheral device while the device is involved in I/O operations. As mentioned earlier, if the device is not ready, the calculator will wait. If required, the calculator can be taken out of the 'wait' status by pressing STOP.

*American Standard Code for Information Interchange.

WRITE STATEMENTS

The WRITE statement is used to output ASCII-coded information to an external device. The WRITE BYTE statement is used to output single, 8-bit binary-coded characters to an external device. The ⟨select code⟩ parameter specifies which device will receive the information.

THE WRITE SYNTAX:

WRT ⟨select code⟩[. format number] ,
⟨parameter$_1$⟩ , ⟨parameter$_2$⟩ , ...
i.e.,
WRT ⟨select code . format number⟩ , ⟨list⟩

Each ⟨parameter⟩ in the list can consist of an integer, a register name (the contents of the register is output), or an expression (the result of the expression is output). The data in each parameter is output in 'free-field' format, if it is in effect, or in the form specified by a FORMAT statement.

If more than one FORMAT statement is used, the 'format number' parameter may be used to reference the desired FORMAT statement (FORMAT statements are described later in this chapter).

DELIMITERS

A delimiter is a character that is used to separate one item from another item inside the list or to terminate the list.

The space (ƀ) and the CR/LF are delimiters that are automatically output during the execution of each WRITE statement. The space (ƀ) is used to separate items within the list, and the CR/LF is used to terminate the list.

FREE-FIELD FORMATTING

The 'free-field' format is automatically set whenever the calculator is turned ON, the MEMORY ERASE key is pressed, or an END instruction is executed. The free-field format causes information to be output in a series of 18-character fields. As each item is output, it appears right-justified in one field. A carriage-return line-feed (CR/LF) is given either after four items have been output or when the list of the output statement is exhausted.

The form in which numeric parameters will appear is determined by the current settings established by the FIXED or FLOAT instructions. A number that is too large to be output under the current fixed-point specification is output under the previous floating-point specification.

The free-field format remains in effect until another format is established with a FORMAT statement (described later).

EXAMPLES:

These examples assume that the free-field format is set and that an appropriate output device such as a Model 9861A Typewriter, is interfaced to the calculator.

1. Execute the following lines:

```
FXD 2⊢
20→R0⊢
5.55→A⊢
WRT 2,R0,A⊢
```

ASCII characters* output:

$\underbrace{\text{ƀƀƀƀƀƀƀƀƀƀƀƀƀ20.00}}_{\text{18-character field}}\underbrace{\text{ƀƀƀƀƀƀƀƀƀƀƀƀƀƀ5.55}}_{\text{18-character field}}$ (CR)(LF)

CR/LF terminates list

2. Execute the following lines:

```
FXD 5⊢
WRT 15,5E9,1/1E5,-20⊢
```

ASCII characters output:

ƀƀƀƀ5.000000000E09ƀƀƀƀƀƀƀƀƀƀƀƀ.00001
ƀƀƀƀƀƀƀƀƀ−20.00000 (CR)(LF)

*ƀ indicates a blank space.

WRITE STATEMENTS

WRITE BYTE

THE WRITE BYTE SYNTAX:

WTB⟨select code⟩,⟨decimal number⟩

The ⟨decimal number⟩ parameter can be expressed as a positive integer, a variable (by using a register name) or an expression. The number must be a positive integer in the range of from '0' to '255'. When this statement is executed, the number is converted to an 8-bit binary-coded equivalent form and sent to the specified peripheral device. A table of decimal/binary equivalent forms is on Page 2-8.

EXAMPLE:

Execute the line:

```
WTB 15,65⊢
```

Result:
The 8-bit code '01000001' (ASCII character 'A') is output to the device set to select code 15.

READ STATEMENTS

The READ statement enables the calculator to receive ASCII-coded numerical data from an external device. The READ BYTE statement permits the calculator to receive single, 8-bit binary-coded characters from an external device. The device which sends the data is specified by the select code in the READ or READ BYTE statement.

READ

THE READ SYNTAX:

RED⟨select code⟩[.format number] ,⟨register name⟩,⟨register name⟩, . . .
i.e.,
RED⟨select code .format number⟩,⟨list⟩

The form of each data item is determined by the referenced FORMAT statement or by the 'free-field format' which is described later. If more than one FORMAT statement is used, the 'format number' parameter is used to reference the required FORMAT statement (see Page 2-6).

DATA ITEMS

The number of parameters in the ⟨list⟩ determines how many data items are read. Each item is stored in its corresponding register.

A data item must consist of only the digits 0 through 9, plus and minus signs, a decimal point, and an 'E' character. All other characters will be treated as data item delimiters. The data item itself can assume the same form as any number which is entered from the keyboard.

FREE-FIELD FORMAT

The free-field format used with READ statements is set by executing the syntax: FMT * . If that syntax is not encountered and a format number is not specified, the READ statement will reference the last encountered FORMAT statement.

Here is a brief description of the data item delimiters used in free-field format; see Page 6-2 for general examples of the READ statement.

- Leading 'spaces' (i.e., any non-numeric characters) in a data item are ignored.

- An initial comma causes the first element to be skipped. Two consecutive commas indicate that no data item is supplied for the corresponding element in the list; in this case the current value of that list element remains unchanged and Flag 13 is set.

- A slash (/) causes the calculator to ignore all following characters until a CR/LF has been encountered.

If a LF is encountered (and it does not correspond to a preceding slash) the READ statement is terminated, the values of any further list elements in the READ statement remain unchanged, and Flag 13 is set.

READ STATEMENTS

- The 'E' character, when part of any of the following forms, causes the preceding data item to be raised by the power of 10 indicated.

1. ⟨data item⟩E⟨one or two digits⟩
2. ⟨data item⟩E⟨a + or – and one or two digits⟩
3. ⟨data item⟩E⟨a space and one or two digits⟩

For example, any of the following data items will be read as the number '1234' (note that the 'E' is missing from the last two data items):

```
1.234E3
1.234Eƀ3
1.234E+3
1.234+3
1.234+03
```

READ BYTE

THE READ BYTE SYNTAX:

RDB ⟨select code⟩[, ⟨register name⟩]

The calculator reads one 8-bit byte (e.g., one 8-line binary-coded character) from the specified device, converts the information to its decimal-equivalent number, and stores the number into the indicated register. If the ⟨register name⟩ parameter is omitted, the number is stored into the Z-register. The range of decimal numbers is from '0' through '255'.

A table of ASCII characters and equivalent decimal codes in on Page 2-8.

Also, the READ BYTE operation can be used as a function within another statement. For example, the following lines perform equivalent operations.

EXAMPLE:

Assume that a paper-tape reader (set to select code 7) is loaded with a tape which contains the following ASCII-coded information:

ƀABCD1.00 (CR)(LF)

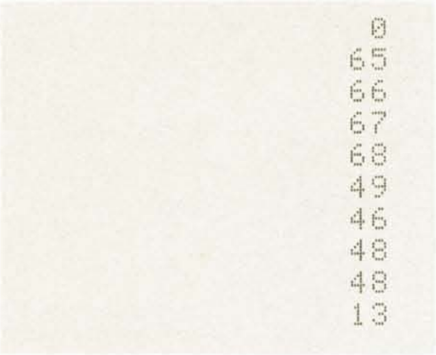
Execute either of the following lines several times:

```
RDB 7;PNT Z⊢
```

or

```
PNT RDB 7⊢
```

Printout:



```
 0
65
66
67
68
49
46
48
48
13
```

FORMAT STATEMENTS

FORMAT statements can be used to precisely control the form in which information is sent or received when using a READ or WRITE statement. In general, a given FORMAT statement can be used to input information in the same form as it was output, provided that it was output by using that FORMAT statement.

FORMAT

THE FORMAT SYNTAX:

FMT [⟨fmt no.⟩ ,] ⟨spec$_1$⟩ , ⟨spec$_2$⟩ , . . .

i.e.,

FMT n , ⟨spec list⟩

Where: n is the format number (0 through 9), and ⟨spec⟩ is either a 'conversion specification' or an 'edit specification'.

FORMAT STATEMENTS

When each FORMAT statement is encountered, its location in memory is noted and the calculator continues with program execution. Then, when a READ or WRITE statement is executed, the I/O statement references any one of ten FORMAT statements by specifying the appropriate 'format number'.

The 'format number' can be any positive integer from '0' through '9'; if the number is omitted, format number '0' is assumed.

If a format number is not specified in an I/O statement, the last encountered FORMAT statement will be referenced, or if a FORMAT statement cannot be found, the 'free-field' format is used.* For example, if the following FORMAT statements are executed within a program,

```
11:
FMT 2,... ⊢
12:
FMT 8,... ⊢
```

then the following WRITE statement will reference 'format number 2' and the READ statement will reference 'format number 8':

```
13:
WRT 15.2,A,B,C⊢
14:
RED 7.8,A,B⊢
```

There is a correspondence between the elements of a FORMAT statement and the parameters in the list of the I/O statement. In general, the correspondence is that the first specification in the FORMAT statement controls the form (appearance) of the first parameter that is input or output; then the second specification controls the form of the next parameter; etc. If the end of the list of parameters is reached before all of the specifications have been used, the I/O operation is terminated. However, if the end of the FORMAT statement is reached before the list of the I/O statement is exhausted, execution of the I/O statement is resumed by repeating each specification of the FORMAT statement.

FORMATTING OUTPUT - CONVERSION SPECS.

Conversion specifications are used to determine the form in which a parameter is output. A conversion specification determines whether the number is output in fixed point or floating point, the number of digits to the right of the decimal point, and the field width in which the number appears (it will be right-justified). A general conversion spec. syntax is shown below:

FMT ⟨integer⟩FXDorFLT⟨integer⟩ . ⟨integer⟩
i.e.,
FMT r FXD w d or FMT r FLT w . d

Where: r is the number of consecutive times the specification is to be used (if r is 1 it may be omitted), and
w is the field width in which the number is to appear, and
d (0 – 9, inclusive) is the number of digits to appear to the right of the decimal point.

(FXD and FLT are the mnemonics for the FIXED N and FLOAT N keys, respectively.)

A conversion specification like FXD 8.2 calls for outputting a fixed-point number with two digits to the right of the decimal point. The number appears (right-justified) in an eight-character field. If d is 0, the decimal point is not output.

A number output under any conversion specification is always correctly rounded according to the number of decimal places specified.

Some guidelines should be observed in selecting w and d. Signs, decimal points, and exponents are part of the number and must fit in the field width specified by w. For fixed-point outputs, w should be greater than or equal to d+3; for floating-point outputs, w should be greater than or equal to d+7.

In general, if a fixed-point specification cannot be met, either because w is not large enough or because the number is simply too large, an attempt is made to output it in floating point (using the same w and d). If the calculator cannot output the number in the field width available, the field is filled with dollar signs.

* The free-field format is described on Page 2-3 and Page 2-4.

FORMAT STATEMENTS

EXAMPLES:

These examples assume that an appropriate output device (such as the Model 9861A Typewriter) is connected to the calculator.

1. Execute the following lines:

```
20→R0⊢
5.55→A⊢
FMT FXD 10.2;WRT 2,R0,A⊢
```

ASCII Characters Output:

```
ƀƀƀƀƀ20.00(CR)(LF)ƀƀƀƀƀƀ5.55(CR)(LF)
```

Now execute the line:

```
FMT FXD 5.0;WRT 2,R0,1E6⊢
```

ASCII Characters Output:

```
ƀƀƀ20(CR)(LF)$$$$$(CR)(LF)
```

2. To output more data items between CR/LF characters, execute the line:

```
FMT 3FXD 10.2;
              WRT 2,1E5,500,20⊢
```

ASCII Characters Output:

```
ƀ100000.00ƀƀƀƀ500.00ƀƀƀƀƀ20.00(CR)(LF)
```

3. To output data items in more than one format between CR/LF characters, execute the line:

```
FMT FXD 10.0,2FLT 10.0;
          WRT 2,1E5,1E5,-1E-10⊢
```

ASCII Characters Output:

```
ƀƀƀƀ100000ƀƀƀƀ1.Eƀ05ƀƀƀ-1.E-10(CR)(LF)
```

DELETING LEADING SPACES

All leading spaces are deleted from an output data item by using the * character in place of a w (field width) parameter in a conversion spec. For example, executing the following program lines:

```
:FMT FXD *.0,2FLT *.0⊢
:WRT 2,1E5,1E5,-E-10⊢
```

would cause this ASCII string to be output (compare this output with the previous example):

```
1000001.Eƀ05-1.E-10(CR)(LF)
```

FORMATTING OUTPUT - EDIT SPECS.

The edit specifications described below are used to control the placement of the output data, to output headings, and to send control characters to an external device. Examples using these edit specs are shown on Page 4-5.

MULTIPLE SPACE SYNTAX:

```
FMT⟨positive integer⟩X
    i.e.,
FMT r X
```

When this spec. is encountered it causes 'r' number of spaces to be output. If r is 1 it may be omitted.

MULTIPLE CR/LF SYNTAX:

```
FMT⟨positive integer⟩/
        i.e.,
FMT r /
```

When this specification is encountered it causes 'r' number of CR/LF's to be output. If r is 1 is may be omitted.

SUPPRESS CR/LF SYNTAX:

```
FMT Z
```

When this specification is encountered it suppresses the automatic CR/LF at the end of the WRITE statement.

MESSAGE OUTPUT SYNTAX:

```
FMT "ascii string or message"
```

When a message output spec. is encountered, the ASCII characters within the quote marks are output. See Table 2-1 for a list of ASCII characters that can be output with the PC II Block.

FORMAT STATEMENTS

Table 2-1. ASCII Output Characters Available with the PC II Block

ASCII Character	EQUIVALENT FORMS★			Model 20 Key
	Binary	Octal	Decimal	
@	1000000	100	64	GO TO
A	1000001	101	65	A
B	1000010	102	66	B
C	1000011	103	67	C
D	1000100	104	68	D
E	1000101	105	69	E
F	1000110	106	70	F
G	1000111	107	71	G
H	1001000	110	72	H
I	1001001	111	73	I
J	1001010	112	74	J
K	1001011	113	75	K
L	1001100	114	76	L
M	1001101	115	77	M
N	1001110	116	78	N
O	1001111	117	79	O
P	1010000	120	80	P
Q	1010001	121	81	Q
R	1010010	122	82	R
S	1010011	123	83	S
T	1010100	124	84	T
U	1010101	125	85	U
V	1010110	126	86	V
W	1010111	127	87	W
X	1011000	130	88	X
Y	1011001	131	89	Y
Z	1011010	132	90	Z
0	0110000	060	48	0
1	0110001	061	49	1
2	0110010	062	50	2
3	0110011	063	51	3
4	0110100	064	52	4
5	0110101	065	53	5
6	0110110	066	54	6
7	0110111	067	55	7
8	0111000	070	56	8
9	0111001	071	57	9
:	0111010	072	58	R ( )
<	0111100	074	60	≤
=	0111101	075	61	=
>	0111110	076	62	>
?	0111111	077	63	?

ASCII Character	EQUIVALENT FORMS ★			Model ☆ 20 Key
	Binary	Octal	Decimal	
NULL	0000000	000	0	SPACE
SOM	0000001	001	1	STOP
EOM	0000011	003	3	≠
EOT	0000100	004	4	$
WRU	0000101	005	5	LOAD
RU	0000110	006	6	&
BELL	0000111	007	7	'
TAB	0001001	011	9	+
LF	0001010	012	10	*
FF	0001100	014	12	,
CR	0001101	015	13	-
DC_0	0010000	020	16	NORMAL
DC_1	0010001	021	17	TRACE
DC_2	0010010	022	18	FIXED N
DC_3	0010011	023	19	FLOAT N
DC_4	0010100	024	20	4
ERR	0010101	025	21	5
SYNC	0010110	026	22	6
LEM	0010111	027	23	7
ƀ	0100000	040	32	SPACE
!	0100001	041	33	STOP
#	0100011	043	35	≠
$	0100100	044	36	$
%	0100101	045	37	%
&	0100110	046	38	&
'	0100111	047	39	'
(	0101000	050	40	(
)	0101001	051	41	)
*	0101010	052	42	*
+	0101011	053	43	+
,	0101100	054	44	,
-	0101101	055	45	-
.	0101110	056	46	.
/	0101111	057	47	/
\	1011100	134	92	√
↑	1011110	136	94	ENT EXP
←	1011111	137	95	→
ACK	1111100	174	124	√
ESC	1111110	176	126	ENT EXP
DEL	1111111	177	127	→
"	0100010	042	34	†
;	0111101	073	59	†
[	1011011	133	91	†
]	1011101	135	93	†

† These are the blank keys (left-hand keyblock) and are shown in the same order as on the keyboard.

