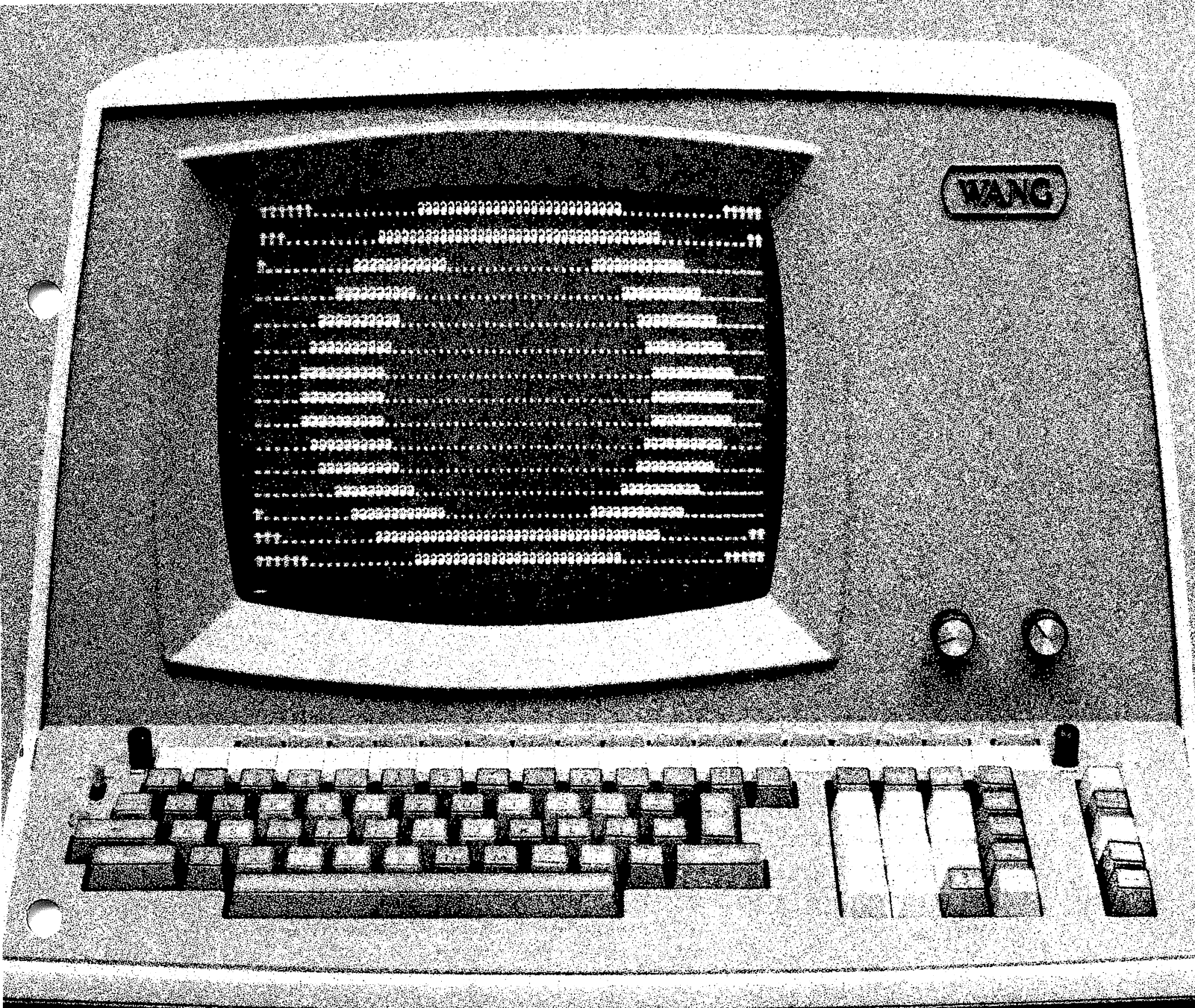
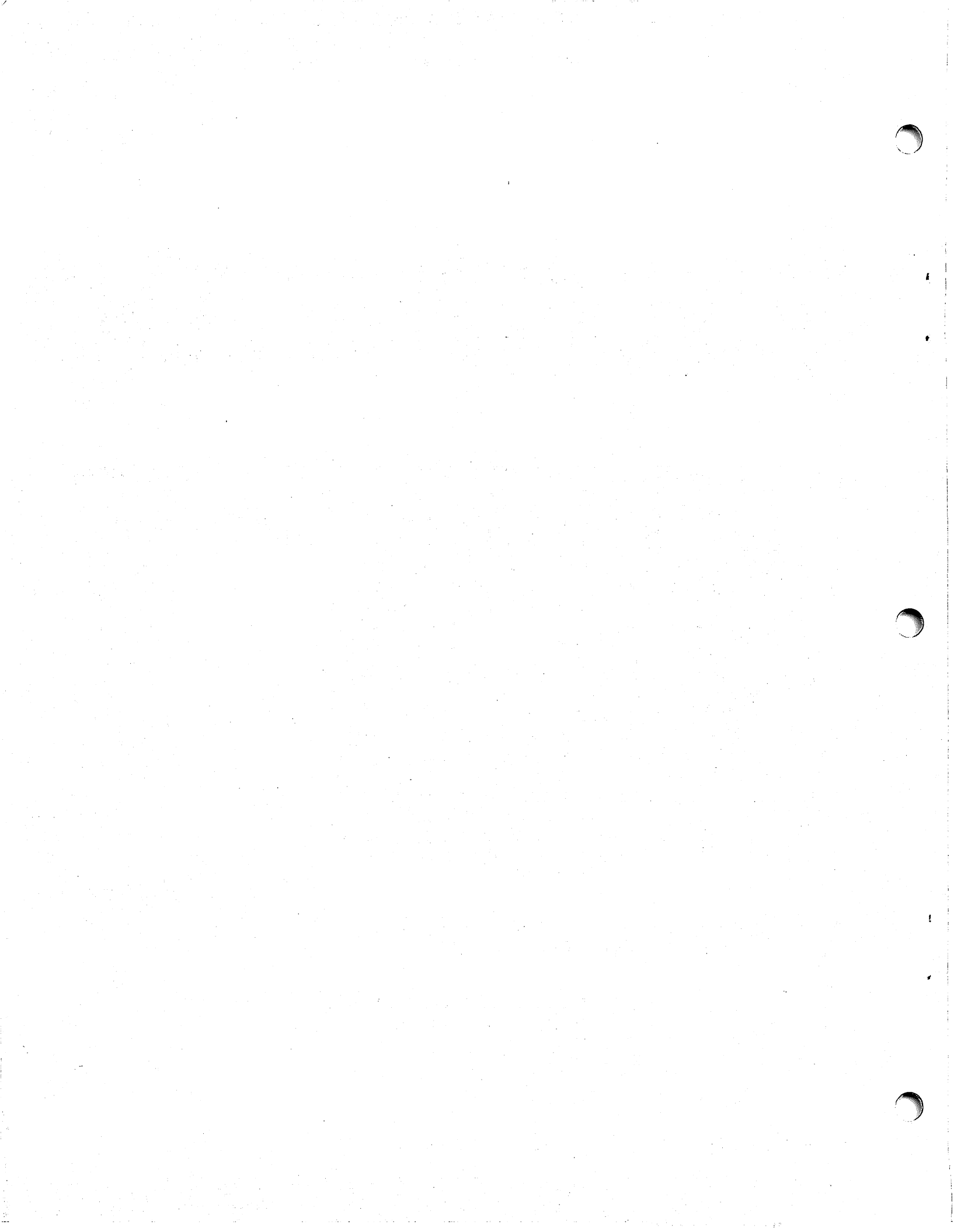


WANG

PORTABLE COMPUTING
SYSTEM
INTRODUCTION

SYSTEM 2200





PORTABLE COMPUTING SYSTEM INTRODUCTION

© Wang Laboratories, Inc., 1976



LABORATORIES, INC.

836 NORTH STREET, TEWKSBURY, MASSACHUSETTS 01876. TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

Disclaimer of Warranties and Limitation of Liabilities

The staff of Wang Laboratories, Inc., has taken due care in preparing this manual; however, nothing contained herein modifies or alters in any way the standard terms and conditions of the Wang purchase agreement, lease agreement, or rental agreement by which this equipment was acquired, nor increases in any way Wang's liability to the customer. In no event shall Wang Laboratories, Inc., or its subsidiaries be liable for incidental or consequential damages in connection with or arising from the use of this manual or any programs contained herein.

WANG

LABORATORIES, INC.

836 NORTH STREET, TEWKSBURY, MASSACHUSETTS 01876, TEL. (617) 861-4111, TWX 710 343-8789, TELEX 94-7421

HOW TO USE THIS MANUAL

This manual introduces the new user to the Wang 2200 Portable Computing System and to the accompanying 2200 system documentation. It is the sole reference documentation of the PCS's physical characteristics, but since the PCS shares its operational characteristics with other Wang 2200 systems it provides only an introductory level knowledge of PCS use and operation.

Chapter 1 describes the PCS in general terms, and provides some fundamental "do's and don'ts" about unpacking it and setting it up. Chapter 2 describes the physical components that make up the PCS. Chapter 3 is a guide to performing some simple operations with the PCS: how to load a program from cassette and initiate its execution, how to perform quick calculations without the use of a program, how to edit a line. These discussions are provided for the occasional user, who may have neither the time nor the need to become more familiar with the PCS, and also as an introduction for the new user, who may wish to eventually become a proficient PCS programmer.

Chapter 4 summarizes the PCS's Extended BASIC language and its system commands. The summary should be most valuable to the PCS user who has some familiarity with BASIC or with the fundamental concepts of programming, and who wants a quick introduction to the PCS's language and programming capabilities. In addition, a brief description is included of the procedure for entering programs via the keyboard. Beginners should refer to the Wang BASIC Language Programming Manual for a fuller, more systematic introduction to programming the PCS.

