TANDY
TRS-80
COMPUTER PRODUCTS ™

Cat. No. 26-1183A

BAR CODE READER

For Model 100

Works with Disk/Video Interface
Hand held scanner reads
bar codes for

- Data Entry
- Program Entry

Useful in inventory control, order entry, data collection, material tracking.

Software included:

- Decoder modules for UPC, 3 of 9 and Plessey
- Simple inventory program

OWNER'S MANUAL

Custom manufactured in U.S.A.
for Radio Shack, a Division of Tandy Corporation

TERMS AND CONDITIONS OF SALE AND LICENSE OF TANDY COMPUTER
EQUIPMENT AND SOFTWARE PURCHASED FROM RADIO SHACK
COMPANY-OWNED COMPUTER CENTERS, RETAIL STORES AND RADIO
SHACK FRANCHISEES OR DEALERS AT THEIR AUTHORIZED LOCATIONS

LIMITED WARRANTY

I. CUSTOMER OBLIGATIONS

A. CUSTOMER assumes full responsibility that this computer hardware purchased (the "Equipment"), and any copies of software included with the Equipment or licensed separately (the "Software") meets the specifications capacity, capabilities, versatility, and other requirements of CUSTOMER

B. CUSTOMER assumes full responsibility for the condition and effectiveness of the operating environment in which the Equipment and Software are to function, and for its installation

II. LIMITED WARRANTIES AND CONDITIONS OF SALE

A For a period of ninety (90) calendar days from the date of the Radio Shack sales document received upon purchase of the Equipment, RADIO SHACK warrants to the original CUSTOMER that the Equipment and the medium upon which the Software is stored is free from manufacturing defects. **This warranty is only applicable to purchases of Tandy Equipment by the original customer from Radio Shack company-owned computer centers, retail stores, and Radio Shack franchisees and dealers at their authorized locations** The warranty is void if the Equipment's case or cabinet has been opened, or if the Equipment or Software has been subjected to improper or abnormal use If a manufacturing defect is discovered during the stated warranty period, the defective Equipment must be returned to a Radio Shack Computer Center, a Radio Shack retail store, a participating Radio Shack franchisee or a participating Radio Shack dealer for repair, along with a copy of the sales document or lease agreement The original CUSTOMER'S sole and exclusive remedy in the event of a defect is limited to the correction of the defect by repair, replacement, or refund of the purchase price, at RADIO SHACK'S election and sole expense RADIO SHACK has no obligation to replace or repair expendable items

B RADIO SHACK makes no warranty as to the design, capability, capacity, or suitability for use of the Software, except as provided in this paragraph Software is licensed on an "AS IS" basis, without warranty. The original CUSTOMER'S exclusive remedy, in the event of a Software manufacturing defect, is its repair or replacement within thirty (30) calendar days of the date of the Radio Shack sales document received upon license of the Software. The defective Software shall be returned to a Radio Shack Computer Center, a Radio Shack retail store, a participating Radio Shack franchisee or Radio Shack dealer along with the sales document

C Except as provided herein no employee, agent, franchisee, dealer or other person is authorized to give any warranties of any nature on behalf of RADIO SHACK.

D **EXCEPT AS PROVIDED HEREIN, RADIO SHACK MAKES NO EXPRESS WARRANTIES, AND ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE IS LIMITED IN ITS DURATION TO THE DURATION OF THE WRITTEN LIMITED WARRANTIES SET FORTH HEREIN.**

E Some states do not allow limitations on how long an implied warranty lasts, so the above limitation(s) may not apply to CUSTOMER

III. LIMITATION OF LIABILITY

A **EXCEPT AS PROVIDED HEREIN, RADIO SHACK SHALL HAVE NO LIABILITY OR RESPONSIBILITY TO CUSTOMER OR ANY OTHER PERSON OR ENTITY WITH RESPECT TO ANY LIABILITY, LOSS OR DAMAGE CAUSED OR ALLEGED TO BE CAUSED DIRECTLY OR INDIRECTLY BY "EQUIPMENT" OR "SOFTWARE" SOLD, LEASED, LICENSED OR FURNISHED BY RADIO SHACK, INCLUDING, BUT NOT LIMITED TO, ANY INTERRUPTION OF SERVICE, LOSS OF BUSINESS OR ANTICIPATORY PROFITS OR CONSEQUENTIAL DAMAGES RESULTING FROM THE USE OR OPERATION OF THE "EQUIPMENT" OR "SOFTWARE." IN NO EVENT SHALL RADIO SHACK BE LIABLE FOR LOSS OF PROFITS, OR ANY INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY BREACH OF THIS WARRANTY OR IN ANY MANNER ARISING OUT OF OR CONNECTED WITH THE SALE, LEASE, LICENSE, USE OR ANTICIPATED USE OF THE "EQUIPMENT" OR "SOFTWARE."**

NOTWITHSTANDING THE ABOVE LIMITATIONS AND WARRANTIES, RADIO SHACK'S LIABILITY HEREUNDER FOR DAMAGES INCURRED BY CUSTOMER OR OTHERS SHALL NOT EXCEED THE AMOUNT PAID BY CUSTOMER FOR THE PARTICULAR "EQUIPMENT" OR "SOFTWARE" INVOLVED.

B. RADIO SHACK shall not be liable for any damages caused by delay in delivering or furnishing Equipment and/or Software

C No action arising out of any claimed breach of this Warranty or transactions under this Warranty may be brought more than two (2) years after the cause of action has accrued or more than four (4) years after the date of the Radio Shack sales document for the Equipment or Software, whichever first occurs

D Some states do not allow the limitation or exclusion of incidental or consequential damages, so the above limitation(s) or exclusion(s) may not apply to CUSTOMER

IV. SOFTWARE LICENSE

RADIO SHACK grants to CUSTOMER a non-exclusive, paid-up license to use the TANDY Software on **one** computer subject to the following provisions:

A Except as otherwise provided in this Software License, applicable copyright laws shall apply to the Software

B Title to the medium on which the Software is recorded (cassette and/or diskette) or stored (ROM) is transferred to CUSTOMER, but not title to the Software

C CUSTOMER may use Software on one host computer and access that Software through one or more terminals if the Software permits this function

D CUSTOMER shall not use, make, manufacture, or reproduce copies of Software except for use on **one** computer and as is specifically provided in this Software License Customer is expressly prohibited from disassembling the Software.

E CUSTOMER is permitted to make additional copies of the Software **only** for backup or archival purposes or if additional copies are required in the operation of **one** computer with the Software, but only to the extent the Software allows a backup copy to be made. However, for TRSDOS Software, CUSTOMER is permitted to make a limited number of additional copies for CUSTOMER'S own use

F CUSTOMER may resell or distribute unmodified copies of the Software provided CUSTOMER has purchased one copy of the Software for each one sold or distributed. The provisions of this Software License shall also be applicable to third parties receiving copies of the Software from CUSTOMER

G All copyright notices shall be retained on all copies of the Software.

V. APPLICABILITY OF WARRANTY

A The terms and conditions of this Warranty are applicable as between RADIO SHACK and CUSTOMER to either a sale of the Equipment and/or Software License to CUSTOMER or to a transaction whereby Radio Shack sells or conveys such Equipment to a third party for lease to CUSTOMER

B The limitations of liability and Warranty provisions herein shall inure to the benefit of RADIO SHACK, the author, owner and or licensor of the Software and any manufacturer of the Equipment sold by Radio Shack

VI. STATE LAW RIGHTS

The warranties granted herein give the **original** CUSTOMER specific legal rights, and the **original** CUSTOMER may have other rights which vary from state to state

8/84

IMPORTANT INFORMATION

This equipment generates and uses radio frequency energy and if not installed and used properly (that is, in strict accordance with the manufacturer's instructions) may cause interference to radio and television reception. It has been type-tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- reorient the receiving antenna.
- relocate the computer with respect to the receiver.
- move the computer away from the receiver.
- plug the computer into a different outlet so that computer and receiver are on different branch circuits.

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

10 9 8 7 6 5 4 3 2

Contents

Introduction

With the expansion of the bar code industry in data and program applications, Radio Shack introduces a bar code wand for your Model 100 Portable Computer.

Radio Shack's Bar Code Reader improves the efficiency of your computer by supplying data faster and more accurately than it can be obtained through a keyboard. This hand-held scanning device speeds information to your computer quickly by scanning bar codes. Together with the special decoding software provided, the wand is capable of reading three code languages, or symbologies — UPC, Code 3 of 9, and Plessey.

This operator's guide provides operating and maintenance information for the Bar Code Reader, including:

- An introduction to bar codes
- A description of the wand
- Operating procedures
- Use of the wand with the Model 100
- Maintenance procedures
- Troubleshooting hints

Also provided are two simple inventory programs to illustrate typical bar code applications. (See Appendix C.)

Code Formats

Bar codes are combinations of marks and spaces that are formatted to represent numeric characters. When scanned by a Bar Code Reader, they constitute an effective alternative to the keyboard as a mechanism for entering information into computers. Bar code scanning is faster than key entry and is also more accurate since most codes have built-in checksums which prevent incorrect reads from being entered.

Reading bar code symbols provides a rapid and accurate method for capturing data. Many industries over the years have developed bar code systems especially tailored to their particular needs. The result has been the creation of a variety of code formats and ''dialects'' of formats. Most of these revolve around products of a single manufacturer or around products in a related industry. Only recently have industry-wide standards begun to emerge.

