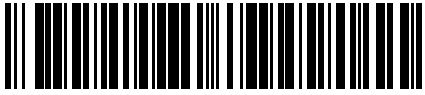
symbol®

# LS 6800


## Product Reference Guide

# LS 6800 Product Reference Guide



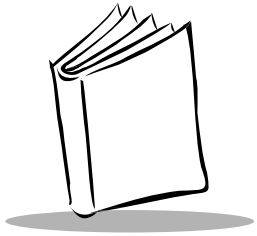
**70-35933-01**
**Revision A — August 1998**

Symbol Technologies, Inc. One Symbol Plaza, Holtsville N.Y. 11742

# *LS 6800 Product Reference Guide*

*70-35933-01*
*Revision A*
*August 1998*

Symbol Technologies, Inc.
One Symbol Plaza
Holtsville, New York 11742-1300
http://www.symbol.com



# Contents

## About This Manual



## Chapter 1. The LS 6800 Series Scanner



## Chapter 2. Setup



## Chapter 3. Scanning with the LS 6800

## Chapter 4. Maintenance and Specifications



## Chapter 5. Programming the LS 6800

## Chapter 6. Advanced Data Formatting

## Chapter 7. ADF Bar Codes

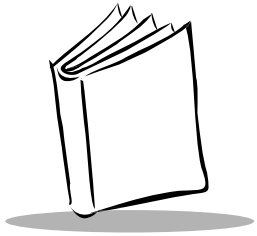


## Appendix A. Programming Reference



## Glossary

## Index



# About This Manual

The *LS 6800 Product Reference Guide* provides general instructions for setup, operation, troubleshooting, maintenance, and programming.

## Notational Conventions

The following conventions are used in this document:

- Bullets (•) indicate:
  - action items
  - lists of alternatives
  - lists of required steps that are not necessarily sequential
- Sequential lists (e.g., those that describe step-by-step procedures) appear as numbered lists.

## Related Publications

- *LS 6800 Quick Reference Guide*, p/n 70-17124-xx
- *LS 4800/4900 and LS 6800 Flash Memory Programming*, p/n 70-17636-xx

## Service Information

If you have a problem with your equipment, contact the Symbol Support Center. Before calling, have the model number, serial number, and several of your bar code symbols at hand.

Call the Support Center from a phone near the scanning equipment so that the service person can try to talk you through your problem. If the equipment is found to be working properly and the problem is symbol readability, the Support Center will request samples of your bar codes for analysis at our plant.

If your problem cannot be solved over the phone, you may need to return your equipment for servicing. If that is necessary, you will be given specific directions.

---

**Note:** *Symbol Technologies is not responsible for any damages incurred during shipment if the approved shipping container is not used. Shipping the units improperly can possibly void the warranty. If the original shipping container was not kept, contact Symbol to have another sent to you.*

---

## *Symbol Support Center*

In the U.S.A, for service information, warranty information or technical assistance, call:

SYMBOL SUPPORT CENTER
**1-800-653-5350**

If you purchased your Symbol product from a Symbol Business Partner, contact that Business Partner for service.

### Canada

Mississauga, Ontario
Canadian Headquarters
(905) 629-7226

### Europe

Wokingham, England
European Headquarters
0734-771-222 (Inside UK)
+441-734-771222 (Outside UK)

### Asia

Singapore
Symbol Technologies Asia, Inc.
337-6588 (Inside Singapore)
+65-337-6588 (Outside Singapore)

# Chapter 1
# The LS 6800 Series Scanner

The LS 6800 hands-free scanner provides a raster scanning pattern specifically designed to scan PDF417 (two dimensional) bar codes, as well as raster and linear patterns for decoding standard 1-D bar codes. Its flexible mounting arrangement provides for both attended and unattended operation.

The LS 6800 is based on the Visible Laser Diode, and can read color bar codes and symbols printed on virtually any substrates. There are two LED indicators: red indicates laser on and alignment of the scanner with the bar code, and green indicates a successful decode.