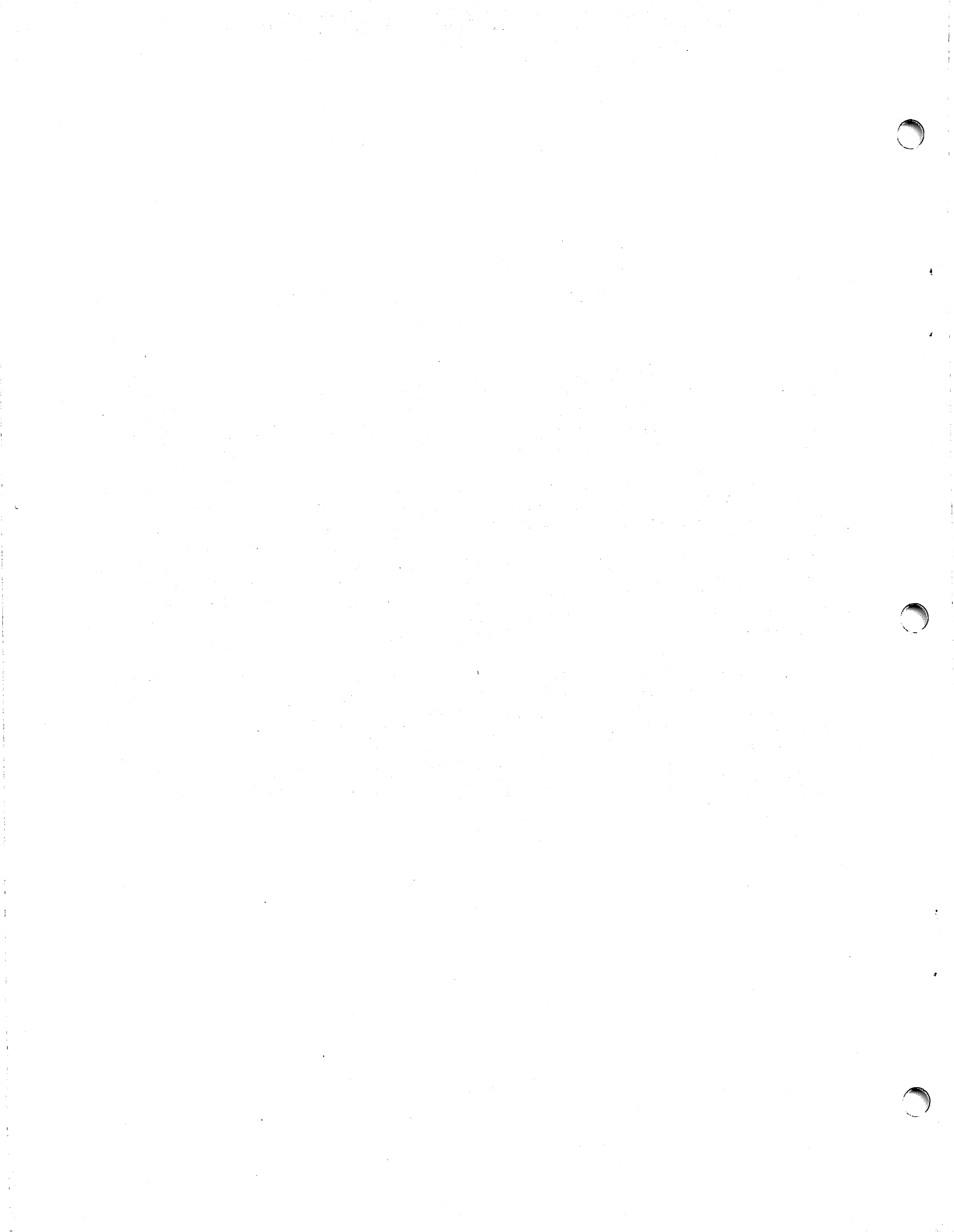
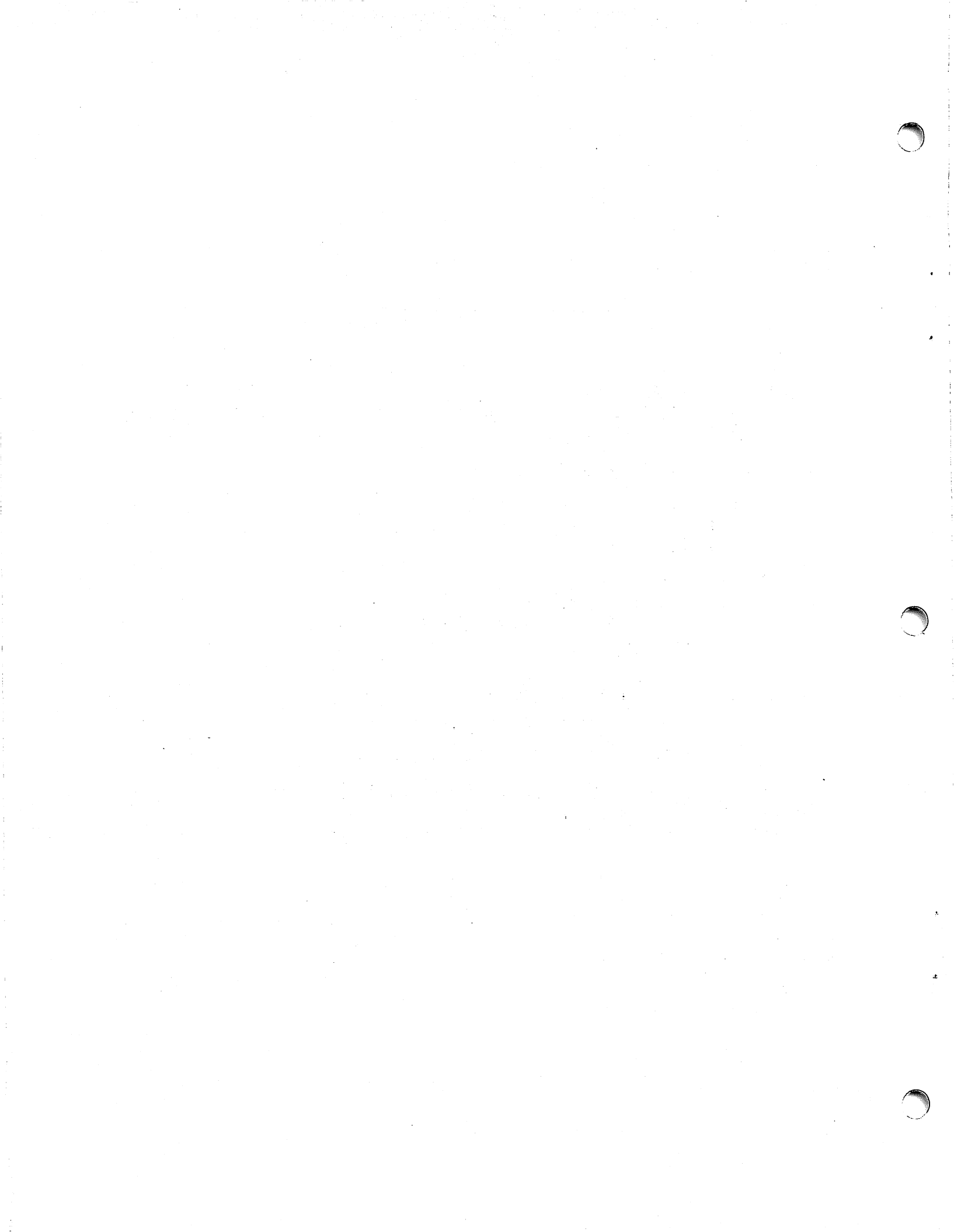


TABLE OF CONTENTS

	PAGE
CHAPTER 1	INTRODUCTION
1.1	The 2200 Portable Computing System 1
1.2	When the PCS Arrives 2
CHAPTER 2	PCS COMPONENTS
2.1	The Keyboard 3
2.2	The Display. 5
2.3	The Tape Cassette Drive. 5
2.4	The PCS Back Panel 6
CHAPTER 3	SIMPLE PCS OPERATIONS
3.1	Turning on the PCS 8
3.2	Loading and Running a Program. 8
3.3	Performing Quick Calculations. 11
3.4	The Edit Mode. 15
CHAPTER 4	PROGRAMMING THE PCS
4.1	The Extended BASIC Language. 18
4.2	PCS Commands 24
4.3	Entering Programs via the Keyboard 25
Appendix A:	Environmental Factors Affecting PCS Performance 28
Appendix B:	PCS Physical Specifications 30
Appendix C:	PCS Options and Peripherals 31



CHAPTER 1

INTRODUCTION

1.1 THE 2200 PORTABLE COMPUTING SYSTEM

The Wang 2200 Portable Computing System is an extremely compact and powerful computing system. It is an outgrowth of Wang's proven 2200 line of Extended BASIC Language computing systems. Its built-in processor includes all the features of the larger 2200 systems (except disk statements). The Extended BASIC language is powerful and easy to learn.

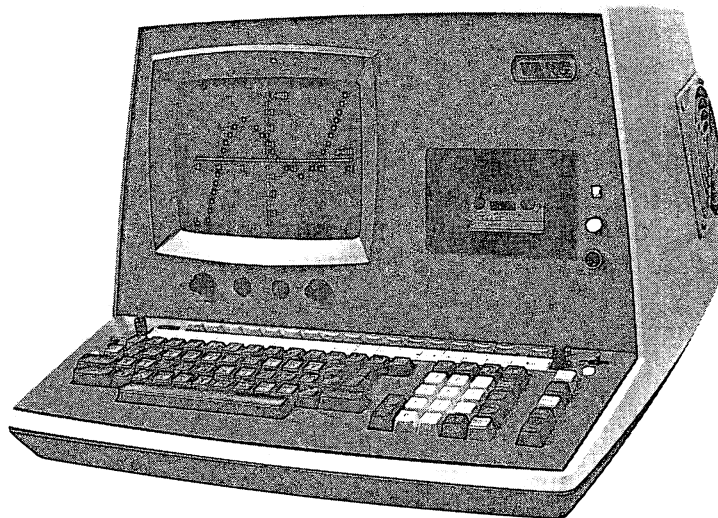


Figure 1-1 The Portable Computing System

The PCS comes with a standard 8192 bytes (8K) of random access user memory, which is expandable in 8K increments to 32K. The versatile BASIC Keyword/Alphanumeric keyboard permits easy program creation and data entry. A large, easy-to-read, 9-inch diagonal CRT can display up to 1024 characters. The PCS's built-in tape cassette drive offers low cost program and data storage capabilities.

The PCS can control a variety of optional peripheral devices including matrix printers, a drum plotter, and an output writer.

1.2 WHEN THE PCS ARRIVES

Special factory packing techniques require that the PCS be unpacked and inspected by your Wang Service Representative. When your PCS arrives, call your Wang Customer Engineering Office and request that this service be performed. Failure to follow this procedure voids your warrantee.