★ Decimal numbers are used with the READ BYTE and WRITE BYTE statements.

☆ The Shifted keys are colored light-brown.

FORMAT STATEMENTS

Since the Model 20 can output more ASCII characters than there are keys on the keyboard, the PC II Block provides the calculator with a 'shifted' keyboard. The term 'shifted' does not refer to the shifting of a typewriter or teletype keyboard, and should be considered local to the calculator.

At the start of any literal, the calculator's keyboard is in the unshifted mode; it is placed in the shifted mode by pressing the DISPLAY key. This also places the symbol ⠿ in the display; however, this does not cause the character to be output, as it is merely an instruction to the calculator to shift the keyboard. The calcualtor's keyboard can be returned to the unshifted mode while in the middle of a literal by placing another ⠿ in the literal (the unshifted mode is automatically set at the end of a literal). See Page 4-5 for examples of literals in FORMAT statements.

KEYBOARD EXECUTION OF OUTPUT STATEMENTS

To execute an output statement from the keyboard the desired FORMAT statement must be in the same line as the output statement. Also, any subsequent output statement executed from the keyboard must also be preceded, in the same line, by the desired FORMAT statement. Failure to follow this rule causes the subsequent statement to be output in 'free-field' format and NOTE 22 results. See Example 4 on Page 4-3.

FORMATTING INPUT - CONVERSION SPECS.

Conversion specifications are used to determine which characters are input from a data input string, and what form the data will appear. When a FORMAT statement is referenced by a READ statement, the read operation is not terminated until both CR and LF characters are read (unless the edit spec 'Z' is used). A general conversion spec. syntax is shown below:

FMT⟨integer⟩FXD⟨integer⟩.⟨integer⟩
i.e.,
FMT rFXDw.d

Where: r is the number of consecutive times the specification is to be used (if r is 1 it may be omitted, and

w is the width of the data field to be read, and

d (0 – 9, inclusive) is the number of digits to appear to the right of the decimal point.

A conversion spec like FXD 10.2 calls for reading ten numeric characters (all non-numerics which precede a numeric are read but not entered) and placing the decimal point two places from the right. However, if there is a decimal point in the data item, the conversion specification is 'overridden' and the number is entered as read. Also, if an 'E' character is read, a number of the form '1Edd' is entered in place of the data item.

EXAMPLES:

1. ASCII characters to be read:

   ƀƀ1234 (CR)(LF) 5678910 (CR)(LF)

   Execute the line:

   ```
   FMT FXD 6.3;RED 7,A⊢
   ```

   Result:

   1.234 → reg. A

   Now execute the line again.
   Result:

   567.891 → reg. A

2. To read more than one data item per statement, execute the following line (read the preceding characters):

   ```
   FMT FXD 7.3;RED 7,A,B⊢
   ```

   Result:

   1.234 → reg. A
   5678.910 → reg. B

3. To show how reading a decimal point will override the format specified, read the following ASCII characters:

   ƀƀƀ12.3456 (CR)(LF)

   Execute the line:

   ```
   FMT FXD 10.0;RED 7,A⊢
   ```

   Result:

   12.3456 → reg. A

   If the decimal point was not present, the result would be:

   123456. → reg. A

4. An 'E' character will also override the format specified. Read the following characters:

   ƀƀƀABCDE06ƀƀƀ (CR)(LF)

   Execute the line:

   ```
   FMT FXD 10.0;RED 7,A⊢
   ```

   Result:

   1000000 → reg. A

5. If non-numeric characters are read after data is read but before the specified field is read, the data entry is terminated, even though the peripheral does not stop reading until a CR/LF is read. Characters to be read:

   ƀƀƀ1234ƀ5678 (CR)(LF)

   Execute the line:

   ```
   FMT FXD 15.2;RED 7,A⊢
   ```

   Result:

   12.34 → reg. A

FORMATTING INPUT - EDIT SPECS.

The edit specifications to be described are used to control the peripheral device when executing READ statements.

FREE-FIELD FORMAT SYNTAX:

```
FMT *
```

That syntax should precede each READ statement if a FORMAT statement is not to be referenced.

SKIP CHARACTER SYNTAXES:

FMT ⟨positive integer⟩X
i.e.,
FMT rX

This spec causes the peripheral device to skip 'r' number of characters before reading a data item.

FMT "ascii string"

This spec causes the peripheral to skip a number of characters equivalent to those in the "ascii string".

SKIP CR/LF SYNTAX:

```
FMT Z
```

The peripheral skips one set of CR/LF characters (i.e., not stop reading characters) for each 'Z' spec which is encountered. For example, if the statement to be executed in part 1 of the preceding example appeared as this:

```
FMT FXD 15.3,Z;RED 7,A⊢
```

the device would not respond to the first CR/LF but would stop reading after the second CR/LF, and the data entry would be:

1234567.890 → reg. A

SKIP DATA ITEM SYNTAX:

FMT ⟨positive integer⟩/
i.e.,
FMT r/

The peripheral skips all characters which precede 'r' number of CR/LF's. For example, to read only the last data item in the ASCII string shown below:

1234 (CR)(LF) 56789.0 (CR)(LF) 3.14E2 (CR)(LF)

the following line could be used:

```
FMT *,2/;RED 7,A⊢
```

Result:

314. → reg. A

THE TRANSFER STATEMENT

The TRANSFER statement is used to transfer data, or other ASCII information, from one device to another.

TRANSFER

THE TRANSFER SYNTAX:

TFR ⟨select code$_1$⟩ ; ⟨select code$_2$⟩
i.e.,
'transfer from device$_1$ to device$_2$'

After encountering a TRANSFER statement, the calculator simultaneously receives and transmits the string of characters sent by the transmitting device. TRANSFER statements may be terminated by pressing STOP or by receiving an ASCII 'EOM' (end of message) character (see the table on Page 2-8). The EOM character is the only delimiter which the calculator responds to during a transfer operation.

ASCII BUS CONTROL

BUS COMMAND

THE BUS COMMAND SYNTAX:

CMD "⟨address⟩" ; "⟨instructions⟩"
[; "⟨address⟩" ; "⟨instructions⟩" ; . . .]

This statement is used to control devices which are connected to the calculator through the -hp- Model 11144A ASCII Bus Interface. Since the BUS COMMAND statement automatically specifies select code 13, it cannot be used to control other 9800-Series peripherals.

Each set of 'address-instruction' parameters is executed in turn, and any number of parameter sets may be included in the statement. The ⟨address⟩ parameter contains characters which instruct the ASCII Bus as to which device is the 'talker' (the sending device) and which device is the 'listener' (the receiving device). The ⟨instruction⟩ parameter contains the characters needed to control the receiving device.

EXAMPLE:

The BUS COMMAND statement can be used to program the Model 3330B Frequency Synthesizer and the Model 3570A Gain-Phase Meter. These instruments are part of the -hp- 3042A Network Analyzer System; a block diagram of that system is shown below.

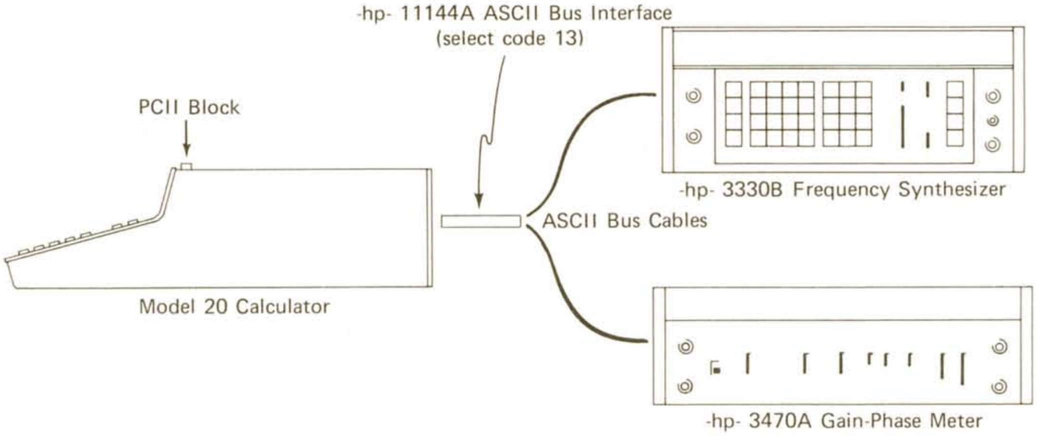

-hp- 11144A ASCII Bus Interface
(select code 13)
PCII Block
-hp- 3330B Frequency Synthesizer
ASCII Bus Cables
Model 20 Calculator
-hp- 3470A Gain-Phase Meter

Figure 2-1. The 3042A Network Analyzer System

ASCII BUS CONTROL

In the 3042A System, the BUS COMMAND syntax is used in the following general form:

```
CMD "?⟨talker address⟩ ⟨listener address⟩","
⟨instruction⟩"
```

These are some of the address characters used with the 3042A System:

? resets all previous addresses on the ASCII Bus.

U is the calculator 'talker' address.

5 is the calculator 'listener' address.

$ is the Model 3330A 'listener' address.

! is the Model 3570A 'listener' address.

A is the Model 3570A 'talker' address.

The ASCII characters used in the ⟨instruction⟩ parameter are described in the manual supplied with the 3042A System.

1. To program the synthesizer for a frequency of 10.8MHz and an output amplitude of −13dbm, the following line could be executed:

```
CMD "?U$","L10.8?N13≤"⊢
        ↑ ↑   └─────┬─────┘
  talker  listener  programming instruction
```

2. To take a reading on the gain-phase meter and store the reading in the A and B registers, the following line could be executed:

```
CMD "?U!","O,","?5A;FMT *;
       ↗  ↖    └┘      ↗ ↖
  talker  listener ↑ data listener data talker
                 instruction   RED 13,A,B⊢
```

More instructions on using the BUS COMMAND statement are given in the manual supplied with the -hp- 11144A Option 20 ASCII Bus Interface.

SPECIAL PROGRAMS

CALL SP PGM

THE CALL SP PGM SYNTAX:

```
CSP "⟨program name⟩"[,⟨parameter⟩, . . .]
```

This statement permits the user to assign parameters, and to run, Special Programs which are available on magnetic cards from -hp-. Special Programs are available to supplement PC II Block operations, provide some functions which are available on other ROM Blocks, or perform unique applications.

When a Special Program is loaded into the calculator, it is stored into the Internal Read-Write Memory (see the memory map in the Appendix of the Model 20 Operating Manual). Also, many Special Programs can be stored in the calculator at one time. Exact instructions on using the CALL SP PGM statement depend upon the particular program; therefore, instructions on using the statement are supplied with each Special Program.

The nearest -hp- Sales and Service Office can supply you with a list of Model 20 Special Programs.

PERIPHERAL STATUS CHECK

READ STATUS

THE READ STATUS SYNTAX:

```
RDS⟨select code⟩ [→⟨register name⟩]
```

The current status of any peripheral device which is set to the indicated select code is checked. A positive integer code is returned to indicate the status of the device. If the '→ register name' instruction is omitted, the status code is stored in the Z-register.

Also, the READ STATUS operation can be used as a function within another statement. For example, although the following two lines perform equivalent operations, the last line is shorter since the READ STATUS operation is used as a function within the IF statement.

```
RDS 5→Z;IF Z=15;
            DSP "OPEN DOOR"⊢

IF RDS 5=15;DSP "OPEN DOOR"⊢
```

The 9800-Series peripherals which can be checked with the READ STATUS statement, and the status codes possible with each peripheral, are listed below.

Table 2-2. Peripheral Status Codes

Peripheral Device	Status Code	Meaning
9861A Typewriter	0 1	Switched OFF or not connected. Switched ON.
9862A Plotter (select code 14)	0 1 3 9	Not connected to calculator. Switched OFF. Ready, pen UP. Ready, pen DOWN.
9863A Tape Reader	0 1	Switched OFF or not connected. Switched ON.
9864A Digitizer	0 1	Switched OFF or not connected. Switched ON.
9865A Cassette Memory	0 1 3 5 7 15 14	Not connected to calculator. Switched ON; tape ready. Switched ON; protected tape cassette. Switched ON; tape on clear-leader. Switched ON; tape on clear leader & protected tape cassette. Switched ON; door open or ajar. Switched OFF.

NOTES

Chapter 3

PLOTTER CONTROL

TABLE OF CONTENTS

CHAPTER 3. PLOTTER CONTROL

INTRODUCTION 3-1

FRONT PANEL CONTROLS 3-1
Line and Chart Hold 3-1
Loading Paper 3-1
Graph Limits 3-1

PLOT STATEMENTS 3-3
The PLOT Syntax 3-3
The PEN UP Syntax 3-3

SCALING THE PLOT 3-4
Examples 3-4

Chapter 3

PLOTTER CONTROL

INTRODUCTION

When the PC II Block is installed, the Model 9862A Plotter can be used to plot graphs or other images being processed by the calculator. The physical size of the finished plot is unrelated to the calculator operations used to create it, since front panel controls on the plotter are used to determine the size of the plot.

FRONT PANEL CONTROLS

LINE AND CHART HOLD

The LINE pushbutton is the power switch for the plotter; press it to apply power and press it again to remove power; the white LINE lamp lights whenever the plotter is ON.

Pressing CHART HOLD activates the electro-static paper hold-down mechanism. Pressing CHART HOLD again deactivates it. The plotter cannot plot when CHART HOLD is deactivated.

LOADING PAPER

To load paper, release CHART HOLD and manually move the pen arm all the way to one side of the plotter. Lay a sheet of paper on the plotting surface and smooth out any irregularities in the paper (also ensure that the paper is squarely against the ridge at the bottom of the plotting surface); then activate CHART HOLD.

GRAPH LIMITS

The Graph Limit controls are used to determine the physical size of the plot.

LOWER LEFT and the two knobs to its left are used to determine the physical location of the lower left-hand corner of the plotting area.

UPPER RIGHT and the two knobs to its right are used to determine the physical location of the upper right-hand corner of the plotting area. Together, the upper right-hand corner and the lower left-hand corner determine the size of the plotting area.

To specify the lower left-hand corner of the plotting area, press LOWER LEFT; the pen will move (without touching the paper) to the lower left-hand corner of the plotting area. This point can be set anywhere within the lower left-hand quarter of the plotting surface (platen) by adjusting the two knobs associated with LOWER LEFT. (If desired, PEN DOWN and PEN UP can be used to mark or determine the exact point on the paper which is the lower left-hand corner of the plotting area.) Once the lower left-hand corner has been set, the upper right-hand corner is set in the same general way by pressing UPPER RIGHT and adjusting the two knobs associated with it. Once the plotting area has been determined, it can be relocated by adjusting the LOWER LEFT controls only – the upper right-hand corner will 'track' the change.

Figures 3-1 and 3-2 show the effect of adjusting the graph limit controls. In each case, the program was the same but the size of the plotting area was changed by adjusting the graph limit controls.

FRONT PANEL CONTROLS

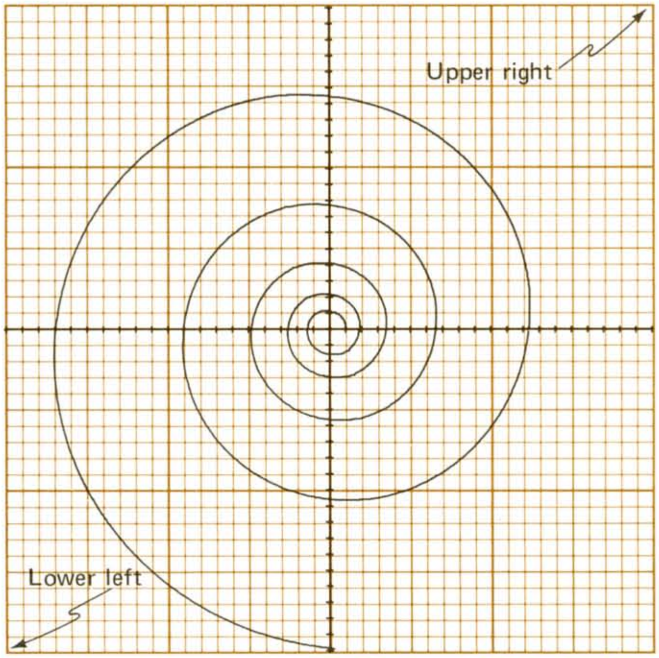


Figure 3-1. A Spiral Plotted with the Graph Limit Controls Set for Four Inches by Four Inches.