There are several ways to activate the scanner: hardwired trigger, photo-sensor, footswitch, RS-232C command, an external object sensor, continuous, or blinking laser mode. The scanner indicates a successful decode through a beep and decode LED.

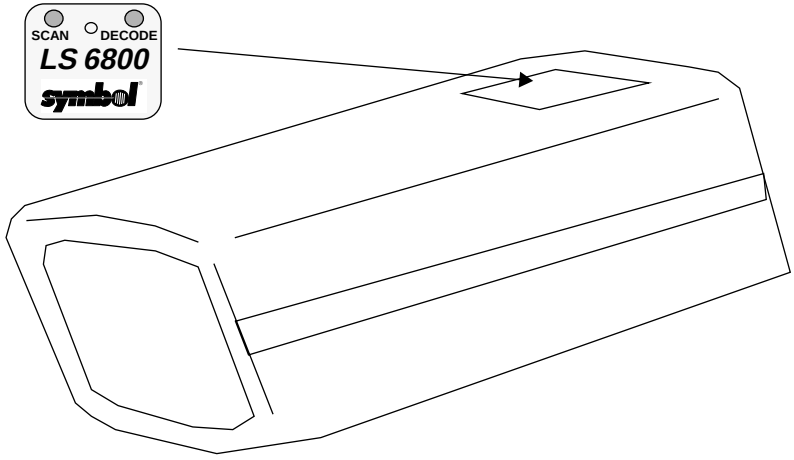



**Figure 1-1. The LS 6800 Scanner**

# Decode Capability

The LS 6800's internal decoder decodes the following symbologies:

- 2-D Symbologies: PDF417, MicroPDF417
- 1-D Symbologies: UPC-A, UPC-E, EAN-8, EAN-13, Code 39, Code 39 Full ASCII, Trioptic Code 39, Code 128, Code 93, Interleaved 2 of 5, Discrete 2 of 5, Codabar, UCC/EAN 128, Bookland EAN, UPC/EAN Coupon Code, MSI Plessey
- Other codes may be added on request (e.g., Code 16K, Code 49, Code 11, Code 32).

# Host Interface

Host interface consists of an appropriate RS-232C cable or Synapse™ "Smart Cable" set.

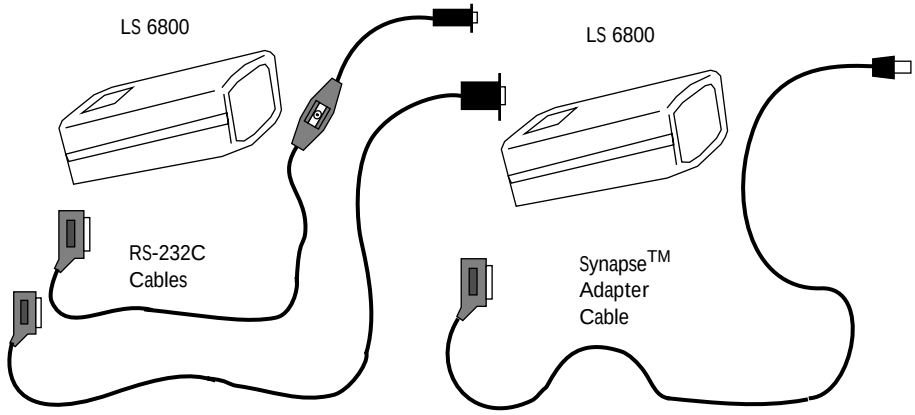
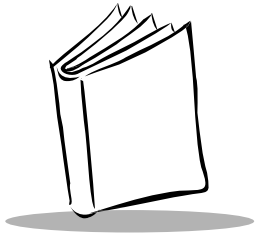



**Figure 1-2. LS 6800 with RS-232C and Synapse Cables**

# Mounting Fixtures

Through use of the proper mounting fixture, the LS 6800 may be mounted on a wall, desk, or other surface. These are discussed in the next chapter.