The PCS is compact, and can fit comfortably on a large desk or any similar work surface. It can be plugged into an ordinary 120-volt AC line; power consumption is approximately 260 watts.

As with all sensitive electronic equipment, the quality of the PCS's operating environment can affect performance. Hence, there are a few things to look out for when selecting a new site for the PCS. For example, you shouldn't plug it into an outlet that also powers motorized equipment, such as typewriters, air conditioners, water coolers or photo copiers. These devices radiate electromotive interference through the power line. Usually this is not destructive to the PCS, but it can cause random errors and malfunctions. Static electricity, airborne dirt and extremes of temperature and humidity also cause malfunctions. If you suspect that the operating environment may be a source of problems, or you are experiencing difficulties, read APPENDIX A. It discusses environmental problems in detail.

CAUTION:

The PCS is equipped with a small exhaust fan on the right side and entry vents on the left (see Figure 1-1). The PCS must not be placed immediately adjacent to file cabinets, or other vertical surfaces, that would impede air flow at these vital heat outlets.

CHAPTER 2

PCS COMPONENTS

2.1 THE KEYBOARD

The keyboard is your means of controlling the PCS. It lets you enter data, perform quick calculations, issue commands to the processor, write programs, and initiate program execution.

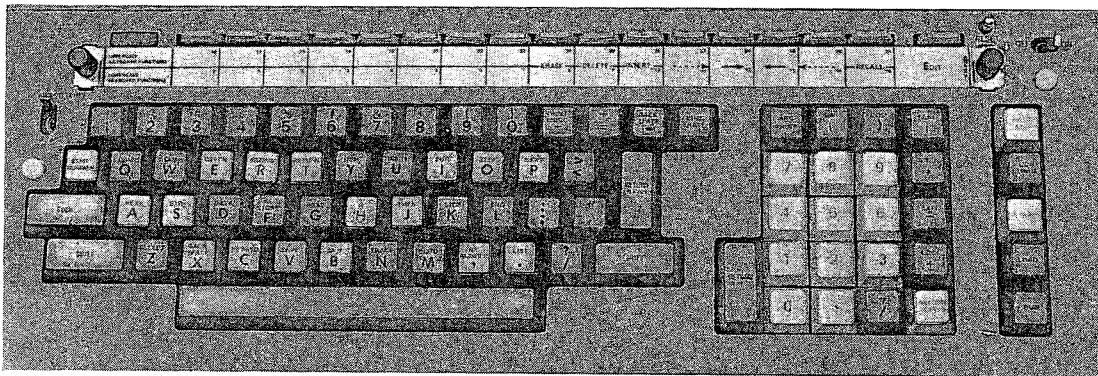


Figure 2-1 The PCS Keyboard

The Alphanumeric Zone

The alphanumeric zone of the keyboard is similar in design to a standard typewriter keyboard. It provides two modes of operation selected by means of the toggle switch, labeled "KEYWORD/A" and "A/a", at the left of the keyboard (see Figure 2-1 above).

The "KEYWORD/A" mode is specifically designed for use in program creation. When the keyboard is in the "KEYWORD/A" mode, the words that appear atop the alphabetic character keys can be entered by simply depressing SHIFT and the desired key. (These words are the principal words used in the BASIC language, and are known as "keywords.") In the "KEYWORD/A" mode, uppercase characters are produced without shifting. "KEYWORD/A" mode is convenient whenever you wish to use only uppercase characters.

In the "A/a" mode the alphanumeric keyboard works just like a typewriter in generating uppercase and lowercase characters; uppercase is obtained by depressing SHIFT together with the character key.

It is important to note that only those alphabetical keys with keywords on them are affected by the keyboard mode switch.

The Numeric Zone

The numeric zone of the keyboard is designed for rapid entry of numeric data, and for easy creation of numeric expressions. The PRINT key in the upper left corner facilitates the use of the system as a powerful, one line calculator. (This use is discussed in Section 3.3.) The numeric keys are grouped here for convenience only; digits may be entered by using the numeric keys in the numeric zone, or by using the numeric keys across the top of the alphanumeric zone.

The Special Function Keys

Across the top of the PCS keyboard are 16 Special Function Keys. These keys can be programmed to perform a variety of different types of tasks. Since each of the keys may be depressed in conjunction with the SHIFT key, an effective total of 32 Special Function Keys is available. The Special Function keys are numbered, on a special function strip, 0 - 15 and 16 - 31, with SHIFT used to obtain the latter range.

The EDIT Keys

When the PCS is in EDIT mode, the Special Function Keys lose their program-designated significance, and the eight keys at the right become system-defined edit keys. The EDIT key, to the right of the Special Function Keys, is used to enter EDIT mode. EDIT mode provides powerful line editing capabilities for program creation and data entry operations. (Use of EDIT mode is discussed in Section 3.4.)

The System Command Keys

The column of command keys at the right side of the keyboard provides single keystroke entry of some of the most commonly used system commands. (The commands are summarized in Section 4.2.)

The RESET Button

The RESET button provides a "last resort" means of terminating program execution. When depressed, it immediately ends program execution, clears the CRT, and displays the symbol:

READY

:_

(RESET should not be used if HALT/STEP can be used instead.)

The Power Switch

The PCS power switch is located in the upper right corner of the keyboard. This switch controls only the PCS power; peripheral devices have their own power switches. Turning off the PCS automatically clears from memory all program text and variables.

The RETURN(EXEC) Keys

There are three keys on the PCS keyboard that are labeled

RETURN
(EXEC)

These keys may be used interchangeably; they are in three locations merely for convenience. RETURN(EXEC) is the standard entry-terminator key. It is used to signal that a keyboard response, or entered program line, is complete. The RETURN(EXEC) key is usually referred to as simply (EXEC) or EXEC.

2.2 THE DISPLAY

The cathode ray tube display (CRT) is the PCS's principal means of giving you information. It serves as a vital feedback mechanism, displaying the result of each keystroke, allowing easy review and editing of data and program lines. Its speed, flexibility, and large character capacity make it ideal for interactive problem solving.

The PCS display has a capacity of 16 lines, with up to 64 characters per line. Its character set includes all the keyboard characters, plus additional foreign language characters and special symbols. Characters and symbols may be displayed with underlining. The complete character set is given in the 2200 Reference Manual. CRT brightness and contrast can be adjusted by means of the two controls on the front panel.

2.3 THE TAPE CASSETTE DRIVE

The PCS's built-in tape cassette drive offers a convenient means for saving programs and data. Turning off the PCS clears memory of all program lines; therefore programs should be saved on cassette if they are to be run repeatedly. After a program has been saved on cassette, it can be quickly and easily re-loaded into memory whenever required. (Section 3.2 discusses how to load and run a program that has been saved on cassette.)

Tape cassettes can also be used for saving numeric and alphanumeric data. (Cassette statement capabilities are summarized in Section 4.1.)

2.4 THE PCS BACK PANEL

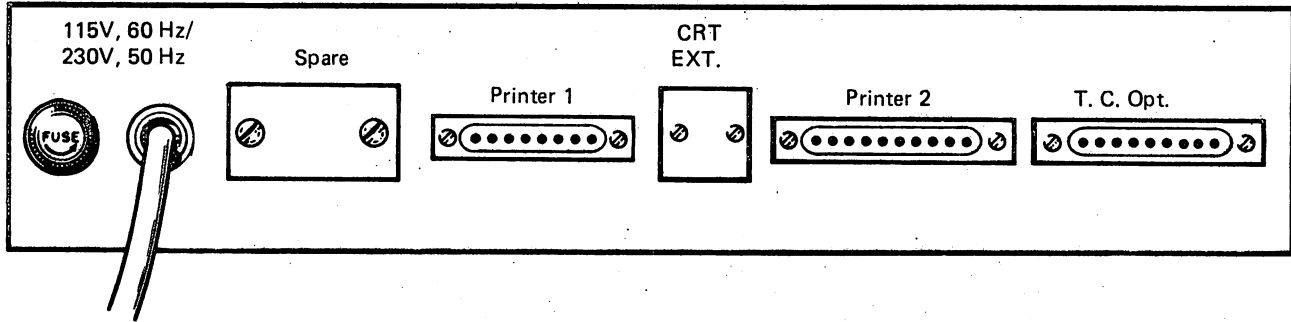


Figure 2-2 The PCS Back Panel

The back panel of the PCS, illustrated above, provides connectors for the PCS's optional peripheral devices. The connectors labeled "PRINTER 1" and "PRINTER 2" may be used for a matrix printer or for the Model 2272 plotter. If Option 61 (Output Writer) is purchased, an additional connector is provided at the location labeled "SPARE". This connector may be used only for the Output Writer. If Option 60 is purchased, an auxiliary CRT connector is installed in the back panel at the location labeled "CRT EXT."

When connecting peripheral cables, notice that the trapezoidal shape of the connectors permits only one orientation. Little pressure is needed to insert a properly aligned connector. After connecting a cable, press the retaining clips into the claws on the side of the plug (see Figure 2-3 below). This ensures a solid physical bond.

When removing peripheral cables from the PCS, first pop open the retaining clips, then grasp the cable's connector, and gently pull with a slight wiggling motion.

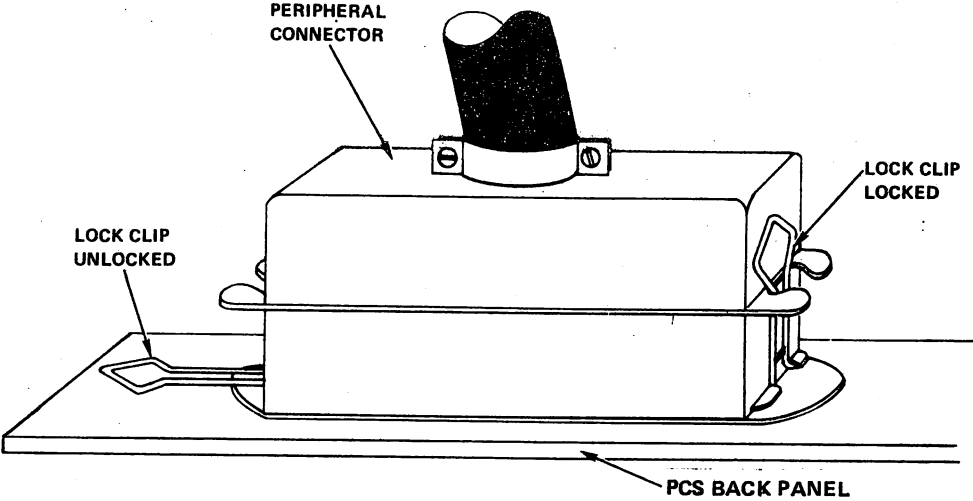


Figure 2-3 Retaining Clips in Side of Plug

CHAPTER 3

SIMPLE PCS OPERATIONS

3.1 TURNING ON THE PCS

The power switch, located in the top right corner of the keyboard, is used to turn on the PCS. Approximately 15 seconds after the power is turned on, the message

READY
:_

should appear in the upper left corner of the display. The colon indicates that the PCS is ready to accept a command, a BASIC statement for immediate execution, or a BASIC statement to be entered into memory as a program line.