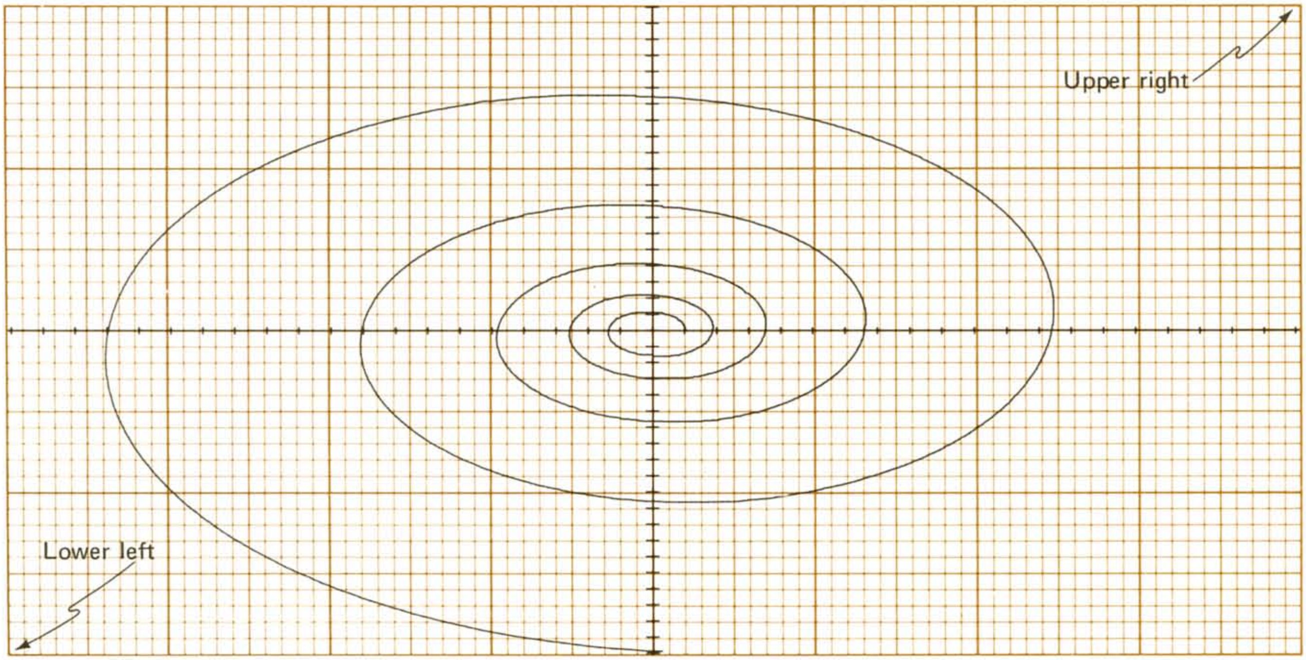


Figure 3-2. The Same Spiral, Plotted with the Graph Limit Controls Set for Four Inches by Eight Inches.

PLOT STATEMENTS

THE PLOT SYNTAX:

PLT ⟨$X_{coordinate}$⟩,⟨$Y_{coordinate}$⟩

The working range of each coordinate is from '0' to '1'*, and each coordinate can be in the form of an integer, a variable (by using a register name), or an expression. If the pen is raised, it will move to the coordinates specified, and then drop. If the pen is already down, it will draw a straight line from its present position to the point specified by the coordinates. Curved lines are plotted by moving the pen in small increments; the series of short, straight line segments produced is often indistinguishable from the actual curve.

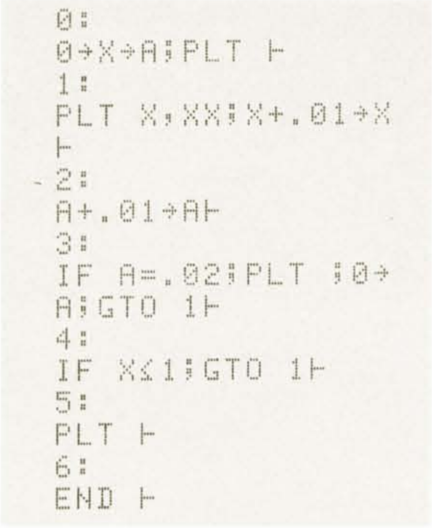
The PEN UP Syntax:

...;PLT;...

When the coordinate parameters are omitted, executing the PLOT statement causes the pen to lift, but not move from, its current position. This syntax can be used when plotting descrete points, plotting dashed lines, and when drawing one graph over another — so that the plotter does not draw a line between the end of one graph and the start of another. A program using 'dashed-line' and 'point' plotting is shown below.

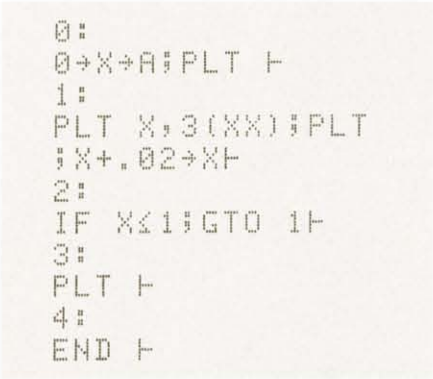
EXAMPLE:

The program for plotting with dashed lines is shown below; the length of each dash (see Figure 3-3) is readily changed by altering the qualifier in the IF statement (line 3).



```
0:
0→X→A;PLT ⊢
1:
PLT X,XX;X+.01→X
⊢
2:
A+.01→A⊢
3:
IF A=.02;PLT ;0→
A;GTO 1⊢
4:
IF X≤1;GTO 1⊢
5:
PLT ⊢
6:
END ⊢
```

The plot of $Y=3X^2$ was made by changing line 1 and deleting lines 2 and 3. The modified program is shown below.



```
0:
0→X→A;PLT ⊢
1:
PLT X,3(XX);PLT
;X+.02→X⊢
2:
IF X≤1;GTO 1⊢
3:
PLT ⊢
4:
END ⊢
```

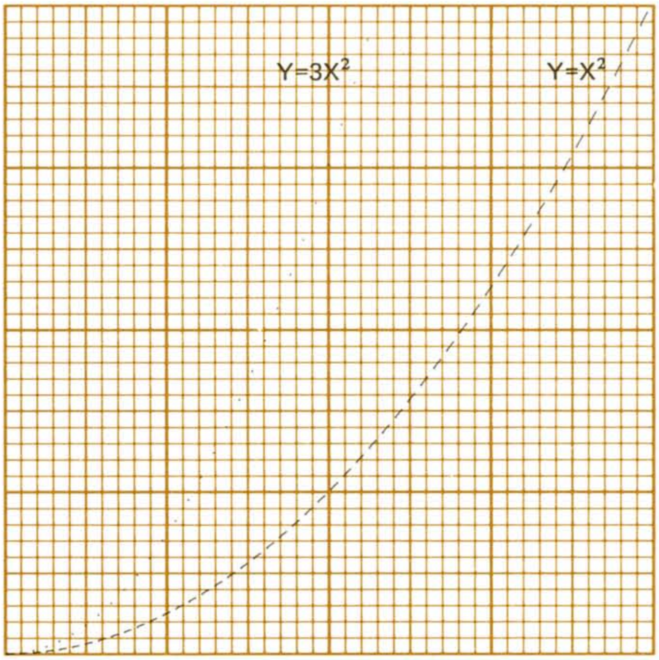


Figure 3-3. Plotting with Dashed Lines and Points.

* See 'SCALING THE PLOT'.

SCALING THE PLOT

As previously mentioned, when using the PC II Block, the range of the plotting area (in either X or Y axes) is from '0' to '1'. This means that, for most applications the user must determine the problem-variable range of his plot and then scale each set of coordinates before they are plotted. This can be done by using the following equations in each PLOT statement:

$$X_{plot} = \frac{X_{coordinate} - X_{min}}{X_{max} - X_{min}}$$

$$Y_{plot} = \frac{Y_{coordinate} - Y_{min}}{Y_{max} - Y_{min}}$$

Where: $X_{coordinate}$ and $Y_{coordinate}$ = coordinate of point to be scaled and plotted

$\left.\begin{matrix} X_{min}, X_{max} \\ Y_{min}, Y_{max} \end{matrix}\right\}$ = problem-variable range limits

X_{plot}, Y_{plot} = scaled coordinates

EXAMPLE A:

If the plotting range is from 0 to 15 on the X axis and 0 to 10 on the Y axis, then the problem-variable range is as follows (see Figure 3-4):

$$X_{min} = 0$$
$$X_{max} = 15$$
$$Y_{min} = 0$$
$$Y_{max} = 10$$

A general PLOT statement which utilizes the scaling equations and the above range limits is this:

```
PLT (X-0)/15-0,(Y-0)/10-0⊢
```

or

```
PLT X/15,Y/10⊢
```

To position the pen at the upper-right corner of the plotting area, execute the line:

```
PLT 15/15,10/10⊢
```

To plot a line to the center of the plotting area, execute the line:

```
PLT 7.5/15,5/10⊢
```

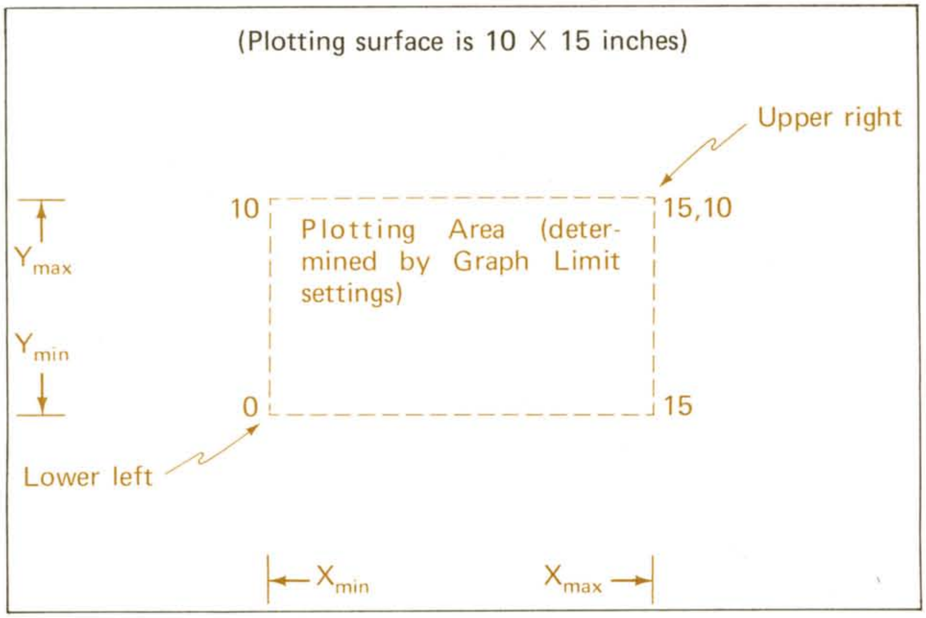

(Plotting surface is 10 X 15 inches)
Upper right
10
15,10
Plotting Area (determined by Graph Limit settings)
Y max
Y min
0
15
Lower left
← X min
X max →

Figure 3-4. The Plotting Surface and the Plotting Area

SCALING THE PLOT

Now plot lines to other points within the plotting area. Notice that when a coordinate greater than '1' is given, the pen raises and moves to the respective side. Once the pen is raised, it will not plot until a set of valid coordinates are given.

EXAMPLE B:

This example shows how to plot a graph of the relationship:

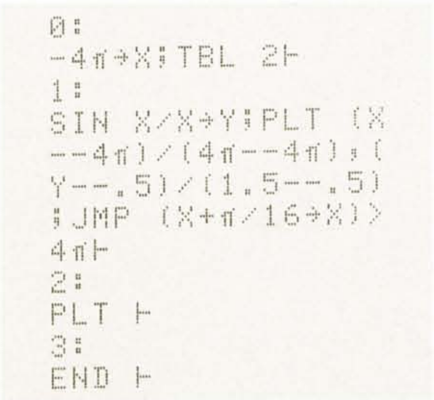
$$y = (\sin x)/x$$

This program requires the use of the Math Block.

The problem-variable range limits (see Figure 3-5) are these:

$X_{min} = -4\pi$ radians
$X_{max} = 4\pi$ radians
$Y_{min} = -.5$
$Y_{max} = 1.5$

The program used to generate the plot in Figure 3-5 is shown below.



```
0:
-4π→X;TBL 2⊢
1:
SIN X/X→Y;PLT (X
--4π)/(4π--4π);(
Y--.5)/(1.5--.5)
;JMP (X+π/16→X)>
4π⊢
2:
PLT ⊢
3:
END ⊢
```

The PLOT statement in line 1 could be expressed as follows:

$$\text{PLT } (X-X_{min})/(X_{max}-X_{min}),$$
$$(Y-Y_{min})/(Y_{max}-Y_{min})$$

Notice that the 'sinX/X → Y' operation could be performed in the PLOT statement instead of in a separate statement.

The plot in Figure 3-6 was obtained by merely changing the values for X_{min} and X_{max} to -8π radians and 8π radians, respectively.

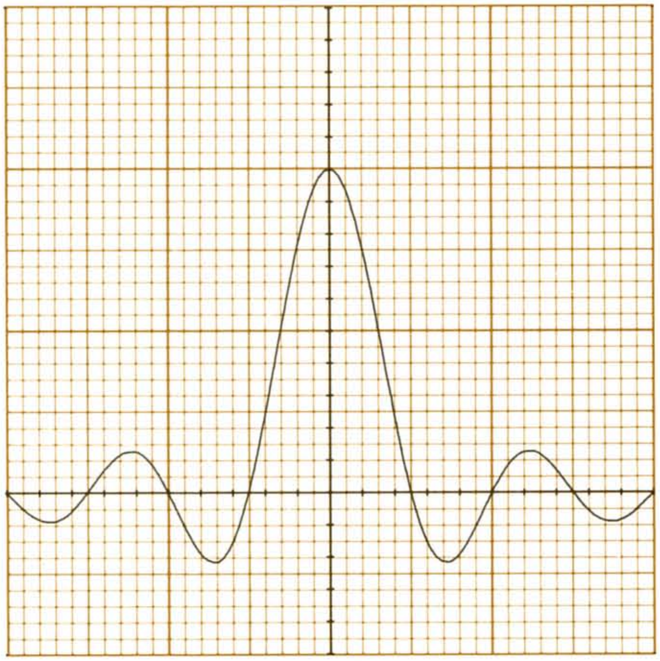


Figure 3-5. Plotting from -4π to $+4\pi$, while Scaling from -4π to $+4\pi$.

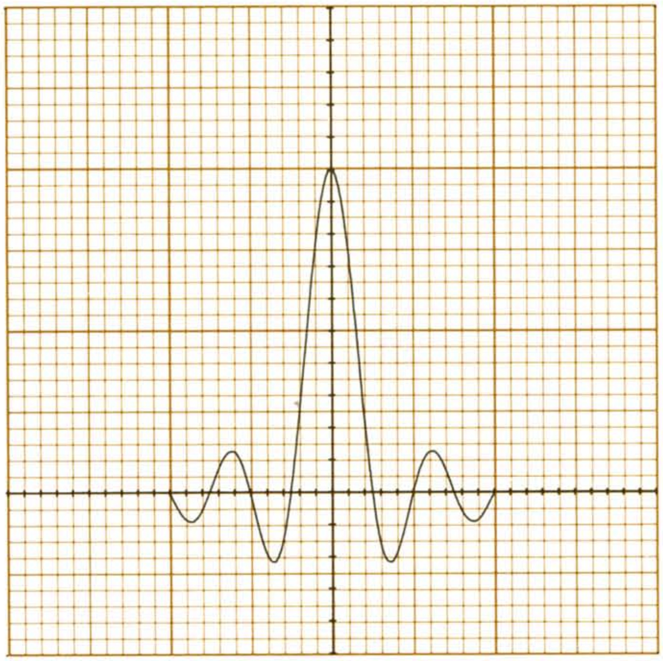


Figure 3-6. Plotting from -4π to $+4\pi$, while Scaling from -8π to $+8\pi$.

NOTES

Chapter 4

TYPEWRITER CONTROL

TABLE OF CONTENTS

CHAPTER 4. TYPEWRITER CONTROL

INTRODUCTION . . . 4-1
Select Code . . . 4-1
Typewriter Status Check . . . 4-1

TYPING DATA . . . 4-1
The TYPE DATA Syntax . . . 4-1
Using the Free-Field Format . . . 4-1
Example . . . 4-2
Using FORMAT Statements . . . 4-2
Examples . . . 4-2

TYPING MESSAGES . . . 4-3
Examples . . . 4-4
Programmed Examples . . . 4-5

ALTERNATE TYPEWRITER CONTROL . . . 4-6
The WRITE BYTE Syntax . . . 4-6
Examples . . . 4-6

Chapter 4

TYPEWRITER CONTROL

INTRODUCTION

The Model 9861A Typewriter is controlled by using WRITE and WRITE BYTE statements. WRITE statements can be used with or without FORMAT statements; the reader should be familiar with the use of FORMAT statements and free-field formatting, as described in Chapter 2.