Applications for bar codes continue to expand, with boundaries apparently set only by the limits of human imagination and innovation. The two general application categories are data entry and program entry. Data entry includes specific applications, such as:

- Factory floor data collection
- Remote site data collection
- Retail point-of-sale data collection
- Inventory control
- Item identification for assemblies in service, repair, or testing
- Security checkpoint verification
- File folder tracking
- ID card recognition

Potential program entry applications include:

- Programming microprocessors in electronic equipment (appliances, video recorders, games, etc.).

- Programming both personal computers and calculators.

One reason that bar codes are attractive for program entry, in addition to speed and accuracy, is that inexpensive mass distribution of software is possible.

Codes are read as the light from the wand's optical sensor, located in the tip of the wand, scans across the bars. The black bars absorb the light, while the white bars reflect the light back into the scanner. These light and dark patterns are transformed into electrical impulses which, in turn, are translated into data for your computer.

The wand can read codes with bars and spaces no narrower than .012 inch (0.3 mm); denser codes (with narrower bars and spaces) will not be recognized by the Bar Code Reader. The upper limit of the unit width is limited by the practical scanning speed. As the bars and spaces become wider, the scanning speed must increase accordingly. We recommend the width of the widest bar or space be no more than .10 inch.

Black and white bars possess the best characteristics for scanning, but the Bar Code Reader is responsive to other colors. The reflectance of a color must reach the specified level in order to be picked up by the scanner.

It is important to understand the importance of a suitable contrast ratio between the printed bars and the background media. There should be a surface reflectivity between 70 and 90 percent. Glossy surfaces may limit the reading angle of the wand.

Bar codes are typically printed with ink on some type of paper surface and, consequently, are subject to both wear and abuse. Common forms of wear and abuse which will have a negative impact on readability and should be guarded against include:

- Liquid spills on the bar code symbol (coffee, cola, etc.)
- Ink smearing or erased bars which result from poor ink quality, a rough scanner tip, or excessive pressure on the bar code surface
- Scratches through the bar code symbol
- Writing across the bar code symbol

The use of a thin, non-reflecting plastic protective cover will help to avert these problems and will prolong print media life. A strip of plastic suitable for this purpose is included with this package. (A strip of non-reflecting mending tape makes a good permanent protective cover.)

Your Bar Code Reader and decoding software have been designed to provide a high **First Read Rate** when a good symbol is scanned.

A low First Read Rate is normally caused by either a poorly printed or damaged bar code, or improper operation of the system by the operator.

Each operator will need to experiment with the wand to learn the proper position for the best read rate. Experts say it may take up to 20 minutes of practice to become proficient in the use of the wand. It should also be noted that hand-held scanners read labels and tags best on flat surfaces. It is a bit trickier to achieve high First Read Rates on the curved surface of a can.

Bar Codes Used with Model 100

Three different types of bar codes can be used to enter data into the Model 100 with the Bar Code Reader.

Universal Product Code (UPC) is a numeric-only retail code that is used on thousands of commercial products. Although not recommended as an industrial code, it must be applied by suppliers of food products, hardware, paperback books and magazines, tapes and records, and other high volume items. It is commonly used for packaging, counting, and data processing. Many types of UPC symbologies exist; however, the decoder program reads only the most popular type, UPC-A.

Code 3 of 9 is an uppercase, alphanumeric code that is widely used by industry. It has also been incorporated by the Department of Defense in its LOGMARS program, a labelling standard designed to expedite materials handling in the military. This code was originally named 3 of 9 because three of the elements are wide and six are narrow, and because the symbology provided for 39 data digits. It now consists of 43 characters: start/stop, 10 digits, 26 letters of the alphabet and six symbols. Your Model 100 program will decode a maximum of 23 characters per line of bar code.

Plessey Code is a binary code in which each character is built from a group of 4 bars and 4 spaces. It generates numeric characters and has been used in many commercial inventory marking systems.

For more information on these bar codes, refer to Appendix D.

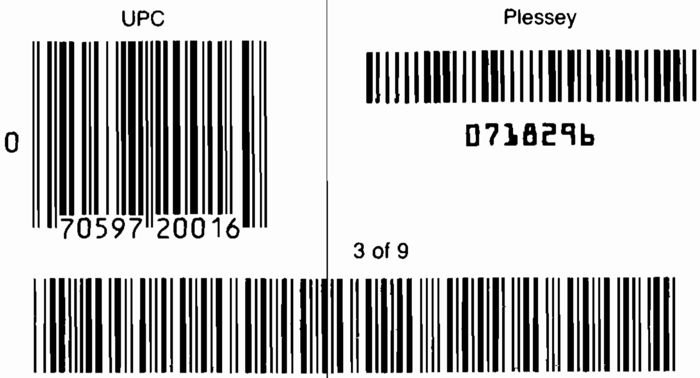


Figure 1. Standard Bar Code Formats

To use the Bar Code Reader in decoding, you will need the following items:

- Model 100 Portable Computer
- Bar Code Reader (Radio Shack Catalog Number 26-1183)
- Product Marking Code-Identification programs supplied with the Bar Code Reader
- Cassette Recorder

The decoders supplied with your Bar Code Reader include:

- UPC-A
- 3 of 9
- Modified MSI Plessey

Other decoder modules will be made available later, as the need for a particular format develops.

Setting Up the Model 100 for Scanning

The Model 100 serves as the power source for the Bar Code Reader, providing the convenience of battery operation or standard 120 VAC power (26-3804) for extended use. See your Model 100 owner's manual for instructions on supplying power to the computer.

Connecting the Bar Code Reader to the Model 100

Before making any connections to your computer, be sure the power switch is in the **OFF** position.

1. Refer to Figure 2 and insert the wand connector ① (9-pin subminiature D-style) into the Bar Code Reader connector ② located on the left side of the Model 100.

2. Move the Model 100's power switch to the **ON** position.

3. Press the Push-to-Read Switch ③ once; then release it. The LED indicator ④ in the wand's tip ⑤ should light only when the push-to-read switch is depressed, indicating the Bar Code Reader is receiving power. The push-to-read switch activates the wand for scanning when it is pressed, and it must remain depressed during the scan.

Note: Remember that, when the LED is lit, a considerable amount of power is being consumed. Use of the AC adapter is recommended whenever possible in order to conserve battery life.

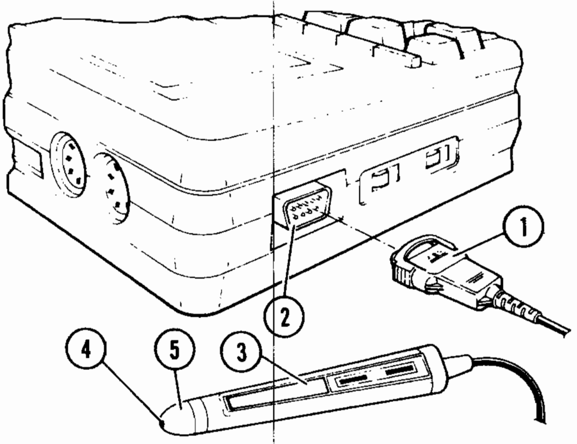


Figure 2. Connecting the Bar Code Reader to the Model 100

Connecting the Cassette Recorder to the Model 100

The machine language decoder programs, which convert bar codes into language understood by the computer, are recorded on the cassette tape supplied with the Bar Code Reader. (It is a good idea to make at least one copy of this tape; in case of loss or destruction of the original, the backup copy can be used.) There are 10 files stored on the cassette tape in the following order:

- B3OF9.CO
- UPC.CO
- PLESY.CO
- READBC.BA
- INVCRE.BA
- INVENT.BA
- B3OF9D.CO
- UPCAD.CO
- PLESYD.CO
- RDBCDV.BA

Note: Two complete sets of these files are stored on the tape. The first six files are for use with the cassette recorder. The last four files are for use with the Disk/Video Interface (26-3806).

1. With the Model 100's power switch in the **OFF** position, plug the cassette recorder connector into the **CASSETTE** connector located on the rear side of the Model 100.

2. Press **EJECT** on the recorder to open the cassette compartment; then insert the tape supplied with the Bar Code Reader.

3. Close the compartment window.

4. Slide the power switch on the Model 100 to the **ON** position.

Before you proceed . . . If you are using a Disk/Video Interface (26-3806) in conjunction with your Model 100, refer to Appendix A for special instructions. For Disk/Video purposes, Appendix A replaces the information contained in the following two sections. (**Loading the Decoder Programs into the Model 100** and **Using the Bar Code Decoder with BASIC**).

Loading the Decoder Programs into the Model 100

When you turn on the power switch of the Model 100, after making the peripheral connections, the Main Menu is displayed on the LCD screen. Notice that the built-in application program **BASIC**, in the upper left-hand corner, is shaded by the Cursor. Model 100 BASIC is an extended version of the BASIC programming language. Press (ENTER) to enter the Execute Mode. Type:

```
SOUND OFF (ENTER)
```

Rewind the tape; then press **PLAY** on the cassette recorder and set the volume control between 4 and 6.

Before the machine language program can be loaded into the computer's memory, space must be provided. Type this statement:

```
CLEAR 11Ø, 61784 ENTER
```

Now type CLOADM "filename" (where filename is the particular decoder program you want). Then press ENTER. If you are using a cassette recorder, the decoder programs you will use are B3OF9.CO, UPC.CO, and PLESY.CO. You may store all 3 decoder programs in your Model 100, or just the one you will be using.

While the computer is looking for the program on the tape, several messages appear on the screen to let you know what stage the search is in. When the program has been loaded, the **Ok** prompt appears.

For example, if you used the filename UPC, your screen should show:

```
Skip :  B30F9
Found:  UPC
Top  :  61788
End  :  62611
Exe. :  61824
Ok
```

The number 61788 labelled "Top" represents the place in the Model 100 memory where the specified machine language program begins. The "End" number 62611 is the ending address for the specified program. The last number 61824 gives you the location at which that program begins execution.