NOTE:

If the "READY" message does not appear, advance the brightness control for the display. If this does not reveal the message, turn the power off, then on again. If the "READY" message still does not appear, call your Wang Service Representative.

The power at any of the PCS's optional peripheral devices may be on or off independently of the PCS itself. Power at these devices should be turned on before the PCS attempts operations on them.

When power is turned on, the PCS memory is clear of program text and variables.

3.2 LOADING AND RUNNING A PROGRAM

Loading a Program from Cassette

Programs that are saved on cassette tape must be loaded into memory before they can be executed. To load a program from cassette tape, the following procedure should be followed:

1. If a cassette is already mounted in the cassette drive, depress the REWIND button to the right of the cassette drive door.
2. When the rewinding stops, press the white cassette door release button. The door should open. Remove the cassette.

3. Key:

CLEAR EXEC

(The word "CLEAR" can be entered character-by-character or by depressing the CLEAR key at the right of the keyboard.)

4. Place the cassette containing the program to be loaded into the cassette drive door. Cassettes are mounted with the open tape edge up and the full spool of tape on the right. Close the cassette drive door.
5. If the program to be loaded has a program name (a program name contains a maximum of eight characters and is assigned to the program when it is saved on cassette), key:

LOAD "program name" EXEC

Where: "program name" is the name of the program enclosed in quotation marks.

(The word "LOAD" can be entered character-by-character, or by depressing the LOAD key at the right of the keyboard.) For example, to load a program named "CHI-SQR", you should key:

LOAD "CHI-SQR" EXEC

The display appears as,

:LOAD "CHI-SQR"

When EXEC is keyed, the cassette drive should begin operation, and the processing light at the right of the keyboard should illuminate. After loading is complete, the colon reappears on the display, and the processing light goes out.

Depress the rewind button next to the cassette drive door. Remove the cassette after rewinding is complete. Put the cassette back into its plastic box.

6. If the program to be loaded does not have a name assigned to it, or the name is not known, the program may be loaded by skipping to the beginning of the program, and then keying

LOAD EXEC

If the program to be loaded is at the very beginning of the cassette, simply key

LOAD EXEC

to load it. If the program is, for example, the fifth file on the cassette (where a file is any preceding program or data file), then the four preceding files must be skipped over before the desired program can be loaded. In this case, you should key

SKIP 4 F EXEC

and then, after the cassette stops and the colon reappears in the display, key

LOAD EXEC

to load the program. In general, if n programs or data files must be skipped over to get to the desired program, you should key

SKIP n F EXEC

where $n=1,2,3,\dots$

then load the program by keying

LOAD EXEC

After loading is complete, the colon reappears in the display. Depress the rewind button next to the cassette drive door. Remove the cassette after rewinding is complete. Put the cassette back into its plastic box.

Initiating Program Execution

To initiate execution of a program that is in memory, key

RUN EXEC

The word "RUN" may be entered character-by-character, or by simply depressing the RUN key at the right of the keyboard.

Responding to Program Prompts

After program execution has begun, the program itself, and any accompanying operating instructions, must provide a guide to the required operator responses and a labeling of the results. However, a few generalizations can be made about responding to program-displayed requests for information.

1. In general, a displayed question mark indicates that a keyboard response is sought.
2. If a numeric value is requested, the system permits a maximum of 13 digits, decimal point, sign, and a signed two-digit exponent. (However, the program itself may impose more restrictive limits on a response.) The sign of the value must precede the digits; the letter "E" is used to mark the beginning of the exponent. Some examples of system-acceptable numeric entries are:

25.15
-79.5
4.56E4
23.2437E-12

3. If an alphanumeric response is requested, any keyboard characters are acceptable to the system; however the program may impose additional restrictions on the response.

4. The line erase key (in the upper right corner of the alphanumeric zone) may be used to change a response, before **EXEC** is keyed. The backspace key (next to line erase) may be used to erase the previous character in a response, before **EXEC** is keyed.
5. When a response has been keyed in, and appears in the display in its desired form, key **EXEC** to enter the response and terminate the keyboard entry operation.
6. If a message of the form:

↑ERR XX

appears after keying **EXEC** and the question mark reappears, the entered response was unacceptable to the system. Check the form of the response, and reenter the desired value.

Stopping Program Execution

Should it be necessary to stop the execution of a program, and the program operating instructions do not describe how to do so, depress the HALT/STEP key at the top of the column of command keys, on the right side of the keyboard. HALT/STEP stops execution at the end of the currently executing BASIC statement, and displays the system colon. If, later, you wish to continue the program from where it was stopped, you may do so by keying CONTINUE EXEC (provided that you haven't done anything that would prevent the program from being continued. See the BASIC Language Reference Manual for actions that prohibit program continuation).

The RESET key also stops program execution, but it should not be used unless HALT/STEP fails to display the system colon. RESET can leave half-written, unreadable information on a tape cassette, if depressed during program execution. It also prevents the use of CONTINUE.

3.3 PERFORMING QUICK CALCULATIONS

Though the full power of the PCS is only realized in its ability to execute stored programs, the PCS can be used for quick calculations, without the need for a stored program. The numeric zone of the keyboard, with the PRINT key in the upper left corner, is designed to facilitate this use of the PCS. For example, to evaluate 745 * 93 key:

PRINT 745 * 93 **EXEC**

The result appears immediately

:PRINT 745 * 93
69285

:_

Operations such as this are known as "Immediate Mode operations", as opposed to execution of a stored program, which is called "Program Mode." When a BASIC statement, such as the PRINT statement above, is entered without a

preceding line number, it is interpreted as an Immediate Mode operation. The statement is executed, and is not stored as part of a program. Effectively, it is lost as soon as execution is complete. (The reappearance of the colon indicates that execution is complete.) Many, though not all, BASIC statements can be executed in the Immediate Mode. When using the PCS for quick calculations, the PRINT statement is commonly used, since it can evaluate an expression and display the result. Execution of PRINT in the Immediate Mode does not affect a program in memory, or any values of variables in memory.

In the above example, $745*93$ is an "expression". In BASIC an expression specifies a calculation to be performed. Thus, to use the PCS for quick calculations, you must know how to create the desired expression. The following table shows the five arithmetic operators of BASIC that are used in constructing expressions, and contrasts them with their commonly used algebraic equivalents.

<u>Operation</u>	<u>Operator</u>	<u>Example in BASIC</u>	<u>Example in Algebra</u>
Addition	+	3 + 4	3 + 4
Subtraction	-	11 - 7	11 - 7
Division	/	14/2	$\frac{14}{2}$
Multiplication	*	8 * 7	8 · 7
Exponentiation (raising to a power)	↑	4 ↑ 3	4 ³

Here are some examples of simple problems, stated in algebraic form, together with the PRINT statements used to evaluate them on the PCS.

<u>Problem</u>	<u>To Evaluate, Key:</u>
$\frac{6.02 \cdot 11.5}{9.46}$	PRINT 6.02 * 11.5/9.46 EXEC
$4.4 \cdot 3.14^3$	PRINT 4.4 * 3.14↑3 EXEC
$-255+4095-2047$	PRINT -255+4095-2047 EXEC
$3^3 \cdot 5^5$	PRINT 3↑3 * 5↑5 EXEC
$\left(\frac{4}{11}\right) 17^2 \cdot 5$	PRINT 4/11 * 17↑ 2.5 EXEC

Order of Evaluation

The order of evaluation for an expression is as follows:

- FIRST: All exponentiation is performed (left to right).
- SECOND: All multiplication and division is performed (left to right).
- THIRD: All addition and subtraction (negation) is performed (left to right).

The order of evaluation can be altered through the use of parentheses. Operations within parentheses are evaluated first. In constructing

expressions, parentheses may be nested within parentheses, with no limit to the number of pairs of parentheses used. The innermost parenthetical portions of the expression are always evaluated first. Here are some more complex problems in algebraic form and the PRINT statements used to evaluate them.

Problem

$$\left(\frac{1}{35}\right)^{4.2}$$

To Evaluate, Key:

PRINT (1/35)^4.2

EXEC

$$\frac{4(3.3+17.91)^{2.5}}{9(11)^{-6}}$$

PRINT 4 * (3.3 + 17.91)^2.5 / (9*11^(-6)) EXEC

$$7(8^3+5.9^6)$$

PRINT 7 * (8^3 + 5.9^6)

EXEC

$$- 3^3$$

PRINT (-3)^3

EXEC

$$- (3^3)$$

PRINT - 3^3

EXEC

$$\frac{\left(\frac{15.1^2}{3}\right) \left(\frac{12.4-3.011}{2}\right)}{14.35}$$

PRINT ((15.1^2/3)*((12.4-3.011)/2))/14.35 EXEC

Notice that there is no implied multiplication in BASIC. The * must always be used for multiplication.

In addition to being used to alter the order of evaluation, parentheses must be used to separate consecutive operators.

INCORRECTCORRECT

4^(-3)

4^(-3)

11/-7

11/(-7)

18*-9

18*(-9)

Built-in Functions

The PCS system has a built-in capability to evaluate a variety of mathematical functions. The built-in functions are listed below.