If you suspect the performance of the typewriter or the typewriter control operations available with the PC II Block, see Supplement G of the Model 20 Electrical Inspection Booklet.

SELECT CODE

The typewriter is set to select code 15 at the factory; however, the typewriter can be set to respond to one of nine other select codes. See the Typewriter Peripheral Manual for instructions on how to set the select code.

TYPEWRITER STATUS CHECK

The READ STATUS statement can be used to check whether the typewriter is switched ON or OFF; see Page 2-13 for further information.

TYPING DATA

THE TYPE DATA SYNTAX:

WRT⟨select code⟩,⟨data item$_1$⟩,
i.e., ⟨data item$_2$⟩, . . .

WRT⟨select code⟩,⟨list⟩

Each ⟨data item⟩ parameter can consist of an integer, a register name, or an expression. A carriage-return line-feed (CR/LF) operation is given after all data items in the list are typed, unless a FORMAT statement indicates otherwise. Also, the typing format is in black unless a FORMAT statement is used to set the red ribbon (see 'Using FORMAT Statements' on the next page).

USING THE FREE-FIELD FORMAT

When the free-field format is used, the typing format consists of four columns of 18 spaces each. The four columns are adjacent to each other, and the left edge of the left-most column is set wherever the carriage is positioned when the TYPE DATA statement is executed; this location is usually over the left margin, because of a previous CR/LF.

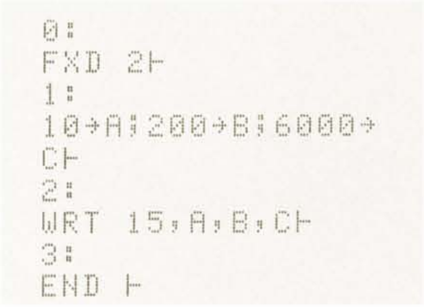
When the TYPE DATA statement is encountered, the first data item is typed (right-justified) in the left column; the next data item is typed in the next column; etc. After each group of four items are typed, a CR/LF is automatically given.

The typed appearance of each data item is controlled by the current settings established by the FIXED N and FLOAT N keys (N refers to the number of digits to be typed after the decimal point). In a fixed-point setting, no decimal point is typed if N is zero. Also, if a number is too large to be typed under the current fixed-point setting, the number is typed under the previous floating-point setting.

TYPING DATA

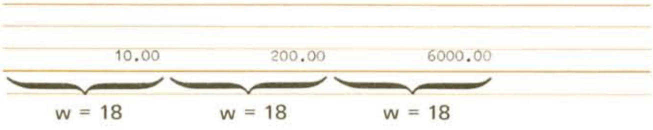
EXAMPLE:

Press ERASE, then load and run the following program.



```
0:
FXD 2⊢
1:
10→A;200→B;6000→
C⊢
2:
WRT 15,A,B,C⊢
3:
END ⊢
```

Result:

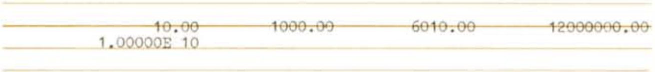


```
          10.00             200.00             6000.00
```

Now execute the line:

```
WRT 15,A,5B,A+C,ABC,A/1E-9⊢
```

Result:



```
            10.00           1000.00           6010.00       12000000.00
      1.00000E 10
```

Notice that a CR/LF is given after four data items are typed; then another CR/LF is given after the data-item list is exausted. Also, note that the typing format switches to 'floating point' when the data item is too large for the 'fixed-point' field.

USING FORMAT STATEMENTS

Typing with FORMAT statements allows more complete and flexible control of the typewriter than is otherwise possible. Typing is under the control of a FORMAT statement when a TYPE DATA statement is encountered and there has been a previously encountered FORMAT statement.

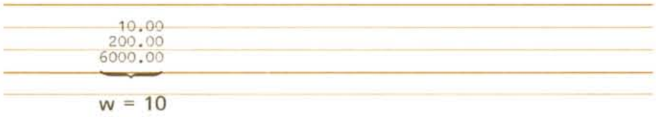
Under these conditions, a one-to-one correspondence is formed between the list of data items in the TYPE DATA statement and the list of specifications in the FORMAT statement. As the items are typed, each specification determines the format of the corresponding item.

EXAMPLE:

1. Execute the lines:

```
10→A;200→B;6000→C⊢
FMT FXD 10.2;WRT 15,A,B,C⊢
```

Result:



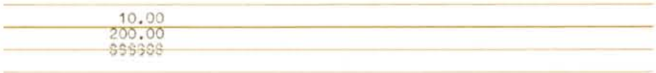
```
     10.00
    200.00
   6000.00
```

Since there was only one conversion specification, a CR/LF was given after each parameter was typed; thus, the use of one specification results in a one-column typeout.

2. Now execute the line:

```
FMT FXD 6.2;WRT 15,A,B,C⊢
```

Result:



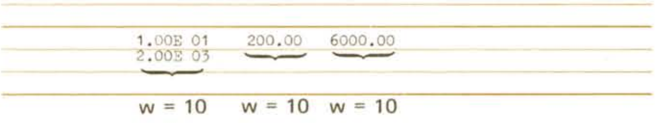
```
 10.00
200.00
$$$$$$
```

Since the field width specified was too small to allow parameter C to be typed, '$' characters were typed.

3. Now execute the line:

```
FMT FLT 10.2,2FXD 10.2;
            WRT 15,A,B,C,AB⊢
```

Result:



```
  1.00E 01    200.00   6000.00
  2.00E 03
```

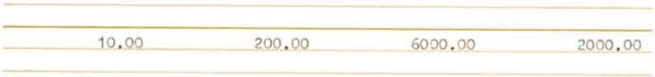
After each specification is used in the FORMAT statement, a CR/LF is given and the next data item is typed by using the first specification again. Thus, the sequence of instructions in a FORMAT statement determines the exact typing format

TYPING DATA

4. Now execute these lines from the keyboard:

```
FMT FLT 2.0⊢
WRT 15,A,B,C,AB⊢
```

Result:



```
      10.00           200.00          6000.00          2000.00
```

Display:

```
NOTE 22
```

Since the FORMAT statement is not included in the same line as the TYPE DATA statement, the data items are typed in the free-field format and NOTE 22 appears to remind the operator of the error.

Actually, even if both statements were executed in one line, the FLT 2.0 specification would have been out of range (see step 2), thus causing $ characters to be typed for each data item.

TYPING MESSAGES

Typewriter functions such as upper and lower case, red or black ribbon, tab activities, etc., can be controlled by using one or more keys as an editing specification in a FORMAT statement. As each key is encountered when executing the TYPE DATA statement, the specified typewriter operation is performed. Figure 4-1 shows the typewriter functions that can be controlled by using the corresponding key as an edit specification. The typewriter control keys available when the calculator keyboard is 'shifted' are colored light-brown.

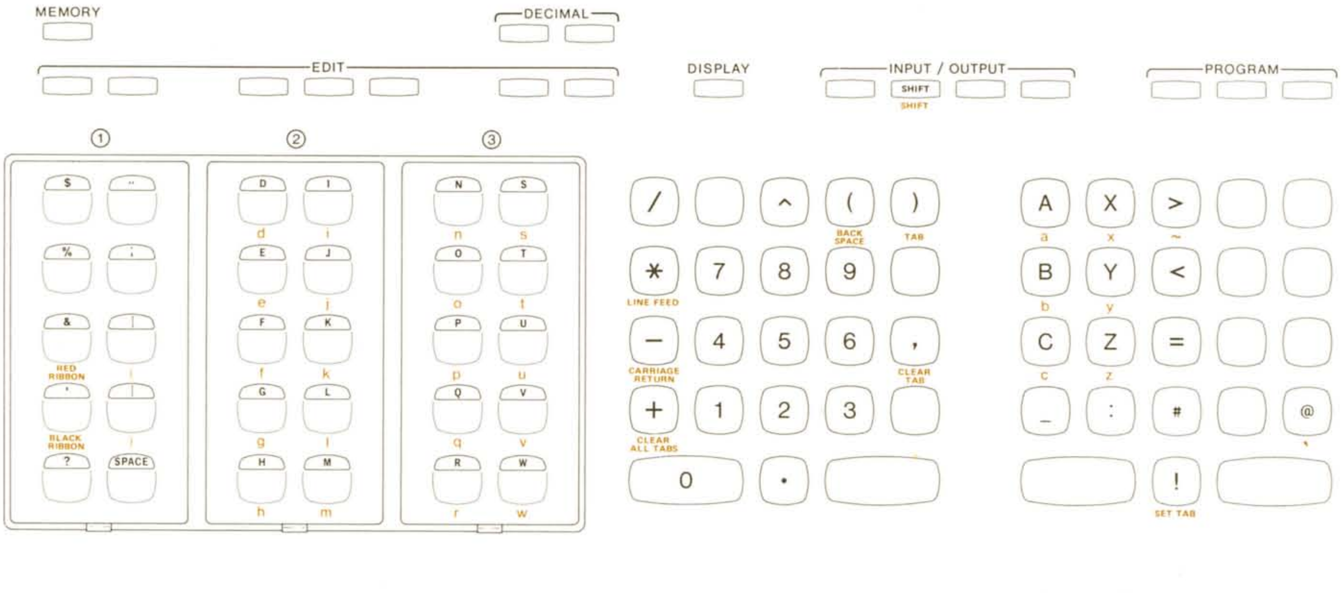


Figure 4-1. Typewriter Control Keys

TYPING MESSAGES

EXAMPLES:

The following are typical methods of controlling discrete typewriter operations (refer to the keyboard in Figure 4-1).

An alternate method of controlling the typewriter, by using the WRITE BYTE statement, is described on Page 4-6.

1. To **clear all tabs**, execute the line:

```
FMT "≛-",150X,"≛+";WRT 15⊢
   (shift)         (shift)
```

Result: edit spec$_1$ causes a carriage return

edit spec$_2$ causes 150 spaces

edit spec$_3$ causes a 'clear all tabs' operation (i.e., all tabs from the current carriage position to the left margin are cleared).

2. To **set a tab**, execute the line:

```
FMT 15X,"≛!",Z;WRT 15⊢
          ↖(set tab)
```

Result: edit spec$_1$ causes 15 spaces

edit spec$_2$ causes a tab to be set

edit spec$_3$ suppresses the automatic CR/LF

3. To **clear the second tab**, execute the line:

```
FMT 2"≛)","≛,",Z;WRT 15⊢
     (tab)    ↖(clear tab)
```

Result: edit spec$_1$ causes two 'tab' operations

edit spec$_2$ causes one tab to be cleared

edit spec$_3$ suppressed the automatic CR/LF

4. To **find the third tab**, execute the line:

```
FMT 3"≛)",Z;WRT 15⊢
     (tab)
```

Result: edit spec$_1$ causes three tab operations

edit spec$_2$ suppresses the automatic CR/LF

5. To **do five line feeds** without a carriage return, execute the line:

```
FMT 5"≛*√",Z;WRT 15⊢
        ↖(square root key)
```

Result: edit spec$_1$ causes 5 carriage return operations

edit spec$_2$ suppresses the automatic CR/LF

If the 'line feed' key is followed by another key, the specified number of line feed operations will not be performed. In this case, the SQUARE ROOT key was used since it does not program a typewriter operation.

6. To **do a carriage return** without a line feed, execute the line:

```
FMT "≛-",Z;WRT 15⊢
```

Result: edit spec$_1$ causes a carriage return operation

edit spec$_2$ suppresses the automatic CR/LF

7. To **do five back space** operations, execute the line:

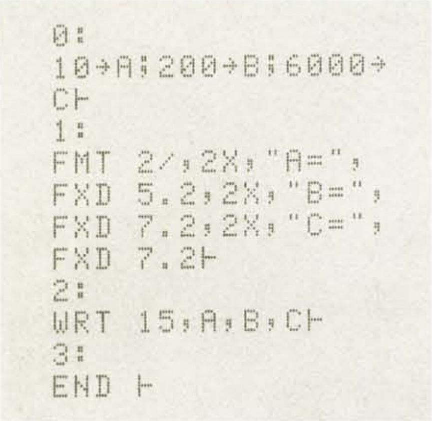
```
FMT 5"≛(",Z;WRT 15⊢
       ↖(back space)
```

Result: edit spec$_1$ causes 5 back space operations
edit spec$_2$ suppresses the automatic CR/LF

TYPING MESSAGES

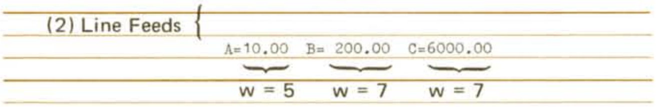
PROGRAMMED EXAMPLES:

1. Load and run the following program.



```
0:
10→A;200→B;6000→
C⊢
1:
FMT 2/,2X,"A=",
FXD 5.2,2X,"B=",
FXD 7.2,2X,"C=",
FXD 7.2⊢
2:
WRT 15,A,B,C⊢
3:
END ⊢
```

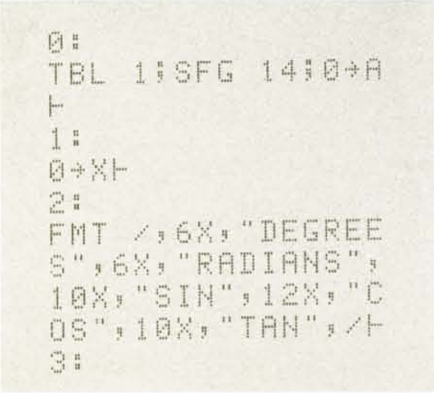
Result:



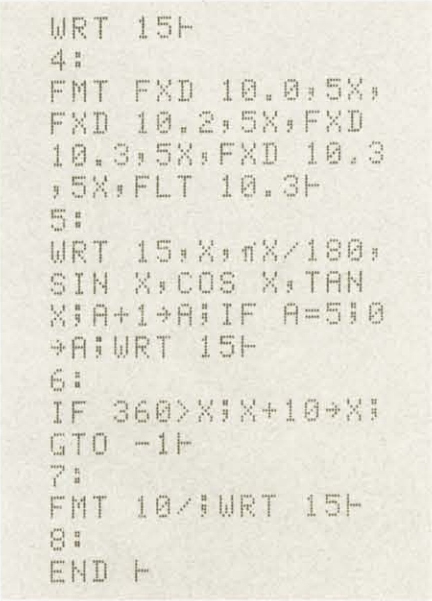
```
A=10.00  B= 200.00  C=6000.00
```

Notice the use of editing specifications to position the typewriter carriage and label the typed parameters.