After loading the decoder into memory from the cassette, you may save it in the directory by typing (all in one line):

```
SAVEM "filename", Top address,
End address, Execution address ENTER
```

where filename is the name of the decoder program (i.e., B3OF9, UPC, or PLESY); and top, end, and execution address are the same numbers which appeared on the display when you loaded the program from the tape. For the UPC decoder, the command would look like this:

```
SAVEM"UPC", 61788, 62611, 61824
```

The decoder program is now stored in the Model 100 and may be accessed by a simple BASIC program for reading bar codes.

Note: If your application requires the use of only one decoder, you may be able to conserve the amount of memory available to BASIC by deleting the SAVEM command. See Appendix E for details.

Using the Bar Code Decoder with BASIC

The following BASIC program illustrates how simple it is to input data into the computer with the Bar Code Reader.

Example 1:

```
    1Ø CLEAR 1ØØ, 61784
    2Ø RUNM "filename"
    3Ø OPEN "WAND:" FOR INPUT AS 1
    4Ø INPUT #1, A$
    5Ø PRINT A$
    6Ø IF A$<>" " THEN 4Ø
    7Ø CLOSE:CALL 61807
```

Line 10 clears 100 characters of string storage for the data which will be read, and also sets 61784 as the TOP address BASIC may use during program execution. (See **CLEAR** in Chapter 16 of the Model 100 owner's manual.)

Line 20 loads the decoder program ''filename'' into the correct RAM location where it may be accessed by the BASIC program. The filename is the name of the particular decoder as you stored it in RAM, i.e., B3OF9, UPC, or PLESY. (See **RUNM** in Chapter 16 of the Model 100 owner's manual.)

Line 30 opens a file for the WAND as an INPUT device with buffer file number 1. (See **OPEN** in Chapter 16 of the Model 100 owner's manual.)

Line 40 inputs the data as a character string called A$. This instruction waits to receive a string of characters from the decoder program when a successful scan of a bar code is completed. The LINE INPUT# command may also be used. (See **INPUT#** and **LINE INPUT#** in Chapter 16 of the Model 100 owner's manual.)

Lines 10 through 40 are common to all bar code BASIC programs. After executing line 40, you will have a string of characters from the bar code decoder (A$) which you may use in your program in any way you wish. Line 50 simply prints the data string on the display and, of course, line 60 loops back to wait for another scan from the Bar Code Reader unless the user has aborted by pressing (ENTER).

Important Note: The code in line 7 is very important. It is essential to execute these two statements before returning to the menu:

```
CLOSE
CALL 61807
```

The CALL to 61807 must be preceded by closing the file with the CLOSE command. These instructions reset some of the operating system parameters and prevent an error condition.

To exit a bar code program, simply press the (ENTER) key instead of scanning a line of bar code. This will return a ''null string'' from the decoder. Your BASIC program can then test for a null string and GOTO an exit routine.

The following is an example of a BASIC program which allows you to first select which type of code you wish to read; it then reads and displays the code as it is scanned with the wand. When you have finished reading or wish to read a different type of code, simply press the ENTER key. The program will then prompt you for a decision as to whether or not you want to continue reading. If you press N, it will return you to the menu; if you press Y, it will return you to the point where you enter the type of code to be read. The BREAK key will also cause a return to the menu.

Note: This program is stored on your cassette tape. Its filename is "READBC".

Example 2:

```
10 MAXFILES=1:CLEAR 100, 61784:CLEAR
19 ' Set up the driver filenames for
   choice
20 F$(1)="UPC":F$(2)="B30F9":F$(3)=
   "PLESY"
30 CLS:PRINT:PRINT" 1. UPCA   2. 3 of
   9     3. Plessey"
40 PRINT@240,"Please choose:";:A=INSTR
   ("123",INPUT$(1)):IF A=0 THEN
   BEEP:GOTO 40 ELSE PRINTA;
50 RUNM F$(A)' run selected driver
60 ON ERROR GOTO 200 ' trap errors for
   orderly exit
70 CLS:OPEN "WAND:"FOR INPUT AS 1
79 'loop to here for more reads
80 PRINT"Wand ready: ";:INPUT #1,
   A$:PRINTA$
90 IF A$<>""THEN 80' get next read
   unless null input
100 'fall through to here if user
    pressed ENTER
```

```
110 CLS:PRINT"RESUME READING WAND?
    (Y/N) ";
119 ' Get response. Ask again if not y
    or n, else obey
120 Q$=INPUT$(1):PRINTQ$:ON
    INSTR(" YyNn",Q$)\2+1 GOTO
    110,80,220
200 ' simple error trap and exit
210 PRINT:IF ERR=18THEN PRINT"BREAK
    pressed" ELSE PRINT"Error ";ERR;".
    Press any key..";INPUT$(1)
220 CLOSE:CALL 61807:MENU
```

Lines 20, 30 and 40 allow you to select the type of code you want to read.

Line 90 tests for a **null string**, which allows you to break out of the program with the ENTER key.

The **ON ERROR** command in line 60 will detect the BREAK key and, in this case, will return to the menu (line 210).

Reading a Bar Code Tag

Holding the Bar Code Reader in the preferred position (see "Bar Code Reader Orientation"), place the tip in the white margin at the beginning of the bar code. Press the push-to-read switch to activate the wand and keep it depressed until the scan is completed.

Begin the scan by moving the wand lightly, but rapidly, across the bar code. A successful scan is indicated by an audible **BEEP** from the Model 100. If there is no BEEP, then you must try again until the BEEP is heard.

To test your Bar Code Reader, load the 3 of 9 decoder, as described previously; then type in the BASIC program listed in Example 2 above (or CLOAD ''READBC'' from cassette), and scan the following test code. The message **TEST** should appear on your Model 100 display.

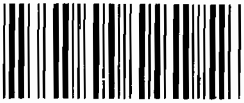


Figure 3. Test Code for 3 of 9

Note: To avoid damaging the test code on this page, you should place a piece of thin, *non-reflective* plastic over it while scanning.

Bar Code Reader Orientation

As the wand operator, you have a significant influence on the successful operation of the Bar Code Reader. Parameters that you have under your control include switch actuation, tilt angle, and scan speed.

To use the Bar Code Reader, hold it in your hand like an ordinary pen or pencil, placing your thumb in the middle of the push-to-read switch. For the best performance, the wand should be held at a slight angle — between 10° and 20° (see Figure 4). The angle between the wand and a line perpendicular to the bar code tag is called the tilt angle. This Bar Code Reader should operate within a range of 0° to 30°. The tilt should be from front to back and not from side to side. When scanning a line of bar code, you should move your hand and forearm in the direction of the scan without twisting your wrist. This will prevent the Bar Code Reader from tilting side to side.

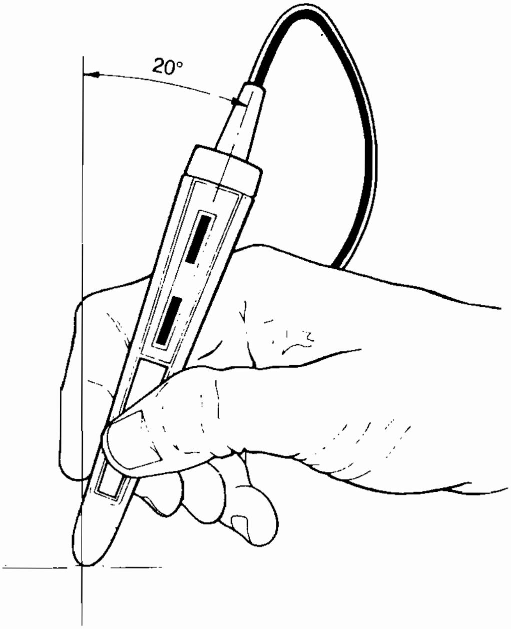


Figure 4. Preferred Bar Code Reader Position

Scan Speed

The scanning speed for the Bar Code Reader is rated at 3-30 inches/sec. (7.6-76 cm/sec.) and should be uniform and rapid. It is not necessary to apply pressure to the label surface when scanning. Instead, place the wand tip lightly in contact with the bar code tag as it is moved across the surface.

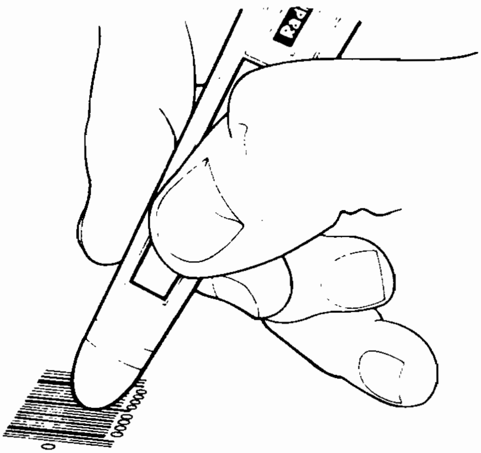


Figure 5. Scanning Bar Codes

Scanning Hints:

The quality of the bar codes can influence the performance of the Bar Code Reader. A good-quality symbol has solid bars with sharp edges, and spaces that are free of ink specks or dirt. Avoid a poor-quality symbol by scanning sections of the bar code with the cleanest and most clearly defined bars and spaces. Placing a thin, non-reflective plastic cover on the surface will help to protect the label from wear and prolong print media life.

Other factors which affect the system's performance include printing specifications and decoding software. Below are some examples of good-quality and poor-quality bar code symbols.