<u>Name</u>	<u>BASIC Form</u>	<u>Defined As</u>
Absolute Value	ABS(expression)	Absolute Value of the expression
Greatest Integer	INT(expression)	The greatest integer less than or equal to the value of the expression
Sign	SGN(expression)	1 if expression > 0 0 if expression = 0 -1 if expression < 0
Cosine	COS(expression,)	cosine of expression, where expression = angle in radians*

CHAPTER 3: SIMPLE PCS OPERATIONS

Sine	SIN(expression)	sine of expression, where expression = angle in radians*
Tangent	TAN(expression)	tangent of expression, where expression = angle in radians*
Arcosine	ARCCOS(expression)	arccosine of expression, where expression = angle in radians*
Arcsine	ARCSIN(expression)	arcsine of expression, where expression = angle in radians*
Arctangent	ARCTAN(expression)	arctangent of expression, where expression = angle in radians*
Natural Log	LOG(expression)	The natural logarithm of the expression
e ^x	EXP(expression)	e raised to the value of the expression
Square root	SQR(expression)	The square root of the expression

*Unless instructed otherwise, the PCS interprets the expression as radians. To use degrees instead, key SELECT D EXEC . All expressions in trigonometric functions are then interpreted as degrees. To reselect radian measure, key SELECT R EXEC ; to select grads, key SELECT G EXEC .

To evaluate

$$\sqrt{229}$$

key

PRINT SQR(229) EXEC

To evaluate

$$\frac{\ln(4 \cdot 1100)}{\ln 2}$$

key

PRINT LOG(4*1100)/LOG(2) EXEC

Since the built-in functions always use parentheses, they are always first in the order of evaluation.

The value π is permanently stored in your PCS, accurate to 13 significant digits. It may be incorporated into any expression by depressing the key marked " π " or by keying "#PI". In either case, it appears on the display as "#PI". For example, to evaluate

$$\pi \cdot 2.25^2$$

key

PRINT π *2.25 \uparrow 2 EXEC

the display appears as

```
:PRINT #PI * 2.25↑2
15.9043128088
```

The Random Number Generator

The PCS is capable of producing random numbers between zero and one. Random numbers are generated by means of the random number function:

RND (expression)

The random number function may be thought of as a means of accessing a fixed "list" of random values. If the value of the expression in the RND function is non-zero, the next random number in the "list" is produced. When the system is Master Initialized, the "list" is reset, so that the first execution of RND after Master Initialization always yields the same random number, the first random number on the "list." The "list" may also be reset by executing RND(0), which, in addition to resetting the "list", yields the zeroth random number in the list.

Entry Errors

If you make an error when constructing an expression, you can backspace to correct it, before keying **EXEC**; however, this has the unfortunate consequence of erasing all the possibly correct characters that are to the right of the error. This apparent dilemma can be resolved through the use of the PCS's EDIT mode. EDIT mode is described in the next section.

3.4 EDIT MODE

The PCS's EDIT mode allows you to easily edit Immediate Mode statements, data, and program lines, before **EXEC** is keyed, and to recall and edit program lines that are in memory.

In EDIT mode the Special Function Keys temporarily lose their user-defined significance, and the eight rightmost Special Function Keys perform specific editing tasks. Whenever the system colon or question mark is displayed, EDIT mode can be entered by depressing the EDIT key located to the right of the Special Function Keys. EDIT mode is signified by the substitution of an asterisk for the system colon or question mark.

The EDIT keys and their functions are:

<u>Key</u>	<u>Operation</u>
EDIT	used to enter EDIT mode. When depressed, an asterisk replaces the colon or question mark at the beginning of the line.
RECALL	used, in conjunction with a line number, to recall a program line from memory for editing.

- ←----- moves the cursor five spaces to the left; no characters affected.
- ← moves the cursor a single space to the left; no characters affected.
- moves the cursor five spaces to the right; no characters affected.
- moves the cursor a single space to the right, no characters affected.
- INSERT expands a line for additional text or data by inserting a space character at the cursor position.
- DELETE deletes the character at the cursor position and shortens the line.
- ERASE erases the line from the cursor position to the right end of the line.

All the character keys on the keyboard perform normally in EDIT mode.

As an example of the use of EDIT, suppose that you wish to perform this calculation in the Immediate Mode:

$$\frac{\left(\frac{15.1^2}{3}\right)\left(\frac{12.4-3.011}{2}\right)}{14.35}$$

Before keying EXEC, you check the display, which appears as:

```
:PRINT ((15.1+2/3)*(12.4-3.011/2))/14.35_
```

You notice that an additional pair of parentheses must surround the operation: 12.4 - 3.011. The keystrokes, and display results, to correct this line and perform the calculation are shown below:

<u>KEYSTROKE</u>	<u>DISPLAY AFTER KEYSTROKE</u>
EDIT	*PRINT ((15.1+2/3)*(12.4-3.011/2))/14.35_
----	*PRINT ((15.1+2/3)*(12.4-3.011/2))/14.35
----	*PRINT ((15.1+2/3)*(12.4-3.011/2))/14.35
----	*PRINT ((15.1+2/3)*(12.4-3.011/2))/14.35
----	*PRINT ((15.1+2/3)*(12.4-3.011/2))/14.35
INSERT	*PRINT ((15.1+2/3)*(_12.4-3.011/2))/14.35
(*PRINT ((15.1+2/3)*((12.4-3.011/2))/14.35
----	*PRINT ((15.1+2/3)*((12.4-3.011/2))/14.35
----	*PRINT ((15.1+2/3)*((1.4-3.011/2))/14.35
INSERT	*PRINT ((15.1+2/3)*((12.4-3.011/2))/14.35
)	*PRINT ((15.1+2/3)*((12.4-3.011)/2))/14.35
EXEC	*PRINT ((15.1+2/3)*((12.4-3.011)/2))/14.35
answer	24.86394761905
	:_

The RECALL key should be used only when a program line is to be recalled from memory for editing. To recall a program line from memory for editing, key:

line number EDIT RECALL

where: "line number" = the number of the line to be edited.

In addition to its normal function, keying **EXEC** when the system is in EDIT mode causes it to leave EDIT mode.

For more detailed information about use of EDIT mode, see the 2200 Reference manual.

CHAPTER 4

PROGRAMMING THE PCS

4.1 THE EXTENDED BASIC LANGUAGE

The programming language of the Portable Computing System is a version of the BASIC language extended to include a variety of powerful capabilities not usually found in BASIC. Extended BASIC is a general-purpose high-level programming language, suitable for technical as well as commercial applications. It is designed to be easily learned by beginning programmers.

The "statement" is the fundamental unit of instruction in BASIC. A statement tells the PCS what operation is to be performed by means of one or more "keywords." (The keywords of BASIC resemble English verbs and usually suggest the operations they cause.) In addition to keywords, most statements require some further specification of values to be used in the operations, and/or that which is to be operated upon. These additional specifications can be called "operands." For example,

```
20  PRINT    "HYPOTENUSE=";  SQR(A*A+B*B)
      keyword      operand          operand
      └──────────────────────────────────┘
                a statement
```

In a BASIC program a statement always appears on a numbered line. In the above example, the line number is the "20" that appears at the left. The PCS allows more than one statement to appear on a single line.

Sequence of Execution

In the normal sequence of statement execution, execution begins at the lowest line number, proceeds through that line until all the statements are executed, and then proceeds to the next higher numbered line. This sequence of execution can be altered by statements in the program. These statements, which are called "branching" statements, can cause the PCS to look to the beginning of any specified line to find its next statement.

Variables and Arrays

Values developed and maintained during program execution are stored in variables and array elements in memory. At the beginning of program execution, memory space is allocated to all variables and arrays used in the program. Variables are referred to by name, array elements by name and subscripts. Variables and arrays may be either numeric or alphanumeric, depending upon the format of the information they are to contain.

In BASIC there are 286 legal numeric variable names. They are, A, B, C...Z and A0, A1, A2,...A9, B1, B2...Z7, Z8, Z9. Names for alphanumeric variables are the same except that a "\$" is always added, for example, A\$, B\$, C\$...Z\$, A0\$, A1\$...Z8\$, Z9\$. One and two-dimensional alphanumeric and numeric arrays may be used. Array names are the same as variable names except that numeric subscripts are added in parentheses. Thus, a two-dimensional numeric array named B2() with dimensions 10 by 10 would have elements

B2(1,1), B2(1,2), B2(1,3)...	B2(1,10)
B2(2,1), B2(2,2), B2(2,3)...	B2(2,10)
.	
.	
.	

B2(10,1), B2(10,2), B2(10,3)...	B2(10,10)
---------------------------------	-----------

Numeric values that are to enter into mathematical operations are stored in a special floating-point format. Only numeric variables, or numeric array elements, can contain values in this format, and therefore only they can be used in mathematical operations. Numeric variables can assume any value in the ranges

$$10^{-99} \leq \text{value} < 10^{100}$$

0

$$-10^{100} < \text{value} \leq -10^{-99}$$

They may contain up to 13 digits, decimal point, sign, and signed two digit integer exponent.

Alphanumeric variables and array elements can be assigned any alphanumeric character strings or hex codes, but may not enter into mathematical operations (however, a very limited binary arithmetic capability exists for values in alphanumeric variables). Alphanumeric variables and array elements can have a length of 1 to 64 characters.

Statement Descriptions

Brief descriptions of the PCS's BASIC statements are given below. The statements categorized as "General BASIC Statements" and "Tape Cassette statements" are completely documented in THE WANG BASIC LANGUAGE REFERENCE MANUAL (#700-3038G). The "General BASIC Statements" are grouped here by programming function. The statements categorized as "Sort Statements", "Matrix Statements", and "General I/O Instruction Set" are completely documented in the reference manuals of those names. It should be noted that some of the statements described in these latter three categories are at least as "general-purpose" as some of the "General BASIC Statements." These categories arose as a result of the continuing development of the Extended BASIC Language in Wang systems over several years. For an elementary textbook on programming in BASIC, the reader is referred to the WANG BASIC LANGUAGE

PROGRAMMING MANUAL. For a detailed description of tape cassette operations, the reader is urged to consult the 2200 TAPE CASSETTE DRIVE REFERENCE MANUAL.

General BASIC Statements - I/O

PRINT PRINTUSING	Used chiefly for outputting data to a printer or display. Alphanumeric or numeric values can be output. PRINT differs from PRINTUSING primarily in that PRINT outputs in various system-prescribed formats, whereas PRINTUSING outputs according to a user specified image statement (%).
HEXPRINT	Used primarily for diagnostic purposes, it prints the hex values of an alphanumeric variable, array element or array.
INPUT	Used primarily for keyboard data entry. Optionally displays a prompt.
KEYIN	Accepts a single character from a device buffer and branches.
PLOT	Moves plotter pen; performs other plotter control functions.