2. Load and run the following program (this program requires the use of the Math Block).

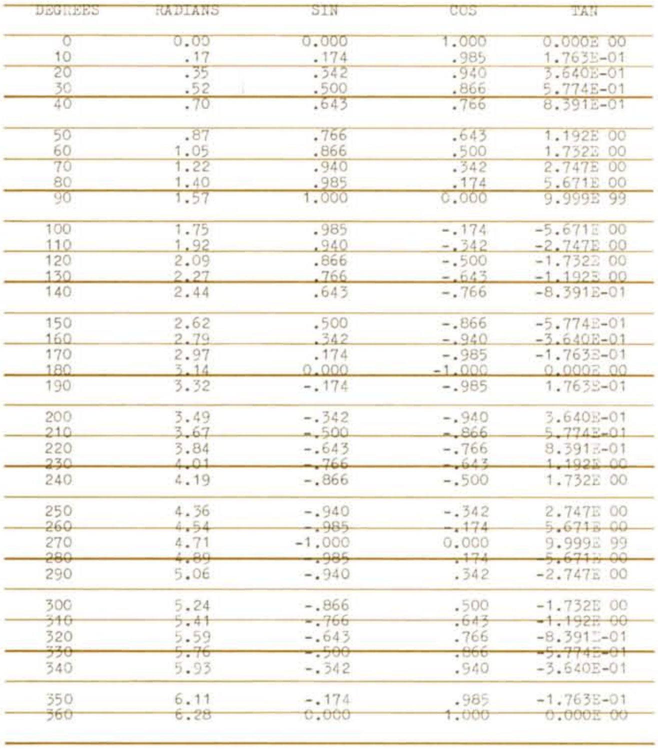


```
0:
TBL 1;SFG 14;0→A
⊢
1:
0→X⊢
2:
FMT /,6X,"DEGREE
S",6X,"RADIANS",
10X,"SIN",12X,"C
OS",10X,"TAN",/⊢
3:
```



```
WRT 15⊢
4:
FMT FXD 10.0,5X,
FXD 10.2,5X,FXD
10.3,5X,FXD 10.3
,5X,FLT 10.3⊢
5:
WRT 15,X,πX/180,
SIN X,COS X,TAN
X;A+1→A;IF A=5;0
→A;WRT 15⊢
6:
IF 360>X;X+10→X;
GTO -1⊢
7:
FMT 10/;WRT 15⊢
8:
END ⊢
```

Result:



```
DEGREES     RADIANS       SIN         COS          TAN

      0        0.00      0.000       1.000      0.000E 00
     10         .17       .174        .985      1.763E-01
     20         .35       .342        .940      3.640E-01
     30         .52       .500        .866      5.774E-01
     40         .70       .643        .766      8.391E-01

     50         .87       .766        .643      1.192E 00
     60        1.05       .866        .500      1.732E 00
     70        1.22       .940        .342      2.747E 00
     80        1.40       .985        .174      5.671E 00
     90        1.57      1.000       0.000      9.999E 99

    100        1.75       .985       -.174     -5.671E 00
    110        1.92       .940       -.342     -2.747E 00
    120        2.09       .866       -.500     -1.732E 00
    130        2.27       .766       -.643     -1.192E 00
    140        2.44       .643       -.766     -8.391E-01

    150        2.62       .500       -.866     -5.774E-01
    160        2.79       .342       -.940     -3.640E-01
    170        2.97       .174       -.985     -1.763E-01
    180        3.14      0.000      -1.000      0.000E 00
    190        3.32      -.174       -.985      1.763E-01

    200        3.49      -.342       -.940      3.640E-01
    210        3.67      -.500       -.866      5.774E-01
    220        3.84      -.643       -.766      8.391E-01
    230        4.01      -.766       -.643      1.192E 00
    240        4.19      -.866       -.500      1.732E 00

    250        4.36      -.940       -.342      2.747E 00
    260        4.54      -.985       -.174      5.671E 00
    270        4.71     -1.000       0.000      9.999E 99
    280        4.89      -.985        .174     -5.671E 00
    290        5.06      -.940        .342     -2.747E 00

    300        5.24      -.866        .500     -1.732E 00
    310        5.41      -.766        .643     -1.192E 00
    320        5.59      -.643        .766     -8.391E-01
    330        5.76      -.500        .866     -5.774E-01
    340        5.93      -.342        .940     -3.640E-01

    350        6.11      -.174        .985     -1.763E-01
    360        6.28      0.000       1.000      0.000E 00
```

Notice that lines 2 and 3 cause the headings to be typed; lines 4 and 5 cause each block of data items to be typed; and line 7 merely causes the carriage to CR/LF ten times before the program is completed.

ALTERNATE TYPEWRITER CONTROL

THE WRITE BYTE SYNTAX:

```
WTB⟨select code⟩ ,⟨typewriter control code⟩
```

The WRITE BYTE statement can be used to perform typewriter operations, one-by-one, by sending the appropriate control codes to the typewriter. Only one code may be included in each statement, and the code may be in the form of a positive integer, a variable (by specifying a register name), or an expression, See Table 4-1 ,for a list of typewriter control codes.

EXAMPLES:

Refer to Table 4-1 while reading these examples, which assume that the typewriter is set to select code 15.

1. To **clear a tab** (after positioning the carriage at the tab), execute the line:

```
WTB 15,12⊢
```

2. To **set a tab** after positioning the carriage, execute the line:

```
WTB 15,1⊢
```

3. To **do a carriage-return** without a line-feed operation, execute the line:

```
WTB 15,13⊢
```

Table 4-1. Typewriter Control Codes when using the WRITE BYTE Statement.

Typewriter Character	Code	Typewriter Character	Code	Typewriter Character	Code	Typewriter Character	Code
A	65	a	97	1	49	>	61
B	66	b	98	2	50	=	62
C	67	c	99	3	51	?	63
D	68	d	100	4	52	@	64
E	69	e	101	5	53	:	58
F	70	f	102	6	54	;	59
G	71	g	103	7	55	[	91
H	72	h	104	8	56	]	93
I	73	i	105	9	57	^	94
J	74	j	106	0	48	—	95
K	75	k	107	!	33	`	96
L	76	l	108	"	34	{	123
M	77	m	109	#	35	}	125
N	78	n	110	$	36	~	126
O	79	o	111	%	37		
P	80	p	112	&	38	**Typewriter Function**	**Code**
Q	81	q	113	'	39		
R	82	r	114	(	40	space	32
S	83	s	115	)	41	carriage return	13
T	84	t	116	*	42	line feed	10
U	85	u	117	+	43	back space	8
V	86	v	118	,	44	tab	9
W	87	w	119	-	45	tab set	1
X	88	x	120	.	46	tab clear	12
Y	89	y	121	/	47	set red ribbon	6
Z	90	z	122	<	60	set blk. riggon	7

Chapter 5

DIGITIZER CONTROL

TABLE OF CONTENTS

CHAPTER 5. DIGITIZER CONTROL

INTRODUCTION 5-1
Digitizer Select Code 5-1
Status Check 5-1

THE STATUS REQUEST STATEMENT 5-1
The Status Request Syntax 5-1
The STOP Key 5-1
Maximum Sample Rate 5-1
Example 5-1

THE 'BEEP' STATEMENT 5-2

DOCUMENT ALIGNMENT 5-2

Chapter 5

DIGITIZER CONTROL

INTRODUCTION

The Model 9864A Digitizer is controlled by using the READ and WRITE statements, which are described in Chapter 2. The instructions in this chapter assume that you are familiar with all general digitizer operations, as described in the Digitizer Peripheral Manual.

If you suspect the performance of the digitizer, refer to Supplement E of the Model 20 Electrical Inspection Booklet. This supplement is supplied with the PC II Block.

DIGITIZER SELECT CODE

The digitizer is set to select code 9 at the factory; for instructions on changing the select code setting, see the Digitizer Peripheral Manual.

STATUS CHECK

The READ STATUS statement can be used to check to see if the digitizer is switched ON or OFF. See Page 2-13 for instructions on how to check the status of the digitizer.

THE DATA REQUEST STATEMENT

THE DATA REQUEST SYNTAX:

RED⟨select code⟩ , ⟨register name$_x$⟩ , ⟨register name$_y$⟩

Where: X coordinate data → reg.$_x$
Y coordinate data → reg.$_y$

If the digitizer is in the continuous mode when the DATA REQUEST statement is encountered, a data sample is immediately accepted and stored in the specified registers. However, if the digitizer is not in the continuous mode, the calculator will wait until either C or S (on the cursor) is pressed before accepting a data sample.

THE STOP KEY

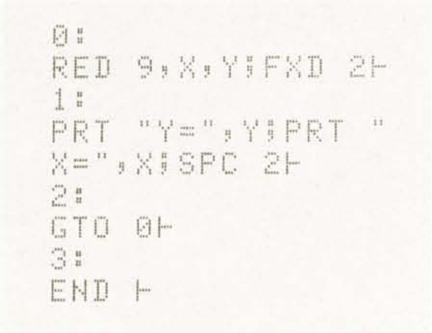
If STOP is pressed while the calculator is executing a DATA REQUEST statement; the operation will be terminated and the program will be halted. Also, pressing STOP while the digitizer is sending a data sample will deactivate the continuous mode.

MAXIMUM SAMPLE RATE

The maximum rate at which the calculator (when using PC II) can request and accept data samples from the digitizer is approximately 32 samples per second. Since the sample rate may be considerably slower due to program execution time, the operator must take care to move the cursor slowly; in order to obtain the maximum possible sample density. The effects of sample rate and sample density are discussed in the Digitizer Peripheral Manual.

EXAMPLE:

The following program instructs the calculator to continuously request and print data samples.



```
0:
RED 9,X,Y;FXD 2⊢
1:
PRT "Y=",Y;PRT "
X=",X;SPC 2⊢
2:
GTO 0⊢
3:
END ⊢
```

THE DATA REQUEST STATEMENT

To run the program:

1. Load the program into your calculator (be sure your digitizer's select code is specified correctly in line 0).
2. Attach the Sample Data Overlay supplied with your digitizer to the platen (see Figure 5-1).
3. Set the origin (press O on the cursor) approximately over point U on the overlay. Press END, RUN PROGRAM. The calculator is now waiting for a data sample from the digitizer.
4. To take continuous samples, press C . Slowly slide the cursor across the digitizing area (the corners of the digitizing area are indicated by the black dots on the platen).
5. To stop the sampling, but not halt the program, press C . Now press S several times; the digitizer supplies one data sample each time S is pressed.

   To stop the program, press STOP.

THE 'BEEP' STATEMENT

THE BEEP SYNTAX:

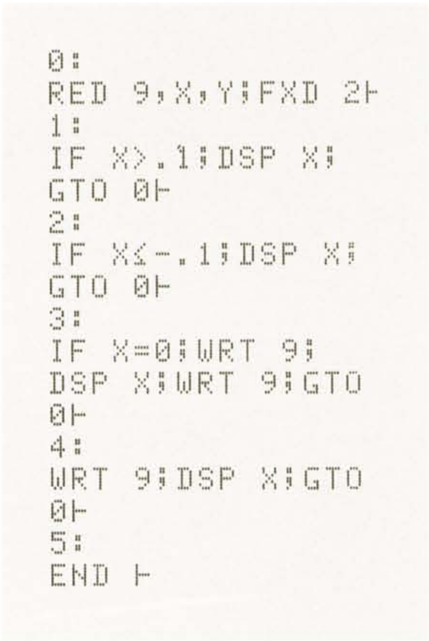
WRT⟨select code⟩

The BEEP statement causes the digitizer to sound an audible tone, which lasts about one-tenth of a second. A series of these statements, when separated by DISPLAY statements, will produce a pattern of 'beeps' which can be used to signal the operator during program operation.

A programmed use of the BEEP statement is shown below.

DOCUMENT ALIGNMENT

The following program can be used to align a document on the digitizing surface. The general procedure used to align a document is described in the Digitizer Peripheral Manual.



```
0:
RED 9,X,Y;FXD 2⊢
1:
IF X>.1;DSP X;
GTO 0⊢
2:
IF X≤-.1;DSP X;
GTO 0⊢
3:
IF X=0;WRT 9;
DSP X;WRT 9;GTO
0⊢
4:
WRT 9;DSP X;GTO
0⊢
5:
END ⊢
```

To run the program:

1. Load the program into your calculator (be sure your digitizer's select code is specified correctly in lines 0, 3, and 4).
2. Attach the Sample Data Overlay (or another document which is to be digitized) to the digitizing surface, as shown in Figure 5-1.
3. Place the cursor cross-hairs over point I (the upper left-hand corner of the document) and press O .
4. On the calculator: press END, RUN PROGRAM.
5. Slide the cursor over point III (the lower left-hand corner of the document) and position the cross-hairs **exactly** over point III; press C . (If point III is currently within ±.1 inch of the X axis as established over point I, the digitizer will be 'beeping' slowly. Whenever point III is positioned exactly over the X axis, the digitizer will 'beep' more rapidly.) Slowly move the cursor and the overlay (together) either right or left until the display (display equals .00) and the audible signal indicate alignment.

DOCUMENT ALIGNMENT

6. Without moving the overlay, tape the remaining three corners of the overlay (or document) to the platen. If necessary, retape the first corner.

7. Verify that the X axis of the document is in precise alignment with the platen by noting the display when the cross-hairs are positioned alternately over points I and III; return to step 3 if the X axis is not precisely aligned.

THE DIGITIZER PLATEN

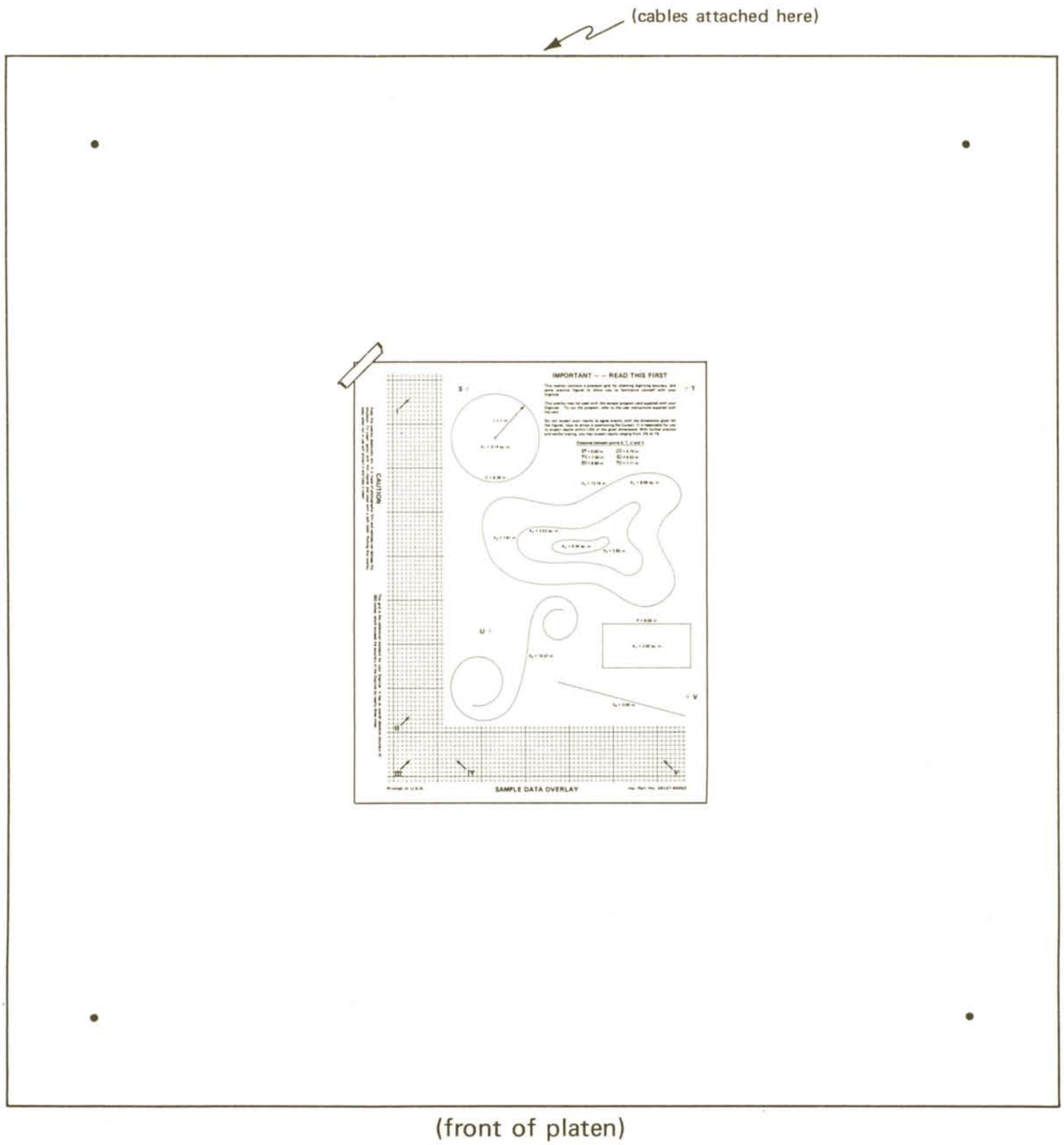


Figure 5-1. Document Alignment Procedure

NOTES

Chapter 6

TAPE READER CONTROL

6-0 TABLE OF CONTENTS

CHAPTER 6. TAPE READER CONTROL

INTRODUCTION 6-1
The READ DATA Syntax 6-1
Tape Reader Select Code 6-1
Status Check 6-1
The STOP Instruction 6-1

NORMAL MODE OPERATION 6-1
Data Entry Delimiters 6-2
Comma 6-2
Slash (/) and Line Feed (LF) 6-2
End of Message (EOM) 6-3
Enter Exponent 6-3

DATA MODE OPERATION 6-3
Selecting Delimiters 6-4
First and Second Delimiters 6-4
Examples 6-4

Chapter 6

TAPE READER CONTROL

INTRODUCTION

When using the PC II Block, the Model 9863A Tape Reader can input numerical data* to the calculator. The tape reader is controlled by using the READ statement, although the tape reader mode of operation determines the effect that data delimiters have on tape reader operation.

The instructions in this chapter assume that you are familiar with all tape reader controls, as described in the manual supplied with the tape reader (please disregard any reference to operation with the Model 10 Calculator).

THE READ DATA SYNTAX:

RED⟨select code⟩ , ⟨register name$_1$⟩ , ⟨register name$_2$⟩ , ...

i.e.,

RED⟨select code⟩ , ⟨list⟩

When a READ statement is executed, the first data item is stored in the first register in the ⟨list⟩; the second item is stored in the second register; etc. The tape reader halts when all registers in the list are filled, when an EOM character is read (see 'NORMAL MODE OPERATION'), or when the end of the tape is detected.