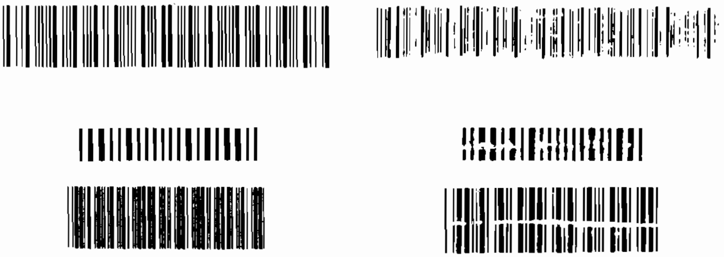


Figure 6. Good- and Poor-Quality Bar Codes

Maintenance

Taking proper care of your Bar Code Reader is one way to insure quality performance. The wand tip should be checked periodically for foreign particles and wear resulting from use.

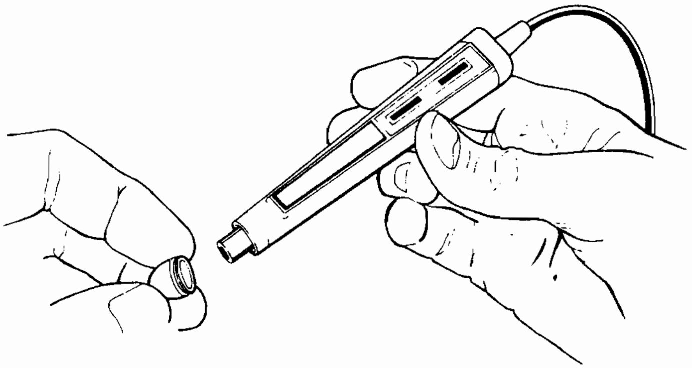


Figure 7. Cleaning the Wand Tip

To remove dirt or obstructions which collect in the tip opening or on the optical sensor window, disconnect the Bar Code Reader from the Model 100 and unscrew the tip from the wand assembly. Use a cotton swab or similar device and a liquid cleaner to clean the tip opening. A lint-free cloth dampened in liquid glass cleaner can be used to remove dirt, dust or fingerprints from the sensor window. Be careful not to disturb the position of the sensor. Never spray cleaner directly on the sensor or wand.

Replacing the Tip

The tip is the only part of the Bar Code Reader that you will be able to service. After a period of time, the tip will become worn due to use and have to be replaced. A replacement tip (Part Number ART5208) can be obtained through the nearest Radio Shack store.

Troubleshooting

With the exception of the tip, the Bar Code Reader contains no parts that can be replaced by the operator. If it fails to function properly, take the wand to your nearest Radio Shack store for assistance. Before returning the wand, however, use the check list below to make sure the malfunction is not in the peripheral equipment.

1. Is the Bar Code Reader's connection secure where it plugs into the Model 100?

2. Is the LED light visible when the push-to-read switch is depressed?

   Check the batteries in the Model 100, or check the AC power source.

3. Is the LED dirty?

   Unscrew the tip and, with a cotton swab, wipe clean the red LED.

4. Be sure there are no errors in your BASIC input routine. Compare your program to lines 10 through 40 listed in Example 1. These commands and yours should be exactly the same.

5. Scan the test code in Figure 3, using the 3 of 9 decoder and the BASIC program given previously (see Example 2).

Appendix A / Use with Disk/Video Interface

Loading the Decoder Programs into the Model 100

Refer to your Disk/Video Interface Owner's Manual for proper connections and operating procedures.

When you turn on the power switch of the Model 100, after making the peripheral connections, the Main Menu is displayed on the LCD screen. Notice that the built-in application program **BASIC**, in the upper left-hand corner, is shaded by the Cursor. Model 100 BASIC is an extended version of the BASIC programming language. Press ENTER to enter the Execute Mode. Type:

```
SOUND OFF ENTER
```

Rewind the tape; then press **PLAY** on the cassette recorder and set the volume control between 4 and 6.

Before the machine language program can be loaded into the computer's memory, space must be provided. Type this statement:

```
CLEAR 11Ø, 56920 ENTER
```

Now type CLOADM ''filename'' (where filename is the particular decoder program you want). Then press ENTER. When you use a Disk/Video Interface, the decoder programs you use are B3OF9D.CO, UPCAD.CO, and PLESYD.CO. You may store all 3 decoder programs in your Model 100, or just the one you will be using.

While the computer is looking for the program on the tape, several messages appear on the screen to let you know what stage the search is in. When the program has been loaded, the **Ok** prompt appears.

For example, if you used the filename UPCAD, your screen should show:

```
Skip :   B30F9D
Found:   UPCAD
Top  :   56924
End  :   57747
Exe. :   56960
Ok
```

The number 56924 labelled ''Top'' represents the place in the Model 100 memory where the specified machine language program begins. The ''End'' number 57747 is the ending address for the specified program. The last number 56960 gives you the location at which that program begins execution.

After loading the decoder into memory from the cassette, you may save it in the directory by typing (all on one line):

```
SAVEM "filename", Top address,
End address, Execution address ENTER
```

where filename is the name of the decoder program (i.e., B3OF9D, UPCAD, or PLESYD); and top, end, and execution address are the same numbers which appeared on the display when you loaded the program from the tape. For the UPCAD decoder, the command would look like this:

```
SAVEM "UPCAD", 56924, 57747, 56960
```

The decoder program is now stored in the Model 100 and may be accessed by a simple BASIC program for reading bar codes.

Note: If your application requires the use of only one decoder, you may be able to conserve the amount of memory available to BASIC by deleting the SAVEM command. See Appendix E for details.

Using the Bar Code Decoder with BASIC

The following BASIC program illustrates how simple it is to input data into the computer with the Bar Code Reader.

Example 1:

```
10 CLEAR 100, 56920
20 RUNM "filename"
30 OPEN "WAND:" FOR INPUT AS 1
40 INPUT #1, A$
50 PRINT A$
60 IF A$<>" " THEN 40
70 CLOSE:CALL 56943
```

Line 10 clears 100 characters of string storage for the data which will be read, and also sets 56920 as the TOP address BASIC may use during program execution. (See **CLEAR** in Chapter 16 of the Model 100 owner's manual.)

Line 20 loads the decoder program ''filename'' into the correct RAM location where it may be accessed by the BASIC program. The filename is the name of the particular decoder as you stored it in RAM, i.e., B3OF9D, UPCAD, or PLESYD. (See **RUNM** in Chapter 16 of the Model 100 owner's manual.)

Line 30 opens a file for the WAND as an INPUT device with buffer file number 1. (See **OPEN** in Chapter 16 of the Model 100 owner's manual.)

Line 40 inputs the data as a character string called A$. This instruction waits to receive a string of characters from the decoder program when a successful scan of a bar code is completed. The LINE INPUT# command may also be used. (See **INPUT#** and **LINE INPUT#** in Chapter 16 of the Model 100 owner's manual.)

Lines 10 through 40 are common to all bar code BASIC programs. After executing line 40, you will have a string of characters from the bar code decoder (A$) which you may use in your program in any way you wish. Line 50 simply prints the data string on the display and, of course, line 60 loops back to wait for another scan from the Bar Code Reader unless the user has aborted by pressing (ENTER).

Important Note: The code in line 70 is very important. It is essential to execute these two statements before returning to the menu:

```
CLOSE
CALL 56948
```

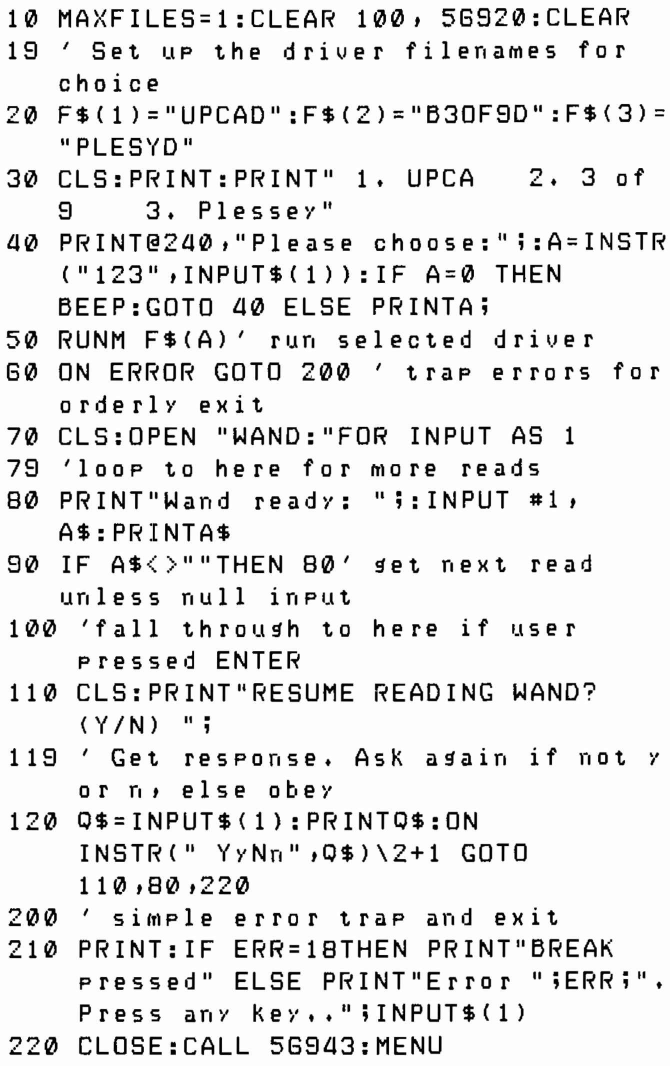
The CALL to 56948 must be preceded by closing the file with the CLOSE command. These instructions reset some of the operating system parameters and prevent an error condition.