# Chapter 2
# Setup

## Unpacking

Remove the equipment from its packing and inspect it for damage. If any equipment was damaged in transit, call Service at one of the telephone numbers from *Symbol Support Center* on page x. **KEEP THE PACKING**. It is the approved shipping container and should be used if you ever need to return your equipment for servicing.

# Mounting the LS 6800

## *Desk Mount*

Switch off all devices to be connected to the LS 6800 scanning system, and mount the scanner using the stand with a stable base and a telescoping arm for adjustable height, as in Figure 2-1

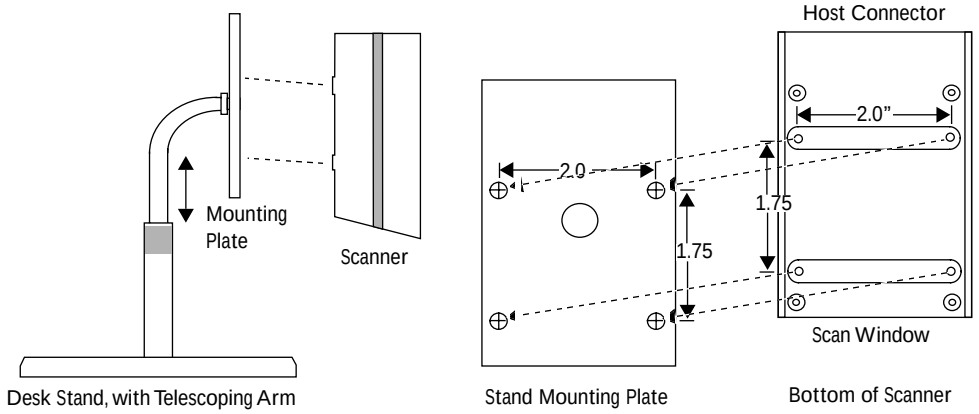



**Figure 2-1. Desk Mount Stand**

1. Attach the plate to the stand’s telescoping arm. The plate’s center hole screws onto the stand’s arm.
2. To mount the scanner to the stand’s mounting plate, align the screw holes in the plate with those on the bottom of the scanner (4x #6-32 tap) and insert appropriate screws.
3. Set the stand in the desired location.

## Wall Mount

If the scanner is to be mounted on the wall, use the wall-mount bracket.

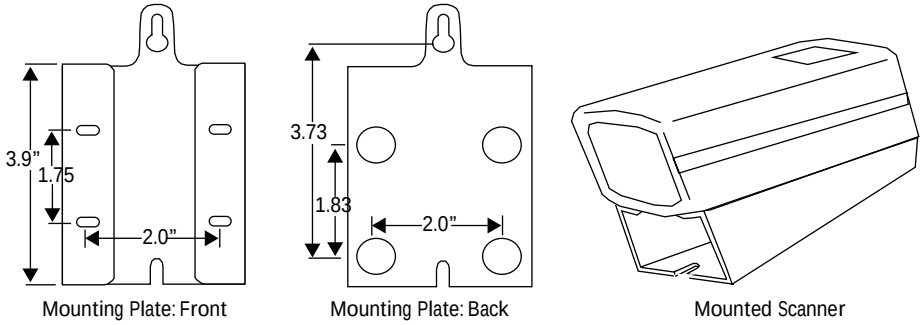



**Figure 2-2. Wall Mount Plate**

Insert appropriate screws or bolts into the screw holes or bolt holes on the bottom plate of the fixture, and attach to the wall.

# RS-232C Connection

The following instructions describe how to connect to a host via RS-232.