NOTE

If the end of the tape is detected while a READ statement is being executed, the program will be held at that statement (the display remains blank) until STOP, RUN PROGRAM is pressed.

TAPE READER SELECT CODE

When shipped from the factory, the tape reader is set to respond to select code 7. For instructions on how to set the select code to another number, see the manual supplied with the tape reader.

NOTE

Do not set the tape reader to respond to either select code 0 or more than one select code at the same time.

STATUS CHECK

The READ STATUS statement can be used to see if the tape reader is switched ON or OFF. See Page 2-8 for instructions on checking the tape reader status.

THE STOP INSTRUCTION

If STOP is pressed while the tape reader is reading, the operation will be immediately halted; however, FLAG 13 might be set and the current data item will probably not be correctly sent to the calculator.

* The data must only be in the form of ASCII codes unless the tape reader has an optional program board. All readable ASCII characters are listed in the manual supplied with the tape reader.

NORMAL MODE OPERATION

When the Mode switch is set to NORMAL, the tape reader is fully controlled by the calculator (i.e., the tape reader program board is deactivated). When in the normal mode, the tape reader accepts the following ASCII characters for numerical data entry: digits, decimal point, +, –, and 'E' (enter exponent).

NORMAL MODE OPERATION

DATA ENTRY DELIMITERS

Except for the ASCII characters to be described, all characters not accepted for data entry (see the preceding paragraph) are recognized as general data delimiters. The tape reader ignores all general delimiters which precede a data item. However, after reading data, reading a general delimiter causes the tape reader to send the previously read data to the calculator. Then, the calculator either resumes execution of the READ statement or, if the ⟨list⟩ is completed, begins execution of the next program statement.

Following is a description of special delimiters usable during normal mode operation.

COMMA

A comma which follows a data item is treated as a general delimiter. However, a comma following any general delimiter which follows a data item (i.e., a second delimiter) will cause FLAG 13 to be set. Also, a comma which immediately precedes a data item will cause FLAG 13 to be set.

EXAMPLE:

List of ASCII characters* on tape:

1ƀ200.5ƀ3ƀ4ƀ,C=5,A=6,

Load and run this program:

```
0:
FXD 2;0→A⊢
1:
RED 7,A⊢
2:
PRT A;SPC 2⊢
3:
END ⊢
```

Result:

```
        1.00
```

Running the program successive times will cause successive data items to be read, stored, and printed. Notice that since a general delimiter and a comma follow data item '4', running the program after '4' is read will cause FLAG 13 to be set but a data item will not be read, since the read operation will be stopped after the comma is recognized. Also, notice that all general delimiters which precede a data item are ignored.

EXAMPLE:

Modify and run the preceding program (see the following listing). Use the same tape as shown in the previous example.

```
0:
FXD 2;0→A→B→R2⊢
1:
RED 7,A,B,R2⊢
2:
PRT A,B,R2;SPC 2
⊢
3:
END ⊢
```

Result:

```
        1.00
      200.50
        3.00
```

Press RUN PROGRAM again.

Result:

```
        4.00
        0.00
        5.00
```

(FLAG 13 is set, but data it not read; thus, register B is left unchanged.)

Notice that a number of data items corresponding to the number of registers specified in the READ statement are read each time the statement is executed (except when delimiters are placed to prevent a data entry).

SLASH (/) AND LINE FEED(LF)

The tape reader ignores all data items which are enclosed in a field width which begins with a slash (/) and ends with a line feed (LF). These characters perform the same operation as the Begin Deletion and End Deletion operations described in 'DATA MODE OPERATION'.

* The ƀ symbol represents a space character.

NORMAL MODE OPERATION

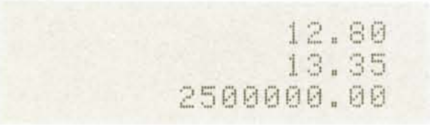
EXAMPLE:

List of ASCII characters on tape:

12.8ƀ13.35ƀ/200.5ƀ(LF)ƀ25E5ƀ

Run the program used in the previous example.

Result:



```
     12.80
     13.35
2500000.00
```

END OF MESSAGE (EOM)

The 'EOM' ASCII character causes the tape reader to halt and a 'RUN PROGRAM' instruction to be executed.

ENTER EXPONENT (E)

The 'E' ASCII character, when followed by any of the following forms, causes the preceding data item to be raised to the power of 10 indicated:

1. ⟨data item⟩E⟨one or two digits⟩
2. ⟨data item⟩E⟨a + or – and one or two digits⟩
3. ⟨data item⟩E⟨a general delimiter and one or two digits⟩

The following list of data items and resulting numeric entries show the effect of the form used to specify exponentiation.

EXAMPLE:

1. Each data item:

   E45
   1.0E45
   E+45
   Eƀ45

   Result:

   ```
   1.0E 45
   ```

2. The data items ('E' is omitted):

   123+4
   123–4

   Result:

   ```
   1.23E 06
   1.23E-02
   ```

3. Each data item.

   10E456
   10E45.6
   10Eƀ456

   Result:

   ```
   NOTE 11
   ```

In the last example, NOTE 11 appears because the exponent data consists of more than two digits or two digits and one other character.

DATA MODE OPERATION

When in the DATA mode, the tape reader responds only to the ASCII characters which are specified as delimiters on the program board and accepts the following ASCII characters for data entry: digits, decimal point, +, –, and 'E' (enter exponent).

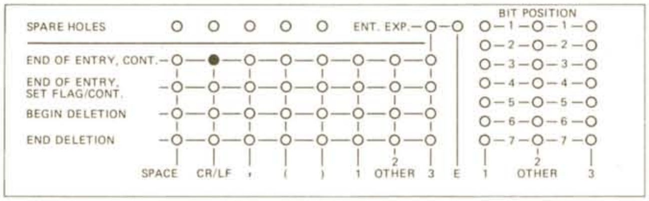
The following operations can be preset when the tape reader is in the data mode.

1. END OF ENTRY, CONTINUE: Upon reading a specified delimiter, the tape reader sends the previously read data item to the calculator. Then, the calculator either resumes execution of the READ statement or, if the ⟨list⟩ is complete, begins execution of the next program statement.
2. END OF ENTRY, SET FLAG, CONTINUE: The tape reader sends a 'SET FLAG 13' instruction to the calculator but does not make a data entry.
3. ENTER EXPONENT: Upon reading a specified delimiter, the following data (n) is entered into the calculator as 10 to the power n (10^n). For example, data written on the tape as 1.23E4 would be interpreted as 1.23E 04, and 123E4 would be interpreted as 1.23E 06.
4. BEGIN/END DELETION: Upon reading a specified delimiter, the tape reader will not transfer subsequent data to the calculator until the prespecified 'END DELETION' delimiter is read on the tape.

DATA MODE OPERATION

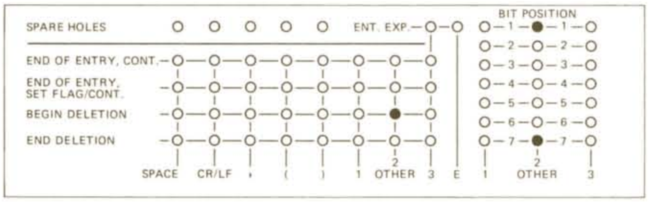
SELECTING DELIMITERS

In general, a pin inserted in one of the delimiter columns (space, CR/LF, comma, left bracket, or right bracket) will initiate the operation defined in the corresponding row when the delimiter is read. For example, if CR/LF is required to initiate END OF ENTRY, CONTINUE, the program board should be plugged as shown below.



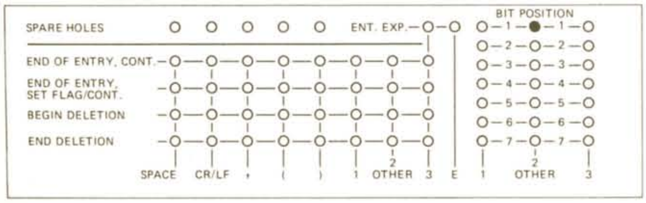
Note that only one pin can be inserted in any column (i.e., one delimiter may not initiate more than one operation).

The columns OTHER 1, OTHER 2, and OTHER 3 allow any ASCII character to be specified as a delimiter. A pin in one of these columns initiates an operation when the character set on the corresponding BIT POSITION column is read. For example, if the letter 'A' is required to initiate BEGIN DELETION (through the use of OTHER 2), pins shou!d be installed in the program board as shown below.

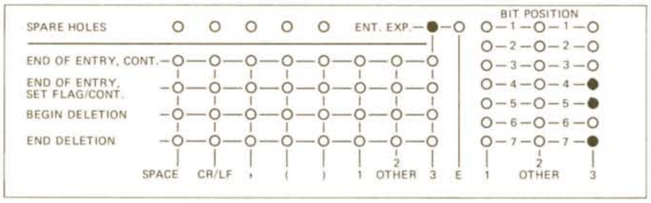


Notice that the program pins are inserted in a BIT POSITION hole when the ASCII code gives the corresponding bit as '1' (see the table of codes in the manual supplied with the tape reader).

If the ENTER EXPONENT operation is required, then either the letter 'E' can be specified as the delimiter (see the next example),

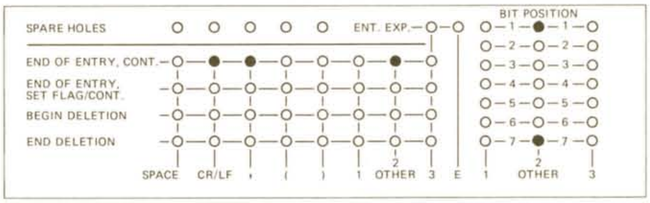


or a delimiter defined by the OTHER 3 position can be specified (as in the following example where 'X' is specified).



Note that the use of OTHER 3 to enter the exponent precludes its use for any other operation. Use of the letter 'E', however, permits OTHER 3 to initiate another operation.

More than one delimiter may be preset to initiate the same operation. For example, END OF ENTRY, CONTINUE will be initiated by CR/LF, or a comma, or the letter 'A' when the program board is set as shown below.



FIRST AND SECOND DELIMITERS

The tape reader always responds to the first delimiter which follows the data item. In addition, the tape reader will respond to the 'second' (different) delimiter which follows a data item.

The following examples show typical uses for a second delimiter.

EXAMPLES:

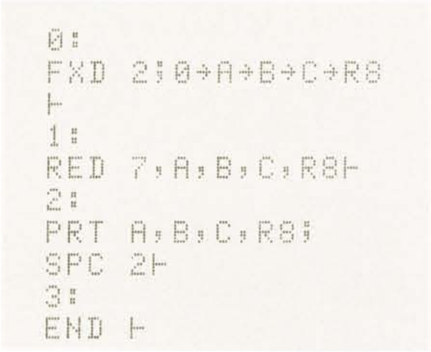
1. List of data items on tape:

   12.2,ƀ1005,,,A20.85,

   (second delimiter)

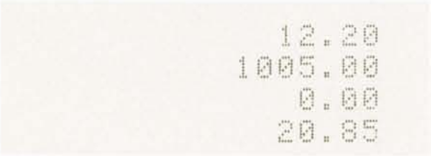
   Preset comma and CR/LF as 'END OF ENTRY, CONT' delimiters and preset the letter 'A' as 'END OF ENTRY, SET FLAG/CONT' delimiter.

Then load and run the program:



```
0:
FXD 2;0→A→B→C→R8
⊢
1:
RED 7,A,B,C,R8⊢
2:
PRT A,B,C,R8;
SPC 2⊢
3:
END ⊢
```

Result:



```
  12.20
1005.00
   0.00
  20.85
```

Notice that the tape reader responded to the second delimiter, even though the first delimiter was repeated.

2. List of data items on tape:

25.2,A CR LF ,,5E06,

(comma, CR/LF, and 'A' are still preset as delimiters)

Run the preceding program.

Result:

```
 25.20
  0.00
506.00
```

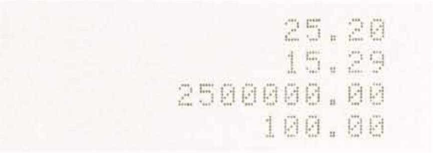
Notice that the third and subsequent delimiters in a series are ignored. Also, note that the 'E' character is ignored since it is not preset as a delimiter.

3. List of data items on tape:

B=25.2,C=(15.2(9,X=25E5,Y=100,

Preset the comma as the 'END OF ENTRY, CONT' delimiter and 'E' as 'ENTER EXPONENT' delimiter. Then run the preceding program.

Result:



```
     25.20
     15.29
2500000.00
    100.00
```

Notice that all characters not preset as delimiters are ignored.

NOTES

DIAGNOSTIC NOTES

Most of the execution and syntax errors associated with the PC II Block are similar in form to those of the basic calculator and cause the same 'notes' to appear. The PC II Block adds NOTE 20, NOTE 22, and NOTE 25. The most likely errors (not all possible errors) are listed below.

INDICATION	ERROR
NOTE 01	An I/O statement (except FORMAT or BUS COMMAND) contains a "literal".
NOTE 20	The select code specified is not within the range of from 1 to 15.
NOTE 22	(In general, the FORMAT statement contains a syntax error.) a. No conversion specification for a corresponding output parameter in a WRITE or TYPE statement. b. Conversion specification does not contain one of the following parameter types: FXD or FLT or " literal " c. Conversion specification does not contain field width (w) parameter or decimal point (d) parameter. d. A parameter is not followed by one of the following: , or ; or ⊢ e. A WRITE or TYPE statement was executed from the keyboard after a similar statement and a FORMAT statement were executed from the keyboard (see Keyboard Execution of Output Statements on Page 2-9).
NOTE 25	The Special Program called is not stored in memory.

Fold Out →

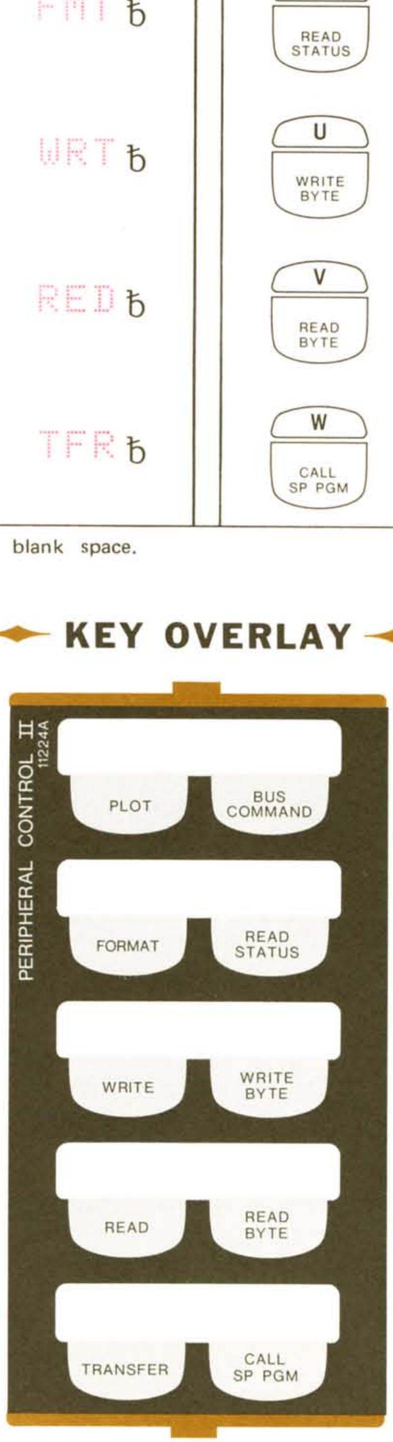
APPENDIX

KEY MNEMONICS

KEY	MNEMONIC☆	KEY	MNEMONIC☆
N PLOT	PLT ƀ	S BUS COMMAND	CMD ƀ
O FORMAT	FMT ƀ	T READ STATUS	RDS ƀ
P WRITE	WRT ƀ	U WRITE BYTE	WTB ƀ
Q READ	RED ƀ	V READ BYTE	RDB ƀ
R TRANSFER	TFR ƀ	W CALL SP PGM	CSP ƀ

☆ ƀ indicates a blank space.