To exit a bar code program, simply press the (ENTER) key instead of scanning a line of bar code. This will return a ''null string'' from the decoder. Your BASIC program can then test for a null string and GOTO an exit routine.

The following is an example of a BASIC program which allows you to first select which type of code you wish to read; it then reads and displays the code as it is scanned with the wand. When you have finished reading or wish to read a different type of code, simply press the (ENTER) key. The program will then prompt you for a decision as to whether or not you want to continue reading. If you press (N), it will return you to the menu; if you press (Y), it will return you to the point where you enter the type of code to be read. The (BREAK) key will also cause a return to the menu.

Note: This program is stored on your cassette tape. Its filename is "RDBCDV".

Example 2:



```
10 MAXFILES=1:CLEAR 100, 56920:CLEAR
19 ' Set up the driver filenames for
   choice
20 F$(1)="UPCAD":F$(2)="B30F9D":F$(3)=
   "PLESYD"
30 CLS:PRINT:PRINT" 1. UPCA   2. 3 of
   9    3. Plessey"
40 PRINT@240,"Please choose:";:A=INSTR
   ("123",INPUT$(1)):IF A=0 THEN
   BEEP:GOTO 40 ELSE PRINTA;
50 RUNM F$(A)' run selected driver
60 ON ERROR GOTO 200 ' trap errors for
   orderly exit
70 CLS:OPEN "WAND:"FOR INPUT AS 1
79 'loop to here for more reads
80 PRINT"Wand ready: ";:INPUT #1,
   A$:PRINTA$
90 IF A$<>""THEN 80' get next read
   unless null input
100 'fall through to here if user
    pressed ENTER
110 CLS:PRINT"RESUME READING WAND?
    (Y/N) ";
119 ' Get response. Ask again if not y
    or n, else obey
120 Q$=INPUT$(1):PRINTQ$:ON
    INSTR(" YyNn",Q$)\2+1 GOTO
    110,80,220
200 ' simple error trap and exit
210 PRINT:IF ERR=18THEN PRINT"BREAK
    pressed" ELSE PRINT"Error ";ERR;".
    Press any key..";INPUT$(1)
220 CLOSE:CALL 56943:MENU
```

Lines 20, 30 and 40 allow you to select the type of code you want to read.

Line 90 tests for a **null string**, which allows you to break out of the program with the (ENTER) key.

The **ON ERROR** command in line 60 will detect the (BREAK) key and, in this case, will return to the menu (line 210).

Appendix B/Specifications

Bar Width (narrow bar)	.012 inch (0.3 mm) minimum .017 inch (.43 mm) nominal
Scan Velocity	3-30 inches/sec.(7.6-76 cm/sec.)
Contrast	70% PCS
Reflectance of Bar	Less than 21%
Reflectance of Media	Greater than 70%
Orientation	
Tilt Angle	0° to 30°
Optimum	10° to 20°
Supply Voltage	5 VDC nominal
Maximum Current (dark bar)	50 mA

Appendix C/Sample Programs

There are two sample programs furnished with your Bar Code Reader. They are intended to be illustrative examples of how a BASIC program can be written to use the Bar Code Reader. They are not intended to be final, smooth products which cover all possibilities in an inventory system. They are usable, but meet fairly limited needs.

The two programs are:

- **INVCRE.BA**, which stands for INVentory CREate. This program is used to read in a bar code label and ask for a description. Options let you add to an existing inventory file.

- **INVENT.BA**, which stands for INVENTory system. Three options allow you to take an inventory listing, print your inventory listing or quit.

Loading the Programs

1. Use the Cursor keys to highlight BASIC and press (ENTER).

You will see:

```
TRS-8Ø Model 1ØØ Software
Copr. 1983 Microsoft
xxxx Bytes free
OK
```

The "xxxx" will vary with each Model 100 and the number of other files that are present.

2. Follow instructions given earlier to load the UPC bar code decoder into the Model 100's memory.

3. Be sure your cassette recorder is set up, connected properly to the Model 100, and load the cassette tape, if you have not already done so. Then type the following:

```
CLOAD "INVCRE" (ENTER)
```

4. After ''INVCRE'' loads, type:

```
SAVE "INVCRE" (ENTER)
```

This stores the program in the memory of the Model 100 and places the name INVCRE in the main menu.

5. Now type:

```
CLOAD "INVENT" (ENTER)
```

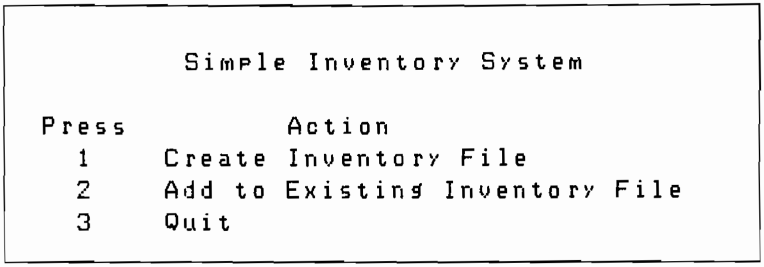
6. After ''INVENT'' loads, type:

```
SAVE "INVENT" (ENTER)
```

Now both inventory programs are stored in the Model 100 memory.

Using INVCRE

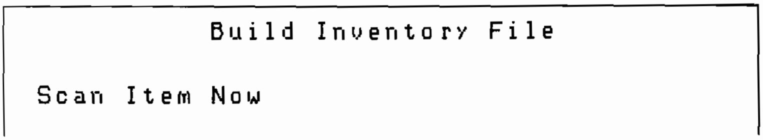
From the main menu, use the Cursor keys to position the Cursor over "INVCRE" and press ENTER. The Model 100 will BEEP and display:



```
        Simple Inventory System

Press           Action
  1     Create Inventory File
  2     Add to Existing Inventory File
  3     Quit
```

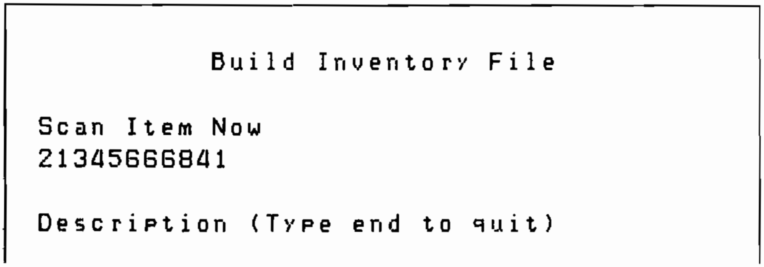
When you first begin using the program, you will need to create your file of descriptions. To do this, press the number 1 (not function key F1). You will then be told that you will erase any existing file and be given an opportunity to continue or return to the above menu. Press the letter C to continue.

You will now see:



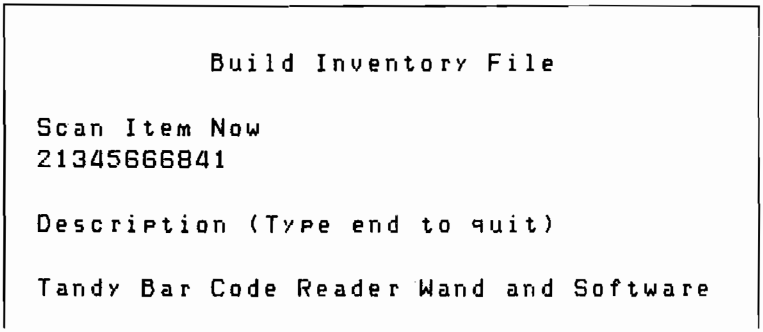
```
          Build Inventory File

Scan Item Now
```

Press the switch on the Bar Code Reader and scan your item's bar code. You will see the actual code that was read displayed beneath. Then the description is asked for:



```
            Build Inventory File

Scan Item Now
21345666841

Description (Type end to quit)
```

Simply type in the description you want associated with the bar code read. You should limit your description to one line (40 characters or fewer) to conserve memory space.

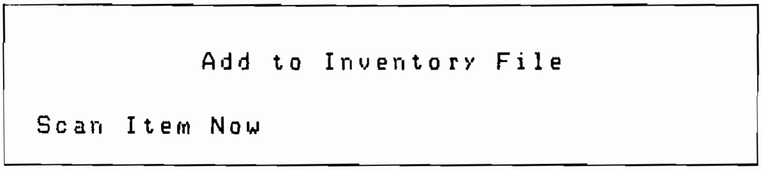


```
            Build Inventory File

Scan Item Now
21345666841

Description (Type end to quit)

Tandy Bar Code Reader Wand and Software
```

The program will now clear the screen and repeat the "Scan Item Now" prompt. When you have finished scanning all items, simply scan any item one more time and, instead of typing in a description, simply type the word "end" (in either upper or lower case letters). You will then be returned to the main menu.

Adding to an Existing Inventory File

From the main menu, press 2 (number key, not function key F2) to add to an existing inventory file. You will see:



```
            Add to Inventory File

Scan Item Now
```

It operates just the same as selection 1, Create Inventory File. Remember to type **end** for description when you are finished.

Saving Your Inventory File

Since you have built your inventory file with bar code numbers and associated descriptions, you will want to save it on cassette tape for future reference: The Simple Inventory System does not have any provision for this because the Model 100 already contains a method for saving data.

1. From the opening menu of SIS, press 3 (number key, not function key) and you will return to the main menu of the Model 100. You will notice a new file **INVTRY.DO**. This is your list of stock numbers (bar codes) and matching descriptions. Use the Cursor keys to position the Cursor over INVTRY.DO and press **ENTER**. You will then be in the TEXT program.