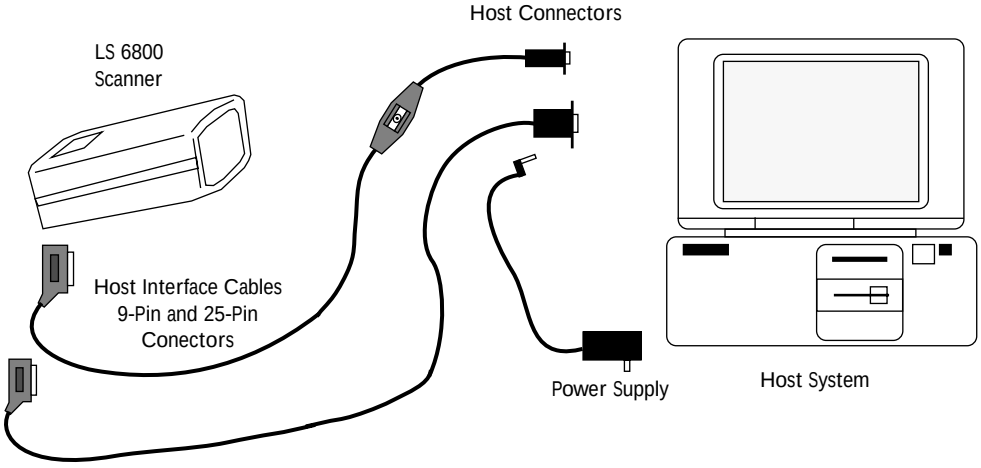



**Figure 2-3. LS 6800 RS-232C Operation**

1. Connect the host interface cable to the port at the rear of the scanner's casing, as indicated in Figure 2-4

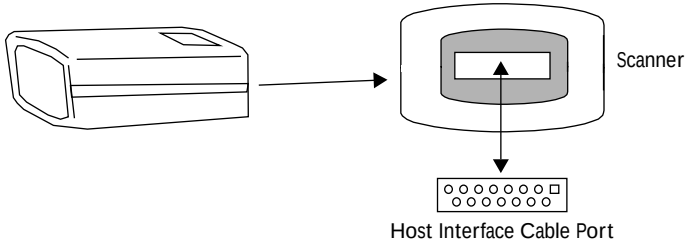



**Figure 2-4. Host Interface Cable to Scanner Port**

2. Connect the power supply to the host interface cable's power input port. For 25-pin connectors, this port is on the side of the host interface cable's host connector; for 9-

pin connectors, the port is either on the side of the host connector or on a power adapter on the cable itself.

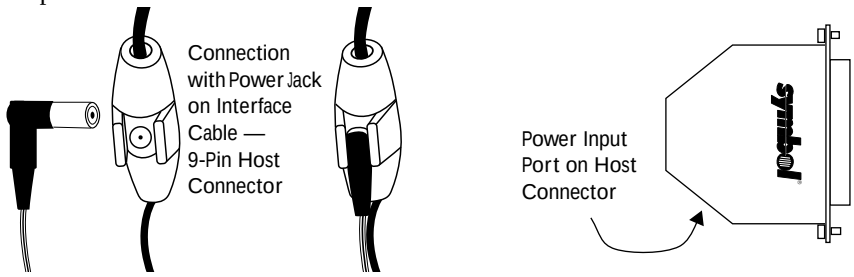



**Figure 2-5. Connecting Power Supply**

3. Connect the other end of the power supply to a receptacle supplying AC power of the proper voltage level.

   Power also may be supplied by the host system. Contact Symbol's Custom Products Group for more information.
4. Plug the host interface cable's host connector into the receiving port on the host system.
5. Program all desirable host interface, decode, and communications parameters. See Chapter 5, *Programming the LS 6800* for bar code menus.
6. Power up the host.

# Synapse™ Cable Connection

The following steps provide Synapse connection instructions.

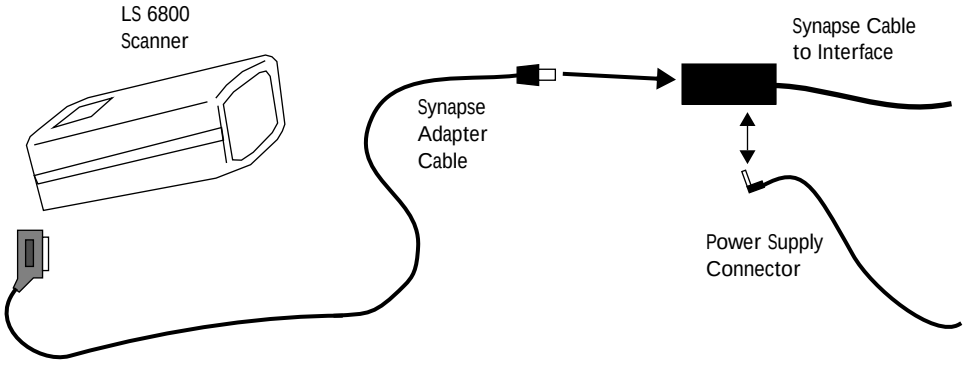
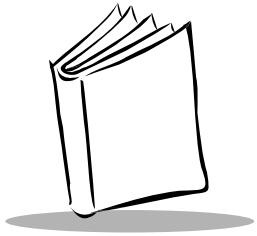