KEY OVERLAY



ELECTRONIC

SALES & SERVICE OFFICES

UNITED STATES

ALABAMA
P.O. Box 4207
2003 Byrd Spring Road S.W.
Huntsville 35802
Tel: (205) 881-4591
TWX: 810-726-2204

ARIZONA
2336 E. Magnolia St.
Phoenix 85034
Tel: (602) 252-5061
TWX: 910-951-1330

5737 East Broadway
Tucson 85716
Tel: (602) 298-2313
TWX: 910-952-1162

CALIFORNIA
1430 East Orangethorpe Ave.
Fullerton 92631
Tel: (714) 870-1000

3939 Lankershim Boulevard
North Hollywood 91604
Tel: (213) 877-1282
TWX: 910-499-2170

1101 Embarcadero Road
Palo Alto 94303
Tel: (415) 327-6500
TWX: 910-373-1280

2220 Watt Ave.
Sacramento 95825
Tel: (916) 482-1463
TWX: 910-367-2092

9606 Aero Drive
San Diego 92123
Tel: (714) 279-3200
TWX: 910-335-2000

COLORADO
7965 East Prentice
Englewood 80110
Tel: (303) 771-3455
TWX: 910-935-0705

CONNECTICUT
12 Lunar Drive
New Haven 06525
Tel: (203) 389-6551
TWX: 710-465-2029

FLORIDA
P.O. Box 24210
2806 W. Oakland Park Blvd.
Ft. Lauderdale 33307
Tel: (305) 731-2020
TWX: 510-955-4099

P.O. Box 13910
6177 Lake Ellenor Dr.
Orlando, 32809
Tel: (305) 859-2900
TWX: 810-850-0113

GEORGIA
P.O. Box 28234
450 Interstate North
Atlanta 30328
Tel: (404) 436-6181
TWX: 810-766-4890

HAWAII
2875 So. King Street
Honolulu 96814
Tel: (808) 955-4455

ILLINOIS
5500 Howard Street
Skokie 60076
Tel: (312) 677-0400
TWX: 910-223-3613

INDIANA
3839 Meadows Drive
Indianapolis 46205
Tel: (317) 546-4891
TWX: 810-341-3263

LOUISIANA
P.O. Box 856
1942 Williams Boulevard
Kenner 70062
Tel: (504) 721-6201
TWX: 810-955-5524

MARYLAND
6707 Whitestone Road
Baltimore 21207
Tel: (301) 944-5400
TWX: 710-862-9157

P.O. Box 1648
2 Choke Cherry Road
Rockville 20850
Tel: (301) 948-6370
TWX: 710-828-9684

MASSACHUSETTS
32 Hartwell Ave.
Lexington 02173
Tel: (617) 861-8960
TWX: 710-326-6904

MICHIGAN
21840 West Nine Mile Road
Southfield 48075
Tel: (313) 353-9100
TWX: 810-224-4882

MINNESOTA
2459 University Avenue
St. Paul 55114
Tel: (612) 645-9461
TWX: 910-563-3734

MISSOURI
11131 Colorado Ave.
Kansas City 64137
Tel: (816) 763-8000
TWX: 910-771-2087

148 Weldon Parkway
Maryland Heights 63043
Tel: (314) 567-1455
TWX: 910-764-0830

NEW JERSEY
W. 120 Century Road
Paramus 07652
Tel: (201) 265-5000
TWX: 710-990-4951

1060 N. Kings Highway
Cherry Hill 08034
Tel: (609) 667-4000
TWX: 710-892-4945

NEW MEXICO
P.O. Box 8366
Station C
6501 Lomas Boulevard N.E.
Albuquerque 87108
Tel: (505) 265-3713
TWX: 910-989-1665

156 Wyatt Drive
Las Cruces 88001
Tel: (505) 526-2485
TWX: 910-983-0550

NEW YORK
6 Automation Lane
Computer Park
Albany 12205
Tel: (518) 458-1550
TWX: 710-441-8270

1219 Campville Road
Endicott 13760
Tel: (607) 754-0050
TWX: 510-252-0890

82 Washington Street
Poughkeepsie 12601
Tel: (914) 454-7330
TWX: 510-248-0012

39 Saginaw Drive
Rochester 14623
Tel: (716) 473-9500
TWX: 510-253-5981

5858 East Molloy Road
Syracuse 13211
Tel: (315) 454-2486
TWX: 710-541-0482

1 Crossways Park West
Woodbury 11797
Tel: (516) 921-0300
TWX: 510-223-0811

NORTH CAROLINA
P.O. Box 5188
1923 North Main Street
High Point 27262
Tel: (919) 885-8101
TWX: 510-926-1516

OHIO
25575 Center Ridge Road
Cleveland 44145
Tel: (216) 835-0300
TWX: 810-427-9129

3460 South Dixie Drive
Dayton 45439
Tel: (513) 298-0351
TWX: 810-459-1925

1120 Morse Road
Columbus 43229
Tel: (614) 846-1300

OKLAHOMA
6301 N. Meridian Avenue
Oklahoma City 73122
Tel: (405) 721-0200
TWX: 910-830-6862

OREGON
Westhills Mall, Suite 158
4475 S.W. Scholls Ferry Road
Portland 97225
Tel: (503) 292-9171
TWX: 910-464-6103

PENNSYLVANIA
2500 Moss Side Boulevard
Monroeville 15146
Tel: (412) 271-0724
TWX: 710-797-3650

1021 8th Avenue
King of Prussia Industrial Park
King of Prussia 19406
Tel: (215) 265-7000
TWX: 510-660-2670

RHODE ISLAND
873 Waterman Ave.
East Providence 02914
Tel: (401) 434-5535
TWX: 710-381-7573

***TENNESSEE**
Memphis
Tel: (901) 274-7472

TEXAS
P.O. Box 1270
201 E. Arapaho Rd.
Richardson 75080
Tel: (214) 231-6101
TWX: 910-867-4723

P.O. Box 27409
6300 Westpark Drive
Suite 100
Houston 77027
Tel: (713) 781-6000
TWX: 910-881-2645

231 Billy Mitchell Road
San Antonio 78226
Tel: (512) 434-4171
TWX: 910-871-1170

UTAH
2890 South Main Street
Salt Lake City 84115
Tel: (801) 487-0715
TWX: 910-925-5681

VERMONT
P.O. Box 2287
Kennedy Drive
South Burlington 05401
Tel: (802) 658-4455
TWX: 510-299-0025

VIRGINIA
P.O. Box 6514
2111 Spencer Road
Richmond 23230
Tel: (703) 285-3431
TWX: 710-956-0157

WASHINGTON
433-108th N.E.
Bellevue 98004
Tel: (206) 454-3971
TWX: 910-443-2303

***WEST VIRGINIA**
Charleston
Tel: (304) 768-1232

FOR U.S. AREAS NOT LISTED:
Contact the regional office nearest you: Atlanta, Georgia . . . North Hollywood, California . . . Paramus, New Jersey . . . Skokie, Illinois. Their complete addresses are listed above.

*Service Only

CANADA

ALBERTA
Hewlett-Packard (Canada) Ltd.
11748 Kingsway Ave.
Edmonton
Tel: (403) 452-3670
TWX: 610-831-2431

BRITISH COLUMBIA
Hewlett-Packard (Canada) Ltd.
4519 Canada Way
North Burnaby 2
Tel: (604) 433-8213
TWX: 610-922-5059

MANITOBA
Hewlett-Packard (Canada) Ltd.
513 Century St.
Winnipeg
Tel: (204) 786-7581
TWX: 610-671-3531

NOVA SCOTIA
Hewlett-Packard (Canada) Ltd.
2745 Dutch Village Rd.
Suite 206
Halifax
Tel: (902) 455-0511
TWX: 610-271-4482

ONTARIO
Hewlett-Packard (Canada) Ltd.
1785 Woodward Dr.
Ottawa 3
Tel: (613) 255-6180, 255-6530
TWX: 610-562-1952

Hewlett-Packard (Canada) Ltd.
50 Galaxy Blvd.
Rexdale
Tel: (416) 677-9611
TWX: 610-492-4246

QUEBEC
Hewlett-Packard (Canada) Ltd.
275 Hymus Boulevard
Pointe Claire
Tel: (514) 697-4232
TWX: 610-422-3022
Telex: 01-20607

FOR CANADIAN AREAS NOT LISTED:
Contact Hewlett-Packard (Canada) Ltd. in Pointe Claire, at the complete address listed above.

CENTRAL AND SOUTH AMERICA

ARGENTINA
Hewlett-Packard Argentina
S.A.C.e.I
Lavalle 1171 - 3°
Buenos Aires
Tel: 35-0436, 35-0627, 35-0431
Telex: 012-1009
Cable: HEWPACK ARG

BRAZIL
Hewlett-Packard Do Brasil
I.e.C Ltda.
Rua Frei Caneca 1119
Sao Paulo - 3, SP
Tel: 288-7111, 287-5858
Cable: HEWPACK Sao Paulo

Hewlett-Packard Do Brasil
Praca Dom Feliciano 78
Salas 806/808
Porto Alegre
Rio Grande do Sul (RS)-Brasil
Tel: 25-8470
Cable: HEWPACK Porto Alegre

Hewlett-Packard Do Brasil
I.e.C. Ltda.
Rua da Matriz 29
Botafogo ZC-02
Rio de Janeiro, GB
Tel: 246-4417, 246-2919
Cable: HEWPACK Rio de Janeiro

CHILE
Héctor Calcagni y Cia, Ltda.
Casilla 16.475
Santiago
Tel: 423 96
Cable: CALCAGNI Santiago

COLOMBIA
Instrumentacion
Henrik A. Langebaek & Kier S.A.
Carrera 7 No. 48-59
Apartado Aereo 6287
Bogota, 1 D.E.
Tel: 45-78-06, 45-55-46
Cable: AARIS Bogota
Telex: 44400INSTCO

COSTA RICA
Lic. Alfredo Gallegos Gurdián
Apartado 10159
San José
Tel: 21-86-13
Cable: GALGUR San José

ECUADOR
Laboratorios de Radio-Ingenieria
Calle Guayaquil 1246
Post Office Box 3199
Quito
Tel: 212-496; 219-185
Cable: HORVATH Quito

EL SALVADOR
Electronic Associates
Apartado Postal 1682
Centro Comercial Gigante
San Salvador, El Salvador C.A.
Paseo Escalon 4649-4° Piso
Tel: 23-44-60, 23-32-37
Cable: ELECAS

GUATEMALA
IPESA
5a via 2-01, Zona 4
Guatemala City
Tel: 63-6-27 & 64-7-86
Telex: 4176 Mahohegu

MEXICO
Hewlett-Packard Mexicana, S.A. de C.V.
622 Adolfo Prieto
Col. del Valle
Mexico 12, D.F.
Tel: 543-4232; 523-1874
Telex: 017-74-507

NICARAGUA
Roberto Terán G.
Apartado Postal 689
Edificio Terán
Managua
Tel: 3451, 3452
Cable: ROTERAN Managua

PANAMA
Electrónico Balboa, S.A.
P.O. Box 4929
Ave. Manuel Espinosa No. 13-50
Bldg. Alina
Panama City
Tel: 230833
Telex: 3481003, Curundu, Canal Zone
Cable: ELECTRON Panama City

PARAGUAY
Z.T. Melamed S.R.L.
Division: Aparatos y Equipos Medicos
Salon de Exposicion y Escritorio: Chile 482
Edificio Victoria—Planta Baja
Asuncion, Paraguay
Tel: 4-5069, 4-6272
Cable: RAMEL

PERU
Compañia Electro Medica S.A.
Ave. Enrique Canaual 312
San Isidro
Casilla 1030
Lima
Tel: 22-3900
Cable: ELMED Lima

PUERTO RICO
San Juan Electronics, Inc.
P.O. Box 5167
Ponce de Leon 154
Pda. 3-PTA de Tierra
San Juan 00906
Tel: (809) 725-3342, 722-3342
Cable: SATRONICS San Juan
Telex: SATRON 3450 332

URUGUAY
Pablo Ferrando S.A.
Comercial e Industrial
Avenida Italia 2877
Casilla de Correo 370
Montevideo
Tel: 40-3102
Cable: RADIUM Montevideo

VENEZUELA
Hewlett-Packard De Venezuela C.A.
Apartado 50933
Caracas
Tel: 71.88.05, 71.88.69, 71.99.30
Cable: HEWPACK Caracas
Telex: 21146 HEWPACK

FOR AREAS NOT LISTED,

CONTACT:
Hewlett-Packard
INTERCONTINENTAL
3200 Hillview Ave.
Palo Alto, California 94304
Tel: (415) 493-1501
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

EUROPE

AUSTRIA
Hewlett-Packard Ges.m.b.H
Handelska 52/3
P.O. Box 7
A-1205 **Vienna**
Tel: (0222) 33 66 06 to 09
Cable: HEWPAK Vienna
Telex: 75923 hewpak a

BELGIUM
Hewlett-Packard Benelux
S.A./N.V.
Avenue du Col-Vert, 1
B-1170 **Brussels**
Tel: (02) 72 22 40
Cable: PALOBEN Brussels
Telex: 23 494

DENMARK
Hewlett-Packard A/S
Datavej 38
DK-3460 **Birkerod**
Tel: (01) 81 66 40
Cable: HEWPACK AS
Telex: 16640 hp as

Hewlett-Packard A/S
Torvet 9
DK-8600 **Silkeborg**
Tel: (06)-82-71-66
Telex: 16640 hp as
Cable: HEWPACKAS

FINLAND
Hewlett-Packard Oy
Bulevardi 26
P.O. Box 12185
SF-00120 **Helsinki** 12
Tel: (90) 13730
Cable: HEWPACKOY-Helsinki
Telex: 17-1563 hel

FRANCE
Hewlett-Packard France
Quartier de Courtaboeuf
Boite Postale No. 6
F-91401 **Orsay**
Tel: (1) 907 78 25
Cable: HEWPACK Orsay
Telex: 60048

Hewlett-Packard France
4 Quai des Etroits
F-69321 **Lyon Cedex** 1
Tel: (78) 42 63 45
Cable: HEWPACK Lyon
Telex: 31617

Hewlett-Packard France
29 rue de la Gare
F-31700 **Blagnac**
Tel: (61) 85 82 29
Telex: 51957

GERMAN FEDERAL REPUBLIC
Hewlett-Packard Vertriebs-GmbH
Berliner Strasse 117
Postfach 560 140
D-6 Nieder-Eschbach/Ffm 56
Tel: (0611) 50-04-1
Cable: HEWPACKSA Frankfurt
Telex: 41 32 49 FRA

Hewlett-Packard Vertriebs-GmbH
Herrenbergerstrasse 110
D-7030 **Böblingen,** Württemberg
Tel: (07031) 66 72 87
Cable: HEPAK Böblingen
Telex: 72 65 739 bbn

Hewlett-Packard Vertriebs-GmbH
Vogelsanger Weg 38
D-4 **Düsseldorf**
Tel: (0211) 63 80 31/35
Telex: 85/86 533 hpdd d

Hewlett-Packard Vertriebs-GmbH
Wendenstr. 23
D-2 **Hamburg** 1
Tel: (0411) 24 05 51/52
Cable: HEWPACKSA Hamburg
Telex: 21 53 032 hphh d

Hewlett-Packard Vertriebs-GmbH
Unterhachinger Strasse 28
ISAR Center
D-8012 **Ottobrunn**
Tel: (0811) 601 30 61/7
Telex: 52 49 85
Cable: HEWPACKSA Müchen

(West Berlin)
Hewlett-Packard Vertriebs-GmbH
Wilmersdorfer Strasse 113/114
D-1000 **Berlin W.** 12
Tel: (0311) 3137046
Telex: 18 34 05 hpbln d

GREECE
Kostas Karayannis
18, Ermou Street
Athens 126
Tel: 3230-303
Cable: RAKAR Athens
Telex: 21 59 62 rkar gr

IRELAND
Hewlett-Packard Ltd.
224 Bath Road
Slough, SL1 4 DS, Bucks
Tel: Slough (0753) 33341
Cable: HEWPIE Slough
Telex: 84413

Hewlett-Packard Ltd.
The Graftons
Stamford New Road
Altrincham, Cheshire, England
Tel: (061) 928-8626
Telex: 668068

ITALY
Hewlett-Packard Italiana S.p.A.
Via Amerigo Vespucci 2
1-20124 **Milan**
Tel: (2) 6251 (10 lines)
Cable: HEWPACKIT Milan
Telex: 32046

Hewlett-Packard Italiana S.p.A.
Piazza Marconi
1-00144 **Rome** - Eur
Tel: (6) 5912544/5, 5915947
Cable: HEWPACKIT Rome
Telex: 61514

Hewlett-Packard Italiana S.p.A.
Vicolo Pastori, 3
I-35100 **Padova**
Tel: (49) 66 40 62
Telex: 32046 via Milan

Hewlett-Packard Italiana S.p.A.
Via Colli, 24
I-10129 **Turin**
Tel: (11) 53 82 64
Telex: 32046 via Milan