2. Press the F3 function key. On the last line of the display you will see:

```
3456789,item description
1234991,item number 2
9876543,item number 3
5555555,item number 4
4444444,item number 5
1233333,item number 6
1234456,item number 7
Save to:
```

You should have your tape recorder connected to the Model 100 with a good quality cassette tape loaded and set for **RECORD**. Be careful not to record over your program tape. (Refer to your Model 100 owner's manual if you need specific instructions on the connections.)

Type INVTRY ENTER and your recorder will begin saving the inventory file onto cassette tape for future use. When complete, press F8 to return to the main menu of the Model 100.

At any future time, you may re-load the inventory file from cassette tape by following the instructions given in AppendixA of your Model 100 owner's manual. (Refer to the section called "Loading Text Files.") Briefly, all you need to do is position the Cursor over **TEXT** and press ENTER. When asked for a file name, type **INVTRY. This is very important — be sure you type the name exactly!**) Then press ENTER. With your recorder connected and the proper tape file (INVTRY) loaded, press F3 and you will be prompted by the statement:

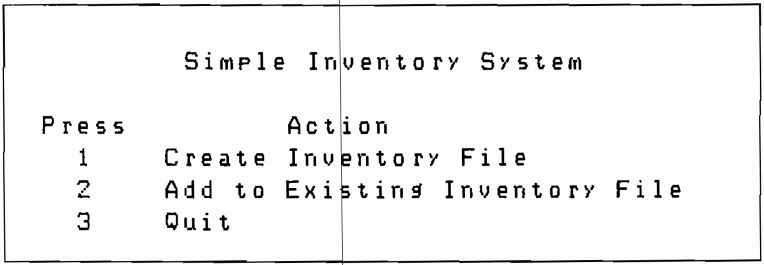
```
Load from:
```

Type INVTRY ENTER. This will load your inventory file into the Model 100's memory. When loading is complete, press F8 to return to the main menu.

Taking Inventory

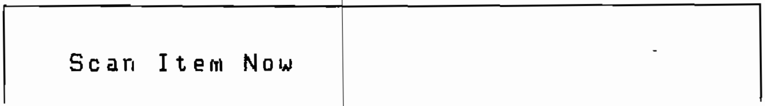
1. Position the Cursor over "INVENT.BA" and press ENTER.

2. The Model 100 will BEEP and display:



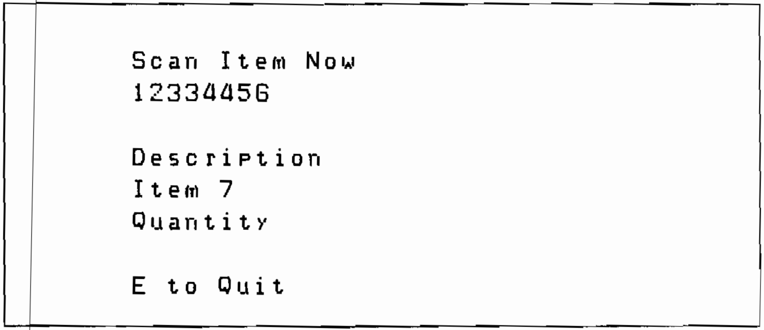
```
          Simple Inventory System

Press           Action
  1     Create Inventory File
  2     Add to Existing Inventory File
  3     Quit
```

3. Press 1 (number, not function key) and you will see:



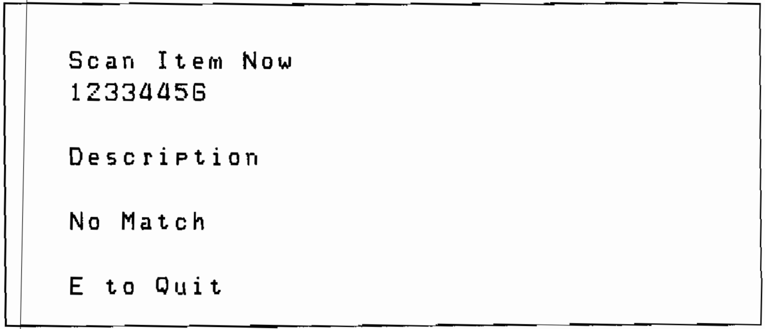
```
   Scan Item Now
```

4. Press the push-to-read switch on the Bar Code Reader and scan the item. When a bar code has been successfully read, you will see:



```
Scan Item Now
12334456

Description
Item 7
Quantity

E to Quit
```

If no match is found in your inventory file, the Model 100 will BEEP and display:



```
Scan Item Now
12334456

Description

No Match

E to Quit
```

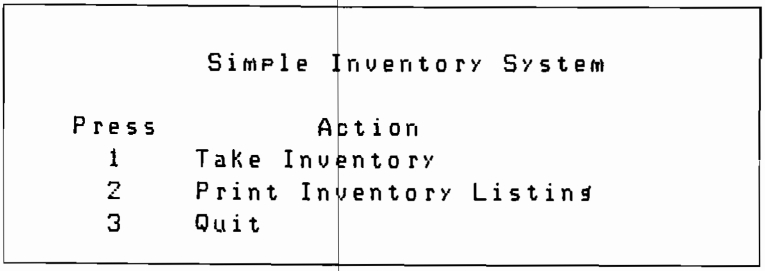
and then return to the ''Scan Item Now'' prompt.

5. Input your quantity and press ENTER after you complete your entry. The Model 100 will return to the ''Scan Item Now'' prompt.

6. When you have completed your inventory, scan any item one more time and answer ''e'' to the quantity. This last entry will not be saved in your count file; it is used only to signal completion of the program.

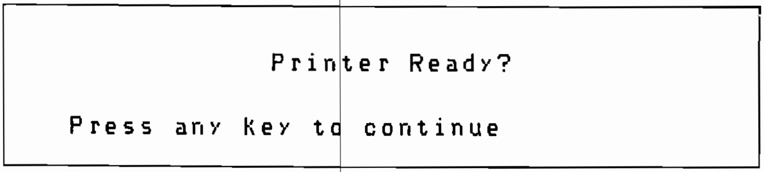
Printing Your Inventory

From the SIS main menu:



```
              Simple Inventory System

   Press              Action
     1      Take Inventory
     2      Print Inventory Listing
     3      Quit
```

press 3 (number, not function key). You will see:



```
                   Printer Ready?

   Press any key to continue
```

Be sure that your printer is connected and that paper is properly loaded. The SIS will print the following:

```
            Simple Inventory System Listing

Item            Description                    Quantity

3456789         item description                     10
1234991         item number 2                         5
9876543         item number 3                         8
5555555         item number 4                        25
4444444         item number 5                        15
1233333         item number 6                         8
1234456         item number 7                         9
```

Notice that this is a very simple system. It does not carry forward into a data base any of the quantities that you have entered. It can be used either as an inventory taking system or as an inventory ordering system; however, it does not maintain any type of perpetual inventory system.

Program Listing INVCRE.BA

```
10 CLS
20 MAXFILES=2:CLEAR 500,61788
30 RUNM"UPC"
40 DEFSTR C,D,E,K
50 E=CHR$(27)
60 CLS:BEEP:PRINT@9,E;"p";" Simple
   Inventory System ";E;"q"
70 PRINT@83,E;"p";" Press ";:PRINT@95,"
   Action ";E;"q"
80 PRINT@125," 1...Create Inventory File"
90 PRINT@165," 2...Add to Existing
   Inventory File"
100 PRINT@206,"3...Quit"
110 K=INKEY$:IFLEN(K)=0THEN110
120 A=ASC(K):A=A-48:IFA<1ORA>3 THEN
    BEEP:GOTO 110
130 ONAGOTO150,290,140
140 CLOSE:CALL 61807:MENU
150 CLS
160 PRINT@82,"This will erase any existing
    inventory"
170 PRINT@122,"file that is present."
180 PRINT@206,E;"p";" Press Letter C to
    continue"
190 PRINT@283," Press Letter M to return to
    MENU ";E;"q";
200 K=INKEY$:IFLEN(K)=0THEN200
210 IFK="M"ORK="m"THENGOTO60
```

```
220 IFK="C"ORK="c"THEN 230 ELSEBEEP:GOTO200
230 '
240 OPEN "WAND:" FOR INPUT AS 1
250 OPEN "INVTRY.DO" FOR OUTPUT AS 2
260 CLS:PRINT@9,E;"p";" Build Inventory
    File ";E;"q"
270 GOSUB 350
280 CLOSE:GOTO60
290 CLS
300 OPEN "WAND:" FOR INPUT AS 1
310 OPEN "INVTRY.DO" FOR APPEND AS 2
320 PRINT@9,E;"p";" Add to Inventory File
    ";E;"q"
330 GOSUB 350
340 CLOSE:GOTO60
350 PRINT@83,E;"p";" Scan Item Now ";E;"q"
360 LINE INPUT#1,C
370 PRINT@123,C
380 PRINT@203,E;"p";" Description (Type end
    to quit) ";E;"q"
390 LINE INPUT D
400 IF D="END" OR D="end" THEN RETURN
410 PRINT#2,C;",";D
420 PRINT@120,E;"K":PRINT@240,E;"K"
430 GOTO 360
```

The following is a line-by-line description of the INVCRE.BA program:

Program Line	Comment
10	Clears the screen
20	Clears 500 bytes for string space and reserves memory beginning at 61788 for machine language driver