General BASIC Statements - Branching

GOTO...	Performs a branch to the first statement at a specified line number.
IF...THEN...	Compares alphanumeric or numeric values, branches to a line number if a specified condition is true.
GOSUB RETURN	These statements are used to branch to an ordinary subroutine and to return after subroutine execution. Up to 45 levels of subroutine nesting may be used.
DEFFN' GOSUB'	DEFFN' designates the beginning of a marked subroutine. GOSUB' branches to the marked subroutine, and can pass multiple argument values to it. Execution of marked subroutines can, optionally, be initiated by Special Function Key from the keyboard. (DEFFN' can also be used to associate an alphanumeric value with a Special Function Key.)
FOR...TO... NEXT...	FOR...TO and NEXT define the boundaries of a program loop, and control its execution.
RETURN CLEAR	Clears the last subroutine return location. Used when a normal RETURN is not desired.

General BASIC Statements - Variable and Array Definition

DIM	Specifies the dimensions of an array and the length of alphanumeric variables and array elements. (Arrays may
-----	---

be one or two-dimensional, numeric or alphanumeric. Alphanumeric variables and array elements may be 1-64 characters in length.)

- COM Dimensions a variable or array, and designates it as common to several program modules.
- COM CLEAR Changes the designation of variables and arrays from common to non-common, or vice-versa.

General BASIC Statements - Data Conversion

- CONVERT Converts numeric values to a character string in an alphanumeric variable, or vice versa.
- PACK
UNPACK PACK converts numeric values to Wang packed decimal format, and assigns them to an alphanumeric variable or array. Used principally to conserve storage space for data files. UNPACK reverses the operation.

General BASIC Statements - Alphanumeric

- INIT Assigns a specific character or hex value to each character location (byte) of an alphanumeric variable or array.
- STR() Allows any string of contiguous character locations in an alphanumeric variable, or array element, to be used as if it were an entire alphanumeric variable or array element.
- POS() Finds the first character in an alphanumeric variable that satisfies a specified relationship.
- LEN() Determines the length of the value of an alphanumeric variable, not counting trailing spaces.
- HEX() Allows any hex codes to be used in a program.

General BASIC Statements - Alphanumeric Binary Logical Operators

- AND
OR
XOR Perform the logic operations AND, OR, EXCLUSIVE OR on the bits of a specified alphanumeric variable.
- BOOL Performs any of sixteen logic operations on the bits of a specified alphanumeric variable.
- ROTATE Rotates the bits of each byte of a specified alphanumeric variable to the left, one to seven bit locations.

General BASIC Statements - Alphanumeric Binary Arithmetic Operators

BIN	Converts a numeric value in Wang internal numeric format to a one byte binary value, and assigns the value to the first byte of an alphanumeric variable.
ADD ADDC	Adds values in binary, with or without propagation of a carry from lower to higher bytes. A one byte binary value may also be added to each byte of an alphanumeric variable.
VAL()	Converts a one byte binary value to a numeric value in Wang internal numeric format.

General BASIC Statements - Miscellaneous

DEFFN	Defines a function of one variable for use in a program.
SELECT	Selects a peripheral device address for a particular class of I/O operations. Performs miscellaneous other selection operations.
TRACE	A debugging aid, it produces a printed record of program execution.
STOP	Stops program execution; permits continuation; optionally displays a message.
END	Ends program execution. Displays amount of unused memory.
REM	Allows programmer's comments to be inserted into a program.
ON ERROR	Causes a branch when a system-detected error occurs during execution. Provides the program with information about the error.
DATA READ RESTORE	Allow a list of program constants to be accessed sequentially, or randomly.
LET...=...	General-purpose assignment statement. It evaluates a numeric expression, and assigns the result to one or more numeric variables; or assigns an alphanumeric value to one or more alphanumeric variables.

Tape Cassette Statements

DATASAVE DATALOAD	The DATASAVE statement transfers values from variables and arrays in memory to cassette tape. DATALOAD reverses the operation. The values saved as a result of a single DATASAVE statement are automatically formatted as one "record".
----------------------	---

SKIP BACKSPACE	These statements are used to skip or backspace the tape over a specified number of files or records, or to skip to the end, or backspace to the beginning, of a file.
DATARESAVE	Used to update a record on cassette.
REWIND	Rewinds the cassette at high speed.
LOAD	Loads a program, and clears all or a portion of the existing program text and non-common variables as it does so. Used for program overlaying.
DATASAVE BT DATALOAD BT	Transfers 256 bytes of unformatted data from memory to one physical tape block, and vice versa.

The Matrix Statements

The matrix statements are a group of statements designed to simplify and speed matrix algebra operations on numeric arrays. The following table summarizes the matrix statement capabilities.

MATRIX STATEMENT OPERATIONS					
Operation	E	I	A/N	Description	Example
Matrix Addition		✓		array = array + array	MAT X = Y + Z
Matrix Subtraction		✓		array = array - array	MAT X = Y - Z
Matrix Multiplication		✓		array = array * array	MAT X = Y * Z
Scalar Multiplication		✓		array = scalar expression * array	MAT X = (3) * Y
Matrix Inversion and Determinant		✓		matrix = inverse matrix and scalar variable = determinant of matrix	MAT X = INV(Y), D
Matrix Transposition		✓		array = transpose of array	MAT X = TRN(Y)
Matrix Assignment		✓		array = array	MAT X = Y
Identity Matrix	✓			array = identity matrix	MAT X = IDN
Zero Matrix	✓			each array element = 0	MAT X = ZER
Matrix Constant	✓			each array element = 1	MAT X = CON
READ Matrix	✓		✓	array elements = successive DATA values	MAT READ X
PRINT Matrix	✓		✓	print all array elements	MAT PRINT X
INPUT Matrix	✓		✓	array elements = values from keyboard	MAT INPUT X
Redimension Array	✓		✓	array shape changed as specified	MAT REDIM X(R, C)
✓ E = Array can be redimensioned explicitly ✓ I = Resultant array redimensioned implicitly ✓ A/N = Can be performed on alphanumeric as well as numeric arrays					

Sort Statements

MAT CONVERT MAT SORT MAT MOVE MAT MERGE	These statements are used to sort numeric and alphanumeric arrays at much higher speeds than could be achieved by writing a sort program using general BASIC statements. MAT MERGE effects a high speed merge of sorted strings.
MAT COPY	This statement effects a high speed transfer of data from alphanumeric array to alphanumeric array. Array element boundaries are ignored.
MAT SEARCH	This statement searches an alphanumeric array for character strings that satisfy a specified relationship.

General I/O Instruction Set

\$GIO	The \$GIO statement, unlike any other BASIC language I/O statement, permits a technique similar to machine language programming to be used to custom-tailor I/O operations, in a "general input/output" format.
\$IF ON	This statement tests the device-ready condition of a specified output device, or the data-ready condition of a specified input device, and initiates a branch to a specified line if a ready condition is sensed.
\$TRAN	The \$TRAN statement provides a high-speed character conversion capability, implemented by a table look-up, or character replacement, procedure.
\$PACK \$UNPACK	Perform packing and unpacking operations of numeric and alphanumeric data, according to a variety of specifiable formats.

4.2 PCS COMMANDS

In all Wang 2200 systems there exists a primary distinction between "commands" and "BASIC statements". BASIC statements are the fundamental units of instruction in a program; they are the means by which the PCS is instructed to carry out its "useful work". Some BASIC statements can also be executed without being part of a stored program, in an Immediate Mode operation. Commands, by contrast, can never be used in a program; they initiate operations auxiliary to the "useful work" carried out by the PCS.

The PCS Commands are summarized below. For more detailed descriptions see the Wang BASIC LANGUAGE REFERENCE MANUAL.

CLEAR	Clears all or a portion of the program text currently in memory and all variables, or clears common or non-common variables only.
CONTINUE	Continues program execution after it has been stopped.

HALT/STEP	Depressed during program execution, HALT/STEP stops execution. Depressed when execution has been stopped, it displays and executes the next statement.
LIST	Lists all the program text currently in memory, or a portion of it. Optionally lists in 15-line segments for CRT display.
RENUMBER	Renumbers the lines of a program in memory, or some portion of a program in memory.
RESET	Immediately stops program execution; displays the "READY" symbol.
RUN	Initiates program execution at the lowest line, or any specified line.
LOAD	Searches for a program on cassette tape, and loads it into memory.
SAVE	Saves the program currently in memory onto cassette tape.

4.3 ENTERING PROGRAMS VIA THE KEYBOARD

Suppose that you wish to enter a simple program such as Example 4.1.

Example 4-1 A Program to Calculate the Arithmetic Mean of Entered Values

```

10 REM A PROGRAM TO CALCULATE THE MEAN OF ENTERED VALUES
20 INPUT "ENTER VALUE", X
30 S = S + X
40 N = N + 1
50 INPUT "MORE VALUES TO ENTER (Y OR N)", R$
60 IF R$ = "Y" THEN 20
70 PRINT "AVERAGE="; S/N

```

The first step is to clear memory of any program text or variables. This can be done by simply keying:

CLEAR

(The word "CLEAR" can be entered by depressing the CLEAR key at the right side of the keyboard, or by typing it in character-by-character in uppercase.) When a program is entered via the keyboard, it is entered one line at a time. Depressing signifies that the line is complete. The processor then examines the line for adherence to the rules of BASIC syntax, and saves it in memory. Thus, to enter the first line of Example 4-1, you simply type:

```
10 REM A PROGRAM TO CALCULATE THE MEAN OF ENTERED VALUES EXEC
```

The number "10" at the left is the line number. It can be entered character-by-character, or by depressing the key labeled "STMT. NO." at the

left of the keyboard. Depressing the "STMT. NO." key always generates a line number 10 greater than the highest numbered line already in memory.

The statement on line 10 is a REM statement (an abbreviation of "REMARK"), the keyword "REM" can be entered by depressing SHIFT and the key labeled

REM Y

provided that the keyboard is in the "KEYWORD/A" mode. The word "REM" can also be entered character-by-character in uppercase.

After

EXEC

 is keyed, the display appears as:

```
:10 REM A PROGRAM TO CALCULATE THE MEAN OF ENTERED VALUES
:
_
```

The colon on the next line indicates that another line can now be entered.