**Figure 2-6. LS 6800 Operation with Synapse Cable**

1. Connect the LS 6800 Synapse Adapter Cable into the port at the end of the scanner's casing.
2. Plug the other end of the Synapse Adapter Cable into the Synapse cable.
3. Follow additional instructions in the *Synapse Interface Guide*. Power is supplied by the Synapse cable interface.



# Chapter 3
# Scanning with the LS 6800

## LS 6800 Scanning Mode Options

The LS 6800 supports several scanning options:

### *Smart Raster*

The LS 6800's "Smart Raster" capability causes the scanner to emit a raster pattern dynamically adjusted to the particular PDF417 bar code's height. To increase scanning efficiency and decrease decode time, the scanner determines the height of the bar code, opening to a size optimal for decoding that bar code.

There are several ways to activate the scanner in normal "Smart Raster" operation: pulling the trigger line to ground, placing a bar code in the blinking laser pattern, or through host software command. Each method causes a slab raster pattern to appear. If the target is a 1-D bar code, the scanner decodes the symbol. If the target bar code is PDF417, the scanning

pattern opens to a full, optimized raster pattern as soon as the scanner is properly aligned over the bar code.

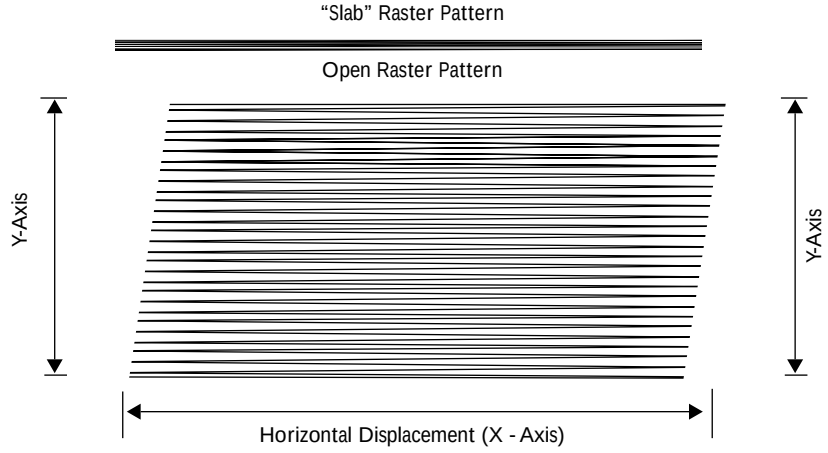



**Figure 3-1. LS 6800 Aiming and Scanning Patterns**

For best operation in smart raster mode, keep the scan pattern as parallel to the symbol's rows as possible, keep the bar code as still as possible, and hold the bar code at an angle which does not give specular reflection (see *Specular Reflection* on page 3-8). Also, the bar code should be in good condition.

Unless otherwise programmed, the LS 6800 operates with Smart Raster performance.

## Slab Only Raster

Scanner activation creates a slab raster pattern which does not open vertically, regardless of bar code type. This may provide optimal performance on small PDF417 and 1D bar codes.

## Always Raster

The LS 6800 directly opens to a full raster pattern or to the user-programmed size whenever the scanner is activated.