LUXEMBURG
Hewlett-Packard Benelux
S.A./N.V.
Avenue du Col-Vert, 1
B-1170 **Brussels**
Tel: (03/02) 72 22 40
Cable: PALOBEN Brussels
Telex: 23 494

NETHERLANDS
Hewlett-Packard Benelux, N.V.
Weerdestein 117
P.O. Box 7825
Amsterdam, Z 11
Tel: 020-42 77 77
Cable: PALOBEN Amsterdam
Telex: 13 216 hepa nl

NORWAY
Hewlett-Packard Norge A/S
Box 149
Nesveien 13
N-1344 **Haslum**
Tel: (02)-53 83 60
Telex: 16621 hpnas n

PORTUGAL
Telectra-Empresa Tecnica de
Equipamentos
Electricos S.a.r.l.
Rua Rodrigo da Fonseca 103
P.O. Box 2531
P-**Lisbon** 1
Tel: (19) 68 60 72
Cable: TELECTRA Lisbon
Telex: 1598

SPAIN
Hewlett-Packard Española, S.A.
Jerez No 8
Madrid 16
Tel: 458 26 00
Telex: 23515 hpe

Hewlett-Packard Española, S.A
Milanesado 21-23
E-**Barcelona** 17
Tel: (3) 203 62 00
Telex: 52603 hpbe e

SWEDEN
Hewlett-Packard Sverige AB
Enighetsvägen 1-3
Fack
S-161 20 **Bromma** 20
Tel: (08) 98 12 50
Cable: MEASUREMENTS
Stockholm
Telex: 10721

Hewlett-Packard Sverige AB
Hagakersgatan 9C
S-431 41 **Mölndal**
Tel: (031) 27 68 00/01
Telex: 21 312 hpmindl s

SWITZERLAND
Hewlett Packard (Schweiz) AG
Zürcherstrasse 20
P.O. Box 64
CH-8952 **Schlieren** Zurich
Tel: (01) 98 18 21/24
Cable: HPAG CH
Telex: 53933 hpag ch

Hewlett-Packard (Schweiz) AG
Rue du Bois-du-Lan 7
P.O. Box 85
1217 Meyrin 2 **Geneva**
Tel: (022) 41 54 00
Cable: HEWPACKSA Geneva
Telex: 27333 hpsa ch

TURKEY
Telekom Engineering Bureau
Saglik Sok No. 15/1
Ayaspasa-Beyoglu
P.O. Box 437 Beyoglu
Istanbul
Tel: 49 40 40
Cable: TELEMATION Istanbul

UNITED KINGDOM
Hewlett-Packard Ltd.
224 Bath Road
Slough, SL1 4 DS, Bucks
Tel: Slough (0753) 33341
Cable: HEWPIE Slough
Telex: 84413

Hewlett-Packard Ltd.
"The Graftons"
Stamford New Road
Altrincham, Cheshire
Tel: (061) 928-8626
Telex: 668068

SOCIALIST COUNTRIES PLEASE CONTACT:
Hewlett-Packard Ges.m.b.H.
Handelskai 52/3
P.O. Box 7
A-1205 **Vienna**
Ph: (0222) 33 66 06 to 09
Cable: HEWPACK Vienna
Telex: 75923 hewpak a

ALL OTHER EUROPEAN COUNTRIES CONTACT:
Hewlett-Packard S.A.
Rue du Bois-du-Lan 7
P.O. Box 85
CH-1217 Meyrin 2 **Geneva**
Switzerland
Tel: (022) 41 54 00
Cable: HEWPACKSA Geneva
Telex: 2.24.86

AFRICA, ASIA, AUSTRALIA

ANGOLA
Telectra Empresa Técnia
de Equipamentos Eléctricos
SAR
Rua de Barbosa Rodrigues
42-1°
Box 6487
Luanda
Cable: TELECTRA Luanda

AUSTRALIA
Hewlett-Packard Australia
Pty. Ltd.
22-26 Weir Street
Glen Iris, 3146
Victoria
Tel: 20-1371 (6 lines)
Cable: HEWPARD Melbourne
Telex: 31 024

Hewlett-Packard Australia
Pty. Ltd.
Corner Bridge & West Streets
Pymble, New South Wales, 2073
Tel: 449 6566
Cable: HEWPARD Sydney
Telex: 21561

Hewlett-Packard Australia
Pty. Ltd.
97 Churchill Road
Prospect 5082
South Australia
Tel: 65-2366
Cable: HEWPARD Adelaide

Hewlett Packard Australia
Pty. Ltd.
2nd Floor, Suite 13
Casablanca Buildings
196 Adelaide Terrace
Perth, W.A. 6000
Tel: 25 6800
Cable: HEWPARD Perth

Hewlett-Packard Australia
Pty. Ltd.
10 Woolley Street
P.O. Box 191
Dickson A.C.T. 2602
Tel: 49-8194
Cable: HEWPARD Canberra ACT

Hewlett-Packard Australia
Pty. Ltd.
6 Harvard Street
P.O. Box 135
Kenmore 4069 Queensland
Tel: 70-4050

CEYLON
United Electricals Ltd.
P.O. Box 681
Yahala Building
Staples Street
Colombo 2
Tel: 26696
Cable: HOTPOINT Colombo

CYPRUS
Kypronics
19 Gregorios & Xenopoulos Road
P.O. Box 1152
Nicosia
Tel: 45628/29
Cable: Kypronks Pandehis

ETHIOPIA
African Salespower & Agency
Private Ltd., Co.
P. O. Box 718
58/59 Cunningham St.
Addis Ababa
Tel: 12285
Cable: ASACO Addisababa

HONG KONG
Schmidt & Co. (Hong Kong) Ltd.
P.O. Box 297
1511, Prince's Building 15th Floor
10, Chater Road
Hong Kong
Tel: 240168, 232735
Cable: SCHMIDTCO Hong Kong

INDIA
Blue Star Ltd.
Kasturi Buildings
Jamshedji Tata Rd.
Bombay 20BR, India
Tel: 29 50 21
Telex: 3751
Cable: BLUEFROST

Blue Star Ltd.
Band Box House
Prabhadevi
Bombay 25DD, India
Tel: 45 73 01
Telex: 3751
Cable: BLUESTAR

Blue Star Ltd.
14/40 Civil Lines
Kanpur, India
Tel: 6 88 82
Cable: BLUESTAR

Blue Star, Ltd.
7 Hare Street
P.O. Box 506
Calcutta 1, India
Tel: 23-0131
Telex: 655
Cable: BLUESTAR

Blue Star Ltd.
Blue Star House,
34 Ring Road
Lajpat Nagar
New Delhi 24, India
Tel: 62 32 76
Telex: 463
Cable: BLUESTAR

Blue Star, Ltd.
Blue Star House
11/11A Magarath Road
Bangalore, 25
Tel: 51473
Telex: 430
Cable: BLUESTAR

Blue Star, Ltd.
1-1-117/1
Sarojini Devi Road
Secunderabad 3
Tel: 7 63 91, 7 73 93
Cable: BLUEFROST

Blue Star, Ltd.
23/24 Second Line Beach
Madras 1, India
Tel: 2 39 55
Telex: 379
Cable: BLUESTAR

Blue Star, Ltd.
1B Kaiser Bungalow
Dindli Road
Jamshedpur, India
Tel: 38 04
Cable: BLUESTAR

INDONESIA
Bah Bolon Trading Coy. N.V.
Djalah Merdeka 29
Bandung
Tel: 4915; 51560
Cable: ILMU
Telex: 08-809

IRAN
Multicorp International Ltd.
Avenue Soraya 130
P.O. Box 1212
Teheran
Tel: 83 10 35-39
Cable: MULTICORP Tehran
Telex: 2893 mci tn

ISRAEL
Electronics & Engineering
Div. of Motorola Israel Ltd.
17 Aminadav Street
Tel-Aviv
Tel: 36941 (3 lines)
Cable: BASTEL Tel-Aviv
Telex: MOTIL IL

JAPAN
Yokogawa-Hewlett-Packard Ltd.
Ohashi Building
1-59-1 Yoyogi
Shibuya-ku, **Tokyo**
Tel: 03-370-2281/92
Telex: 232-2024YHP
Cable: YHPMARKET TOK 23-724

Yokogawa-Hewlett-Packard Ltd.
Nisei Ibaragi Bldg.
2-2-8 Kasuga
Ibaragi-Shi
Osaka
Tel: (0726) 23-1641
Telex: 5332-385 YHP OSAKA

Yokogawa-Hewlett-Packard Ltd.
Ito Building
No. 59, Kotori-cho
Nakamura-ku, **Nagoya** City
Tel: (052) 551-0215

Yokogawa-Hewlett-Packard Ltd.
Nitto Bldg.
2-4-2 Shinohara-Kita
Kohoku-ku
Yokohama 222
Tel: 045-432-1504
Telex: 382-3204 YHP YOK

Yokogawa-Hewlett-Packard Ltd.
Chuo Bldg.
Rm. 603 3,
2-Chome
IZUMI-CHO,
Mito, 310
Tel: 0292-25-7470

KENYA
Kenya Kinetics
P.O. Box 18311
Nairobi, Kenya
Tel: 57726
Cable: PROTON

KOREA
Amtraco Corporation
Industrial Products Div.
Seoul P.O. Box 1103
8th floor, DaeKyung Bldg.
107 Sejong Ro
Chongro-Ku, Seoul
Tel: 73-8924-7
Cable: AMTRACO Seoul

LEBANON
Constantin E. Macridis
P.O. Box 7213
RI-Beirut
Tel: 220846
Cable: ELECTRONUCLEAR Beirut

MALAYSIA
MECOMB Malaysia Ltd.
2 Lorong 13/6A
Section 13
Petaling Jaya, **Selangor**
Cable: MECOMB Kuala Lumpur

MOZAMBIQUE
A. N. Goncalves, LDA.
4.1 Apt. 14 Av. D. Luis
P.O. Box 107
Lourenco Marques
Cable: NEGON

NEW ZEALAND
Hewlett-Packard (N.Z.) Ltd.
94-96 Dixson St.
P.O. Box 9443
Courtenay Place
Wellington, N.Z.
Tel: 56-559
Cable: HEWPACK Wellington

Hewlett Packard (N.Z.) Ltd.
Box 51092
Pukuranga
Tel: 56-9837
Cable: HEWPACK, Auckland

PAKISTAN
Mushko & Company, Ltd.
Oosman Chambers
Abdullah Haroon Road
Karachi 3
Tel: 511027, 512927
Cable: COOPERATOR Karachi

Mushko & Company, Ltd.
38B, Satellite Town
Rawalpindi
Tel: 41924
Cable: FEMUS Rawalpindi

PHILIPPINES
Electromex Inc.
5th Floor, Architects
Center Bldg.
Ayala Ave., Makati, Rizal
C.C.P.O. Box 1028
Makati, Rizal
Tel: 86-18-87, 87-76-77
Cable: ELEMEX Manila

SINGAPORE
Mechanical and Combustion
Engineering Company Ltd.
9, Jalan Kilang
Red Hill Industrial Estate
Singapore, 3
Tel: 642361-3; 632611
Cable: MECOMB Singapore

Hewlett-Packard Far East
Area Office
P.O. Box 87
Alexandra Post Office
Singapore 3
Tel: 633022
Cable: HEWPACK SINGAPORE

SOUTH AFRICA
Hewlett Packard South Africa
(Pty.), Ltd.
P.O. Box 31716
Braamfontein Transvaal
Milnerton
30 De Beer Street
Johannesburg
Tel: 725-2080, 725-2030
Telex: 0226 JH
Cable: HEWPACK Johannesburg

Hewlett Packard South Africa
(Pty.), Ltd.
Breecastle House
Bree Street
Cape Town
Tel: 3-6019, 3-6545
Cable: HEWPACK Cape Town
Telex: 5-0006

Hewlett Packard South Africa
(Pty.), Ltd.
641 Ridge Road, Durban
P.O. Box 99
Overport, Natal
Tel: 88-6102
Telex: 567954
Cable: HEWPACK

TAIWAN
Hewlett Packard Taiwan
39 Chung Shiao West Road
Sec. 1
Overseas Insurance
Corp. Bldg. 7th Floor
Taipei
Tel: 389160,1,2, 375121,
Ext. 240-249
Telex: TP824 HEWPACK
Cable: HEWPACK Taipei

THAILAND
UNIMESA Co., Ltd.
Chongkoinee Building
56 Suriwongse Road
Bangkok
Tel: 37956, 31300, 31307,
37540
Cable: UNIMESA Bangkok

UGANDA
Uganda Tele-Electric Co., Ltd.
P.O. Box 4449
Kampala
Tel: 57279
Cable: COMCO Kampala

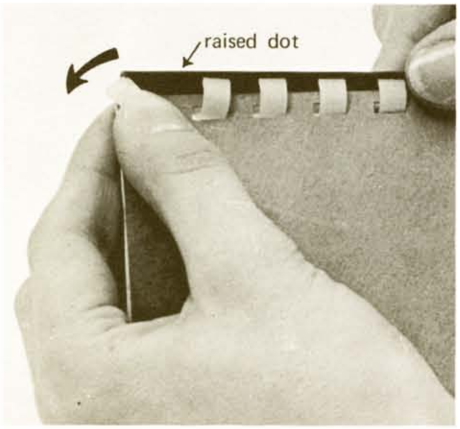
VIETNAM
Peninsular Trading Inc.
P.O. Box H-3
216 Hien-Vuong
Saigon
Tel: 20-805, 93398
Cable: PENTRA, SAIGON 242

ZAMBIA
R. J. Tilbury (Zambia) Ltd.
P.O. Box 2792
Lusaka
Zambia, Central Africa
Tel: 73793
Cable: ARJAYTEE, Lusaka

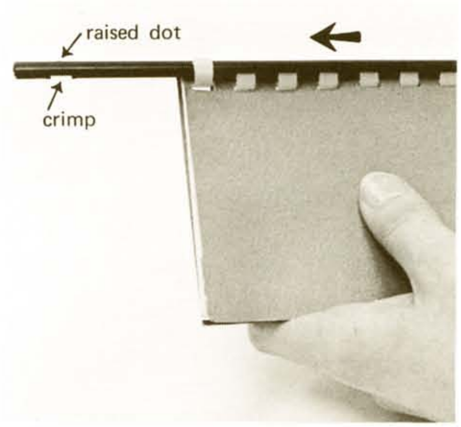
MEDITERRANEAN AND MIDDLE EAST COUNTRIES NOT SHOWN PLEASE CONTACT:
Hewlett-Packard
Co-ordination Office for
Mediterranean and Middle
East Operations
Via Marocco, 7
I-00144 Rome-Eur, Italy
Tel: (6) 59 40 29
Cable: HEWPACKIT Rome
Telex: 61514

OTHER AREAS NOT LISTED, CONTACT:
Hewlett-Packard
INTERCONTINENTAL
3200 Hillview Ave.
Palo Alto, California 94304
Tel: (415) 326-7000
(Feb. 71 493-1501)
TWX: 910-373-1267
Cable: HEWPACK Palo Alto
Telex: 034-8300, 034-8493

DISASSEMBLING THE BINDING

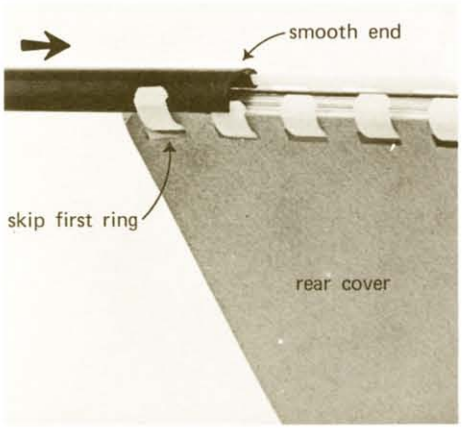


1 Unhook the end ring.

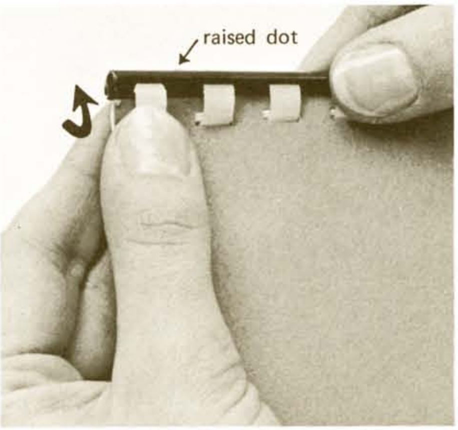


2 Slide the binding apart.

ASSEMBLING THE BINDING



3 Engage the binding, but skip the first ring.



4 Snap the ring into the groove.

Part No. 09820–90024
Microfiche No. 09820–99024

Printed in U.S.A.
Sept 1972