The remaining lines of Example 4-1 are entered the same way as line 10. Spaces may be entered wherever you feel they will contribute to clarity. Unless they are within quotation marks, they are ignored by the PCS, and will have no effect on program execution.

Correcting Mistakes before | | |------| | EXEC | |------| Has Been Keyed

If you make an error in typing in a line, and notice the error before you key

EXEC

, you can backspace to correct it (with the "BACKSPACE" key that is in the upper right corner of the alphanumeric zone). Alternatively, you can enter EDIT mode, by depressing the EDIT key at the right of the Special Function strip, and then use the EDIT Special Function Keys to move the cursor and correct the error. (See Section 3.4 for more about EDIT mode.) You should not depress the RECALL key unless the line to be edited is already in memory, that is,

EXEC

 has been keyed.

Correcting Mistakes after | | |------| | EXEC | |------| Has Been Keyed

If you make an error when entering a line, and the error constitutes a violation of BASIC language syntax, then an "ERR" message will appear when

EXEC

 is depressed. The "ERR" message looks like this:

```
↑ERR XX
```

where XX is a two digit error code.

The two digit error code signifies the type of error detected. A list of error codes and their significance can be found in the WANG BASIC LANGUAGE REFERENCE MANUAL.

If an error message appears, the line must be corrected, because the program will not execute with the line as it stands. However, there are other types of errors that may not cause an error message, but that you may notice and wish to correct. For example, you might accidentally type "VALEUS" instead of "VALUES" in line 50 of Example 4-1.

There are two ways to correct any error in a line that is in memory. (After `EXEC` is keyed, the line is in memory even if it has a syntax error.) The first way is simply to reenter the line correctly. As long as the same line number is used, the new line will replace the old one. The second way is to recall the erroneous line with the EDIT mode RECALL key. Suppose line 50 is in memory, and has an error to be corrected. Key:

```
50 EDIT RECALL
```

Line 50 appears in the display with an asterisk at the left, indicating that the PCS is in EDIT mode. The EDIT Special Function keys can now be used to move the cursor, insert, delete and erase; the other keys perform normally. When the line is corrected, key `EXEC` to substitute the corrected line for the incorrect one, and leave EDIT mode.

A program line can be deleted by keying its line number followed immediately by `EXEC`.

Program lines can be entered in any sequence; the PCS automatically keeps them in ascending line number sequence in memory. Any whole number in the range 0 to 9999 may be used as a line number. Line numbers in intervals of 10 are often used to accommodate the insertion of additional lines.

In Example 4-1 each line contains just one BASIC statement. Though programmers frequently write just one statement per line for clarity, multiple statements can appear on a single line. Each statement must be separated from the next by a colon. The maximum number of statements that can appear on a line is restricted only by the maximum line length, which is 192 keystrokes.

APPENDIX A

ENVIRONMENTAL FACTORS AFFECTING PCS PERFORMANCE

Physical Environment

In selecting a PCS location there are three important physical environmental factors to be considered. These are temperature, humidity, and cleanliness.

The temperature is the most important factor to consider because it can vary greatly from day to day. The recommended operating temperature range is from 65°F to 75°F, but the allowable range is from 20°F to 90°F.

Low outside temperatures are usually not a problem because nearly all locations are heated. High temperatures can be a problem because many locations do not have air conditioning. If the PCS is used where temperatures exceed the maximum specified, component failure rates will drastically increase.

If an air conditioning unit is already installed, or if one is to be installed, it is imperative that a separate power line be used. If a separate power line is not used, system errors will occur when the air conditioning is in use.

While air conditioning is good for maintaining the proper temperature, it also removes moisture from the air, thereby lowering the humidity. If the PCS is used in a carpeted room, the lower humidity plus the static generating capability of carpets and synthetic clothing impart static electrical charges to operating personnel. When an operator comes in contact with the PCS, the resultant static spark is uncomfortable, causes system malfunctions, and can even destroy recorded data.

If carpeting is to be installed where the PCS is used, be sure it is a non-static variety. If carpeting already exists, and it is not a non-static carpet, it can be treated with an anti-static spray. Alternatively, an electrically conductive mat can be installed to prevent a static charge build-up. Carpets treated with an anti-static spray should be thoroughly cleaned before the first treatment, and retreated at least once every three months thereafter. If an electrically conductive mat is used, it should be installed under the PCS operating area and must be properly connected to an earth ground.

The recommended humidity range is from 35% to 65% R.H., but 20% to 80% R.H. is allowable. (In cold weather, the humidity in heated buildings can be 10% or lower.) Humidifiers and dehumidifiers can be installed to increase or decrease the humidity as required.

Dirt and grease can accumulate rapidly on circuit boards and components, and can form a film that traps heat and provides a leakage path for signals. To prevent failures due to dirt, all air conditioning, heating and ventilating units should have air filters. These filters should be cleaned or replaced regularly. In areas where ordinary filters do not remove airborne dirt sufficiently, an electrostatic filter can be installed.

Electrical Environment

A 15 ampere, 115 VAC power line is adequate for the PCS. This line should be regulated to within +10% and should be noise free. For optimum performance, it is recommended that the system have its own AC power line.

If the line is not sufficiently regulated to the limits indicated above, a constant voltage transformer can be installed. If the line is noisy, however, a detailed analysis of the problem must be performed to insure a correct solution.

Computers and peripherals are extremely susceptible to Electromagnetic Interference (EMI). In general, the source of the EMI must be determined before a solution is proposed. EMI can enter the PCS by conduction along wiring and cabling or by direct radiation. If sources of EMI, which include office machines, air conditioning units, electric motors, machinery and arc welders, are in close proximity to the PCS, EMI will enter by direct radiation. The noise generating device should be moved, repaired or filtered to prevent it from interfering with the PCS. If the source of the noise cannot be found, an EMI filter with a cut-off frequency near 10 KHz should be installed on the PCS's AC power line. In all cases, check that the AC power line has been properly installed in steel conduit, and the conduit is properly connected to junction boxes. Also, insure that other devices, including fluorescent lighting, are not connected to the same AC power line. In extreme locations, such as those with nearby arc welders, it may be necessary to shield any peripheral cables.

APPENDIX B

PCS PHYSICAL SPECIFICATIONS

Size

Height 13 1/2 in. (34.3 cm)
Depth 20 1/2 in. (52 cm)
Width 19 3/4 in. (50.2 cm)

Weight

57 lb (25.8 kg)

CRT

Display Size 9 in. diagonal (22.9 cm)
Capacity 16 lines, 64 characters/line
Character Size:

Height 0.125 in. (0.32 cm)
Width 0.125 in. (0.32 cm)

Cassette Drive

Cassette Capacity 300 record blocks per 150 ft
cassette (256 bytes per block)
Recording/Search Speed 7.5 in./sec (19 cm/sec)
Rewind Speed 7.5 ft/sec (2.3 m/sec)

Power Requirements

115 or 230 VAC + 10%
50 or 60 Hz \pm 1/2 Hz
260 watts

Fuses

2.5a @ 115v/60 Hz
1.2a @ 230v/50 Hz

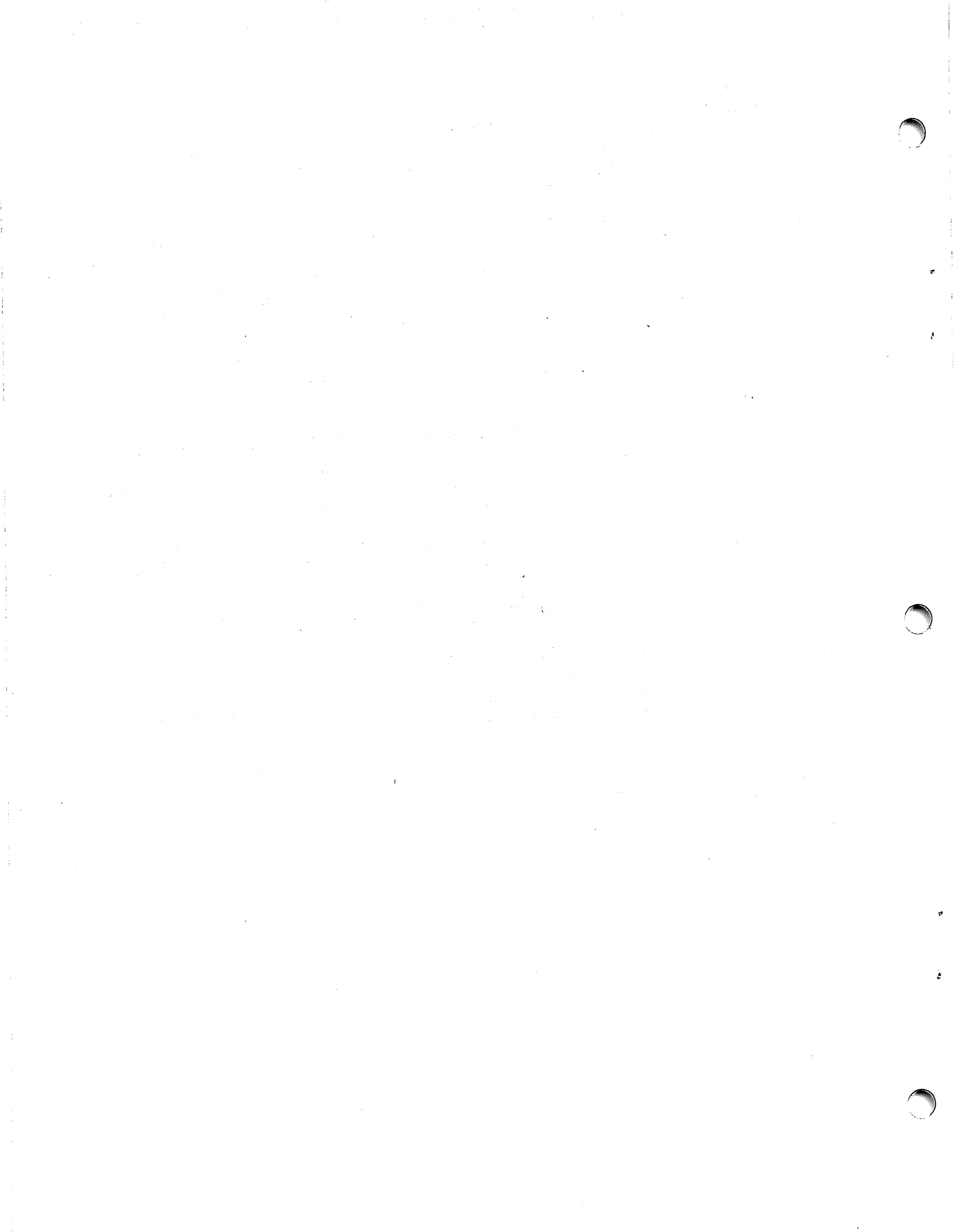
Operating Environment

65°F to 75°F (13°C to 24°C) recommended
20°F to 90°F (10°C to 32°C) allowable
20% to 80% relative humidity allowable
35% to 65% relative humidity recommended