## Programmable Raster

The user programs the height of the raster pattern and the rate at which it expands. Scanner activation creates the slab raster pattern which only opens for PDF417 bar codes, useful when decoding low-profile 1D and 2D bar codes where over-scanning is not desired. Note that the

height and expansion rate are directly, but not linearly, proportional to their respective parameter values.

### *Line Only*

Scanner activation creates a single scan line, similar to a 1D scan line, which never opens to a raster pattern. This is useful in applications requiring high-speed scanning where a raster pattern is not necessary.

## Trigger Modes

The LS 6800 trigger modes that follow enable the scanner to be set up to scan with (attended mode) or without (unattended mode) operator intervention.

In attended mode, the operator may present the bar code to a scanner in Blinking Laser Mode, or may physically trigger the scanner using Hardware Trigger Mode.

In unattended mode, the scanner activates when bar codes automatically move into its scan path through Continuous Mode or Software Trigger Command. For unattended mode, the system must be set up so that bar codes and scanning patterns properly align and triggering occurs when the bar code is in the proper position for scanning (see *Scan: Alignment or Orientation* on page 3-5).

### *Hardware Trigger*

To install an external triggering switch, plug the trigger switch into the flying lead of the host D-connector (9-pin or 25-pin) of the interface cable. You may use a standard, off-the-shelf stereo mini-plug, such as those used for portable cassette or CD players. The drawing below gives wiring details.

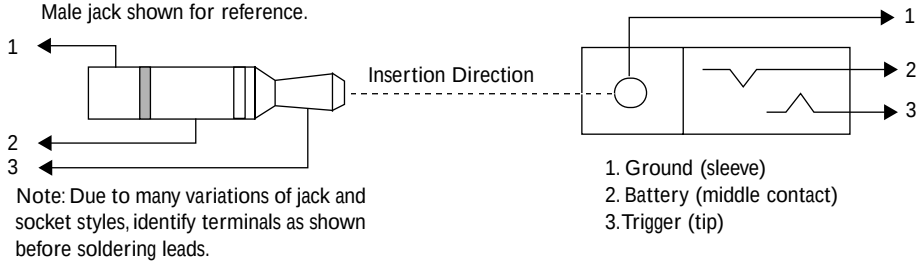



**Figure 3-2. Installing External Triggering Switch**

In the host cable mode, the scanner triggers if the interface line is pulled to ground.

### Software Trigger Command (RS-232C)

In this mode, the LS 6800 scans when commanded by the host. Receipt of an FS character (ASCII decimal value of 28) — equivalent to a trigger pull — turns on the laser and begins a decode attempt. Receipt of a CAN character (ASCII decimal value 24) — equivalent to a trigger release — turns off the laser and terminates a decode attempt.

### Blinking Laser Mode

In Blinking Laser Mode, the laser blinks at approximately a 50% duty cycle to conserve laser life. This is the default operating mode.

This mode may be programmed through bar code menus in Chapter 6.

### Continuous Mode

In Continuous Laser Mode, the scanner continuously attempts to decode a bar code. This mode may be necessary for medium-speed conveyor belts or other time-critical applications.

## Scanning 1-D or 2-D Bar Code

Make sure all connections are secure.

### Present

Make sure the symbol to be scanned is within the scanning range (3 to 8 inches in blinking mode; otherwise refer to *LS 6800 Decode Zones* on page 3-11).

## Scan: Alignment or Orientation

To scan, present the symbol in the direct path of the scan pattern. For proper orientation, position the bar code so that the aiming pattern centers itself on the bar code, as illustrated below.

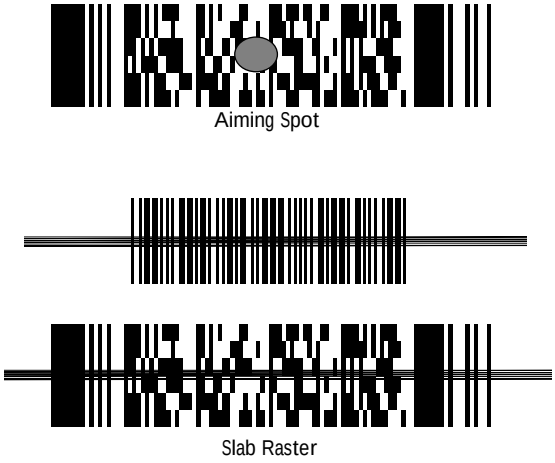



**Figure 3-3. Aiming Patterns on 1-D and 2-D Bar Codes**

*For a 1-D symbol,* the slab raster pattern reads the bar code. A full raster pattern also reads a 1-D bar code so aggressively that scan pattern alignment does not have to be precise.