Program Line	Comment
30	Loads and executes UPC bar code driver machine language program
40	Defines as string variables the variable names C, D, E and K
50	Defines variable E to be the ESC character
60-100	Clears screen and presents main SIS menu
110	Inkey routine to get 1 character from keyboard
120	Checks that key received is numeric and in range from 1 to 3; if not, beeps and goes to line 110
130	Dispatch line to go to section of program depending on answer of 1 to 3 1 goto line 150 2 goto line 290 3 goto line 140
140	Closes files, resets bcr driver to null device and returns to Model 100 main menu
150	Clears the screen
160-190	Screen to warn that continuing will erase inventory file
200	Inkey routine to get 1 character from keyboard
210-220	Checks for continue or menu answer
230	Remark or dummy line
240	Sets up bar code wand as device 1
250	Sets up a file named INVTRY.DO as device 2
260	Clears screen and prints heading
270	Goes to subroutine at line 350

Program Line	Comment
280	Closes devices 1 and 2 and goes to line 60
290	Clears the Screen
300	Same as line 240
310	Sets up INVTRY.DO file so that subsequent input will be added to the end of the file
320	Prints heading
330	Goes to subroutine at line 350
340	Closes devices 1 and 2 and goes to line 60
350	Prints heading
360	Accepts input from bar code wand
370	Prints code read from bar code reader
380	Prints description with exit reminder
390	Accepts input from keyboard for the description
400	Checks if description entered in line 390 is the word end — if so, exits the subroutine; if not, continues to line 410
410	Stores decoded bar code input and description to device 2 (INVTRY.DO file)
420	Clears 2 lines (bar code and description) from LCD
430	Loops back to line 360 to read another value

Program Listing INVENT.BA

```
10 CLS
20 CLEAR 500,61788
30 RUNM"UPC"
40 DEFSTR C,D,E,K
50 E=CHR$(27)
60 CLS:BEEP:PRINT@9,E;"p";" Simple
    Inventory System ";E;"q"
70 PRINT@83,E;"p";" Press ";:PRINT@95,"
    Action ";E;"q"
80 PRINT@125," 1...Take Inventory"
90 PRINT@165," 2...Print Inventory Listing"
100 PRINT@206,"3...Quit"
110 K=INKEY$:IFLEN(K)=0THEN110
120 A=ASC(K):A=A-48:IFA<1ORA>3 THEN
    BEEP:GOTO 110
130 ONAGOTO150,420,140
140 CLOSE:CALL 61807:MENU
150 CLS
160 OPEN "WAND:" FOR INPUT AS 1
170 OPEN "INVTRY.DO" FOR INPUT AS 2
180 N=1
190 LINE INPUT #2, I$
200 IF EOF(2) THEN 210 ELSE N=N+1:GOTO 190
210 CLOSE #2:DIMI$(N),Q(N)
220 OPEN"INVTRY.DO" FOR INPUT AS 2
230 FOR X=1 TO N
240 LINE INPUT#2,I$(X):Q(X)=0
250 NEXT X
260 CLOSE 2
270 PRINT@43,E;"p";" Scan Item Now ";E;"q"
280 LINE INPUT #1,C
290 PRINT@83,C
300 PRINT@283,E;"p";" E to Quit ";E;"q";
310 PRINT@163,E;"p";" Description: ";E;"q"
320 F=0
```

```
330 FOR X=1TON
340 IFLEFT$(I$(X),LEN(C))=CTHENP=X:X=N:F=1
350 NEXT X
360 IF F=0THENSOUND 11000,50:PRINTTAB(4);
    "No Match Found":FORQ=1TO400:NEXTQ:
    PRINT@80,E;"K":PRINT@200,E;"K":
    GOTO 270
370 PRINTTAB(5);RIGHT$(I$(P),(LEN(I$(P))
    -LEN(C))-1)
380 PRINT@243,E;"p";" Quantity ";E;"q";
390 INPUT K:IFK="e"ORK="E"THENGOTO60
400 Q(P)=VAL(K)
410 PRINT@80,E;"K":PRINT@200,E;"J":
    GOTO 270
420 CLS:PRINT@51,E;"p";" Printer Ready?
    ";:PRINT@126," Press any Key to
    continue ";E;"q"
430 K=INKEY$:IFK=""THEN430
440 LPRINT TAB(25);"Simple Inventory System
    Listing"
450 LPRINT:LPRINT
460 LPRINT "Item";TAB(25);"Description";
    TAB(65);"Quantity"
470 LPRINT
480 FOR X=1 TO N
490 K=I$(X):J=INSTR(1,K,",")
500 LPRINT LEFT$(K,J-1);TAB(25);
    RIGHT$(K,LEN(K)-J);TAB(65);Q(X)
510 NEXT X
520 GOTO 60
```

The following is a line-by-line description of the INVENT.BA program:

Program Line	Comment
10	Clears the screen
20	Clears 500 bytes for string space and reserves memory beginning at 61788 for machine language driver
30	Loads and executes UPC bar code driver machine language program
40	Defines as string variables the variable names C, D, E and K
50	Defines variable E to be the ESC character
60-100	Clears screen and presents main SIS menu
110	Inkey routine to get 1 character from keyboard
120	Checks that key received is numeric and in range from 1 to 3; if not, beeps and goes to line 110
130	Dispatch line to go to section of program depending on answer of 1 to 3 1 goto line 150 2 goto line 420 3 goto line 140
140	Closes files, resets bcr driver to null device and returns to Model 100 main menu
150	Clears the screen
160	This line sets up bar code wand as device 1
170	This line sets up file INVENT.DO as device 2
180	Sets a counter to 1
190	Reads in one line of inventory from file INVENT.DO

Program Line	Comment
200	If the end of the file has been reached, go to line 210; else increment the counter by one and go read another line at 190
210	Closes file 2 (INVENT.DO) and dimension 2 arrays to the exact number of elements needed as determined by the counter
220	Sets up file INVENT.DO as device 2 again. It is necessary to reopen it because reading all the elements in lines 190-200 caused the file pointer to be at the end of the file. The only way to go back to the beginning of the file is to close it and then re-open it again
230-250	Loop to read the exact number of inventory items into an array
260	Closes file INVENT.DO since all the data is now in a memory array
270	Prints message header
280	Accepts input from bar code wand
290	Prints bar code read
300	Prints exit reminder
310	Prints description prompt
320	Sets found variable to null
330-350	Searches in memory inventory file for match to code read by wand. If match is found, sets variable F=1 to indicate success
360	If no match (F=0), then say so and go back to line 270

Program Line	Comment
370	Match was found; print the description under heading of description
380	Prompt for quantity
390	Accepts quantity as a string variable. Check for end. If end is desired, goto 60; if not, then go to next line
400	Changes string variable to numeric variable
410	Clears 2 lines (code read and description) from LCD
420	Clears the screen and reminds you to connect the printer
430	Inkey routine to get any keypress in order to continue
440-470	Prints headings on report
480-510	Loop to go through entire inventory listing
520	Goes back to main SIS menu

Appendix D/Description of Bar Codes

The three decoders provided in this package will cover a large number of applications in a variety of industries. They all may be scanned bidirectionally and offer a high degree of reliability. A brief description of each code follows.

Plessey

Plessey Code is a binary code in which each character consists of 4 bars and the adjacent 4 spaces. Each bar/space pair contains one information bit. The particular version of the Plessey family which this decoder recognizes is the MSI Code. In the MSI Code, the zero bit is a one-unit bar followed by a two-unit space, and the one bit is a two-unit bar followed by a one-unit space. Complete four-bit characters are thus 12 units wide.

Your Plessey decoder reads an MSI format which consists of a start character, 7 data characters, a 4-bit check character, a termination bar, and a reverse start character. The code may be recognized by the start character at the beginning of the code (a two-unit bar followed by a one-unit space) and the termination bar and reverse start character at the end of the code (a one-unit bar followed by a two-unit space and a one-unit bar). (See Figure 1.)

The 4-bit check character is calculated by different methods for different variations of this code. In order to make the decoder more versatile, we have not included the checksum calculation in the decoder program. The check character is, instead, returned to the BASIC program as the last character in the string and must be verified by the BASIC program.

Note: The decoder program will return a string of data with a **BEEP** when it recognizes the start character at the beginning and the termination bar and reverse start character at the end. If there is a mis-read of the data in between, it will not know it. Therefore, a BASIC routine must be used to calculate the check character and produce a low-frequency BEEP (or another indication, if you so desire) if the check character indicates a bad read.

3 of 9

3 of 9 code is the most widely used industrial bar code. It can easily be printed on a dot matrix printer, such as the Radio Shack DMP-200, DMP-400, DMP-500, etc.

Each character of this code is derived from 5 bars and the included 4 spaces for a total of 9 elements. The code is so named because 3 of the 9 elements are wide while the other 6 are narrow. A zero bit is a one-unit bar or space and a one bit is a three-unit bar or space. This arrangement provides a strong self-checking property such that a high level of data reliability is built in without the use of a checksum character. The checksum is optional in 3 of 9 and, if it is used, it will simply be returned to the BASIC program at the end of the data string. The BASIC program would then have to verify the checksum and provide an audible alarm in case of an error.

3 of 9 code may be recognized by the start character at the beginning (an asterisk coded as 010010100) and the same character at the end.

Your 3 of 9 decoder will recognize the standard 43-character set shown in Figure 8. The full ASCII character set may be decoded by using the standard characters preceded by one of the special characters $, /, + or %. These 2 character pairs may be decoded by a BASIC subroutine which converts them to the appropriate characters. This allows all 128 ASCII characters to be decoded. The full ASCII set and a subroutine to decode the full ASCII set are shown in Appendix F.

Note: A maximum of 23 characters per line of bar code may be decoded with the Model 100 decoder.