APPENDIX C

PCS OPTIONS AND PERIPHERALS

- Option 60: Option 60 offers, as a package, the following PCS enhancements:
1. Keyboard clicker: installed in the keyboard, it emits an audible click when a key has been sufficiently depressed to enter a character.
 2. Auxiliary Display Connector: A rear-panel connector that permits an auxiliary CRT display to be connected.
 3. Audio Alarm: a device that generates a brief audible tone when an ASCII bell code (hex 07) is received at the CRT address.
- Option 61: Option 61 is an Output Writer (a modified IBM Selectric® II typewriter) capable of fully automatic printing under PCS control. Average printing speed is approximately 13 characters per second. An additional rear panel connector is installed in the PCS for the Output Writer.
- Model 2221W: A high speed, quiet, 200 cps matrix printer with a full upper and lowercase character set and a 132-character line.
- Model 2231W: A compact, quiet, 120 cps matrix printer featuring a full upper and lowercase character set and a 112-character line.
- Model 2272-1-2 A digital drum plotter that provides point plotting and continuous line plotting over an area up to 16 inches (40.6 cm) wide and of unlimited length. The 2272-1 is equipped with a single pen. The 2272-2 is equipped with three selectable pens.
- Additional CPU Memory CPU memory can be expanded in 8K byte increments to 32K bytes.



To help us to provide you with the best manuals possible, please make your comments and suggestions concerning this publication on the form below. Then detach, fold, tape closed and mail to us. All comments and suggestions become the property of Wang Laboratories, Inc. For a reply, be sure to include your name and address. Your cooperation is appreciated.

700-4027

TITLE OF MANUAL: PORTABLE COMPUTING SYSTEM INTRODUCTION

COMMENTS:

Fold

Fold



Fold

FIRST CLASS
PERMIT NO. 16
Tewksbury, Mass.

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

— POSTAGE WILL BE PAID BY —

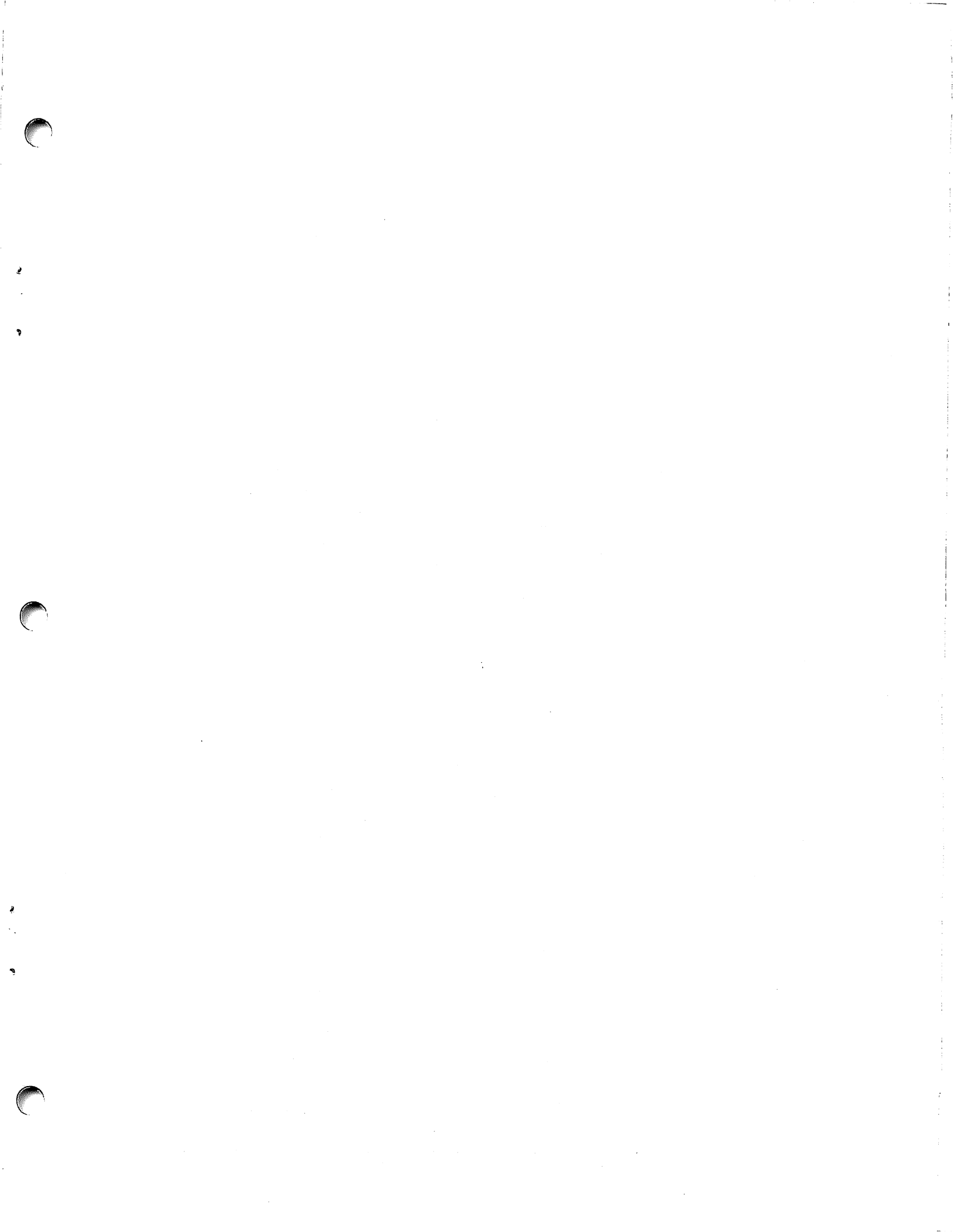
WANG LABORATORIES, INC.
836 NORTH STREET
TEWKSBURY, MASSACHUSETTS 01876

Attention: Marketing Department

Fold

Cut along dotted line.

Printed in U.S.A.



**WANG LABORATORIES
(CANADA) LTD.**
49 Valleybrook Drive
Don Mills, Ontario M3B 2S6
TELEPHONE (416) 449-2175
Telex: 069-66546

WANG EUROPE, S.A.
Buurtweg 13
9412 Ottergem, Belgium
TELEPHONE 053/704514
Telex: 26077

**WANG DO BRASIL
COMPUTADORES LTDA.**
Rua Barao de Lucena No. 32
Botafogo ZC-01 20,000
Rio de Janeiro RJ, Brasil
TELEPHONE 226-4326, 266-5364
Telex: 2123296 WANG BR

**WANG COMPUTERS
(SO. AFRICA) PTY. LTD.**
Corner of Allen Rd. & Garden St.
Bordeaux, Transvaal
Republic of South Africa
TELEPHONE (011) 48-6123
Telex: 960-86297

**WANG INTERNATIONAL
TRADE, INC.**
836 North Street
Tewksbury, Massachusetts 01876
TELEPHONE (617) 851-4111
TWX 710-343-6769
Telex: 94-7421

WANG SKANDINAVISKA AB
Pyramidvaegen 9A
S-171 36 Solna, Sweden
TELEPHONE 08/27 27 95
Telex: 11498

WANG NEDERALND B.V.
Damstraat 2
Utrecht, Netherlands
(030) 93-09-47
Telex: 47579

WANG PACIFIC LTD.
902-3 Wong House
26-30, Des Voeux Road, West
Hong Kong
TELEPHONE 5-435229
Telex: 74879 WANG HX

WANG INDUSTRIAL CO., LTD.
110-118 Kuang-Fu N. Road
Taipei, China
TELEPHONE 784181-3
Telex: 21713

WANG GESELLSCHAFT M.B.H.
Murlingasse 7
A-1120 Vienna, Austria
TELEPHONE 85.13.54, 85.13.55
Telex: 74640 Wang a

WANG S.A./A.G.
Markusstrasse 20
CH-8042 Zurich 6, Switzerland
TELEPHONE 41-1-60 50 20
Telex: 59151

WANG COMPUTER PTY. LTD.
55 Herbert Street
St. Leonards, 2065, Australia
TELEPHONE 439-3511
Telex: 25469

WANG ELECTRONICS LTD.
Argyle House
Joel Street
Northwood Hills
Middlesex, HAG ILN
TELEPHONE Northwood 28211
Telex: 923498

WANG FRANCE S.A.R.L.
Tour Gallieni, 1
78/80 Ave. Gallieni
93170 Bagnolet, France
TELEPHONE 33.1.3602211
Telex: 680958F

WANG LABORATORIES GmbH
Moselstrasse 4
6000 Frankfurt AM Main
West Germany
TELEPHONE (0611) 252061
Telex: 04-16246

WANG DE PANAMA (CPEC) S.A.
Apartado 6425
Calle 45E, No. 9N. Bella Vista
Panama 5, Panama
TELEPHONE 69-0855, 69-0857
Telex: 3282243

WANG COMPUTER LTD.
302 Great North Road
Grey Lynn, Auckland
New Zealand
TELEPHONE Auckland 762-219
Telex: CAPENG 2826

WANG COMPUTER PTE., LTD.
37, Hill Street
Singapore 6, Republic of Singapore
TELEPHONE 333641, 321791
Telex: RS 23987 GENERCO

WANG COMPUTER SERVICES
836 North Street
Tewksbury, Massachusetts 01876
TELEPHONE (617) 851-4111
TWX 710-343-6769
Telex: 94-7421

DATA CENTER DIVISION
20 South Avenue
Burlington, Massachusetts 01803
TELEPHONE (617) 272-8550

WANG

LABORATORIES, INC.

836 NORTH STREET, TEWKSBURY, MASSACHUSETTS 01876, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

Printed in U.S.A.
700-4027
8-76-3M

Price: see current list