CHAR.	PATTERN	BARS	SPACES
1	▬ ■ ■ ■ ▬	10001	0100
2	■ ▬ ■ ■ ▬	01001	0100
3	▬ ▬ ■ ■ ■	11000	0100
4	■ ■ ▬ ■ ▬	00101	0100
5	▬ ■ ▬ ■ ■	10100	0100
6	■ ▬ ▬ ■ ■	01100	0100
7	■ ■ ■ ▬ ▬	00011	0100
8	▬ ■ ■ ▬ ■	10010	0100
9	■ ▬ ■ ▬ ■	01010	0100
0	■ ■ ▬ ▬ ■	00110	0100
A	▬ ■ ■ ■ ▬	10001	0010
B	■ ▬ ■ ■ ▬	01001	0010
C	▬ ▬ ■ ■ ■	11000	0010
D	■ ■ ▬ ■ ▬	00101	0010
E	▬ ■ ▬ ■ ■	10100	0010
F	■ ▬ ▬ ■ ■	01100	0010
G	■ ■ ■ ▬ ▬	00011	0010
H	▬ ■ ■ ▬ ■	10010	0010
I	■ ▬ ■ ▬ ■	01010	0010
J	■ ■ ▬ ▬ ■	00110	0010
K	▬ ■ ■ ■ ▬	10001	0001
L	■ ▬ ■ ■ ▬	01001	0001
M	▬ ▬ ■ ■ ■	11000	0001
N	■ ■ ▬ ■ ▬	00101	0001
O	▬ ■ ▬ ■ ■	10100	0001
P	■ ▬ ▬ ■ ■	01100	0001
Q	■ ■ ■ ▬ ▬	00011	0001
R	▬ ■ ■ ▬ ■	10010	0001
S	■ ▬ ■ ▬ ■	01010	0001
T	■ ■ ▬ ▬ ■	00110	0001
U	▬ ■ ■ ■ ▬	10001	1000
V	■ ▬ ■ ■ ▬	01001	1000
W	▬ ▬ ■ ■ ■	11000	1000

CHAR.	PATTERN	BARS	SPACES
X	■ ■ ▬ ■ ▬	00101	1000
Y	▬ ■ ▬ ■ ■	10100	1000
Z	■ ▬ ▬ ■ ■	01100	1000
-	■ ■ ■ ▬ ▬	00011	1000
•	▬ ■ ■ ▬ ■	10010	1000
SPACE	■ ▬ ■ ▬ ■	01010	1000
*	■ ■ ▬ ▬ ■	00110	1000
$	■ ■ ■ ■ ■	00000	1110
/	■ ■ ■ ■ ■	00000	1101
+	■ ■ ■ ■ ■	00000	1011
%	■ ■ ■ ■ ■	00000	0111

The * symbol denotes a unique start/stop character which must be the first and last character of every bar code symbol.

Figure 8. Standard ASCII Set

UPC

UPC (Universal Product Code) is the industry standard for the retail goods industry. It is used by the supermarket industry to facilitate automatic scanning of item numbers with associated price look-up at the point of sale. Its use is also spreading to other types of retail establishments.

There are several versions of the UPC code; however, your Model 100 decoder recognizes only the most popular version, UPC-A. An example of a UPC-A symbol is shown in Figure 1. This code is easily recognizable by the left, center, and right guard bands which are longer than the other bars. The UPC-A decoder returns a string of 11 numeric characters. The first character is the industry code and is printed to the left of the bar code symbol. The other 10 characters are normally printed beneath the bar code.

UPC provides two levels of checking, which result in an extremely high reliability rate. The coding technique for UPC is very complex and will not be described in detail in this manual.

Appendix E/Cassette Load and Directory Operation

When a program is loaded from cassette, it is stored in memory at the location where it is to be executed. It will remain at that location until it is replaced by another program or data. If you store the program in the directory with a **SAVE** or **SAVEM** command, the program will be stored in a separate buffer area in memory where it may be recalled at any time with a **LOAD** or **LOADM** command. If your application requires only one bar code decoder program and/or one BASIC application program, then you may conserve buffer memory by not **SAVE**ing the program(s) to the directory. You should, of course, always make a backup copy of your program(s) on cassette tape in case of system failure.

When a Bar Code Driver (or any other machine code program) is loaded into memory, specific memory addresses are reserved for the program's use. This reserved area remains even if the program is subsequently deleted from memory. To regain use of the reserved memory after deleting a machine code program, type:

```
    CLEAR 256,MAXRAM (ENTER)
```

Appendix F/Encoding the Full ASCII Character Set with 3 of 9

ASCII	CODE 3 of 9	ASCII	CODE 3 of 9
NUL	%U	SP	Space
SOH	$A	!	/A
STX	$B	"	/B
ETX	$C	#	/C
EOT	$D	$	/D
ENQ	$E	%	/E
ACK	$F	&	/F
BEL	$G	'	/G
BS	$H	(	/H
HT	$I	)	/I
LF	$J	*	/J
VT	$K	+	/K
FF	$L	,	/L
CR	$M	—	—
SO	$N	•	•
SI	$O	/	/O
DLE	$P	0	0
DC1	$Q	1	1
DC2	$R	2	2
DC3	$S	3	3
DC4	$T	4	4
NAK	$U	5	5
SYN	$V	6	6
ETB	$W	7	7
CAN	$X	8	8
EM	$Y	9	9
SUB	$Z	:	/Z
ESC	%A	;	%F

ASCII	CODE 3 of 9	ASCII	CODE 3 of 9
FS	%B	<	%G
GS	%C	=	%H
RS	%D	>	%I
US	%E	?	%J
@	%V	`	%W
A	A	a	+A
B	B	b	+B
C	C	c	+C
D	D	d	+D
E	E	e	+E
F	F	f	+F
G	G	g	+G
H	H	h	+H
I	I	i	+I
J	J	j	+J
K	K	k	+K
L	L	l	+L
M	M	m	+M
N	N	n	+N
O	O	o	+O
P	P	p	+P
Q	Q	q	+Q
R	R	r	+R
S	S	s	+S
T	T	t	+T
U	U	u	+U
V	V	v	+V
W	W	w	+W
X	X	x	+X
Y	Y	y	+Y
Z	Z	z	+Z

ASCII	CODE 3 of 9	ASCII	CODE 3 of 9
[	%K	{	%P
\	%L	¦	%Q
]	%M	}	%R
	%N	~	%S
_	%O	DEL	%T

Note: Character pairs /M and /N decode as a minus sign and a period, respectively. Character pairs /P through /Y decode as 0 through 9.

Full ASCII Decoder Subroutine

```
10 LINEINPUT#1,A$
20 IF A$=""THEN 190
30 L=LEN(A$):P=1
40 V=ASC(MID$(A$,P,1))
50 IFV=36THENOS=64:GOTO100
60 IFV=37THENOS=11:GOTO120
70 IFV=47THENOS=932:GOTO100
80 IFV=43THENOS=-32:GOTO100
90 P=P+1:IFP>LTHEN PRINTA$:GOTO10 ELSE 40
100 MID$(A$,P)=MID$(A$,P+1):L=L-1:
    A$=LEFT$ (A$,L)
110 B$=MID$(A$,P,1):B=ASC(B$):B=B-OS:
    B$=CHR$(B):MID$(A$,P,1)=B$:GOTO90
120 T=ASC(MID$(A$,P+1,1))
130 IF T<70 THEN OS=38:GOTO100
140 IF T<75 THEN OS=11:GOTO100
150 IF T<80 THEN OS=-16:GOTO100
160 IF T<85 THEN OS=-43:GOTO100
170 IF T=85 THEN OS=85:GOTO100
180 IF T=86 THEN OS=22:GOTO100 ELSE OS=-9:
    GOTO100
190 CLOSE:CALL61807:GOTO<EXITROUTINE>
```

Line 10 is the input command which is common to all bar code programs. This line assigns the label A$ to the string of data from the decoder.

Line 90 defines what you do with the fully decoded string of data (A$). In this example, line 90 simply prints the data string (A$) on the display; then it returns to line 10 of the program to read another line of bar code. This program may be modified to suit your application.

Lines 20 and 190 are used to exit the program when the ENTER key is pressed.

SERVICE POLICY

Radio Shack's nationwide network of service facilities provides quick, convenient, and reliable repair services for all of its computer products, in most instances. Warranty service will be performed in accordance with Radio Shack's Limited Warranty. Non-warranty service will be provided at reasonable parts and labor costs.

Because of the sensitivity of computer products, and the problems which can result from improper servicing, the following limitations also apply to the services offered by Radio Shack:

1. If any of the warranty seals on any Radio Shack and Tandy computer products it sells are broken, Radio Shack reserves the right to refuse to service the product or to void any remaining warranty on the product.
2. If any computer product purchased from Radio Shack has been modified so that it is not within manufacturer's specifications, including, but not limited to, the installation of any non-Radio Shack parts, components, or replacement boards, then Radio Shack reserves the right to refuse to service the product, void any remaining warranty, remove and replace any non-Radio Shack part found in the product, and perform whatever modifications are necessary to return the product to original factory manufacturer's specifications.
3. The cost for the labor and parts required to return the computer product to original manufacturer's specifications will be charged to the customer in addition to the normal repair charge.

RADIO SHACK, A DIVISION OF TANDY CORPORATION

U.S.A.: FORT WORTH, TEXAS 76102
CANADA: BARRIE, ONTARIO L4M 4W5

TANDY CORPORATION

AUSTRALIA

91 KURRAJONG AVENUE
MOUNT DRUITT, N.S.W. 2770

BELGIUM

PARC INDUSTRIEL
5140 NANINNE (NAMUR)

U. K.

BILSTON ROAD WEDNESBURY
WEST MIDLANDS WS10 7JN

11/84-TM

Printed in U.S.A.