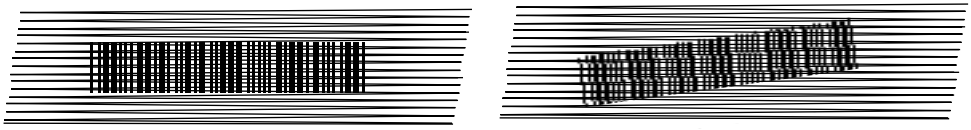


**Figure 3-4. Full Raster Patterns on 1-D Bar Codes**

*For a 2-D symbol,* the pattern spreads vertically to cover the symbol if it is parallel to the symbol's rows. Keep the bar code in the same horizontal plane as the scan pattern.

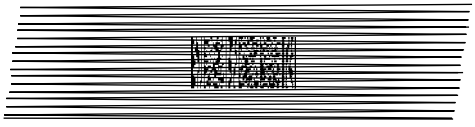


**Figure 3-5. Full Raster Pattern on MicroPDF417 Symbol**

If the pattern does not cover the top and bottom of the 2-D symbol, pull the bar code back until it does. Make sure the scan pattern extends at least *three quarters of an inch* beyond the edges of the bar code. Practice shows what works best.

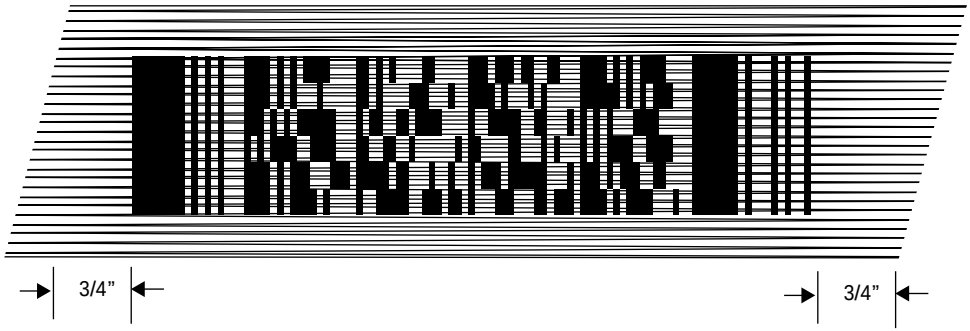



**Figure 3-6. Raster Pattern Expanding Over PDF417 Symbol**

In Always Raster Mode, the raster pattern spreads fully over the bar code, regardless of symbology.

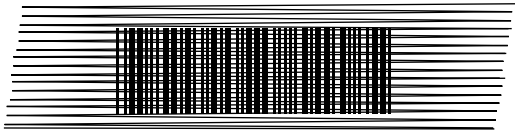


**Figure 3-7. Always Raster Pattern on 1-D Bar Code**

If the symbol is held in a position which makes it unreadable, the red LED blinks to indicate that condition.

## *Successful Decode*

The green LED and a short, high-tone beep indicate a successful decode. The laser turns off.

After a successful decode, remove the symbol from the scan path. If the symbol does not decode, remove the symbol from the scan path and try again. Make sure the bar code is on a clean, white, non-reflective surface.

### Output

Decoded data is transmitted to the host device. Make sure that the RS-232C parameters (e.g., baud rate, parity, etc.) are set properly. See *RS-232C Options* on page 5-71.

## Scan the Entire Bar Code Symbol

- The larger the symbol, the farther away you should position the symbol to permit the raster pattern to cover the symbol. See *LS 6800 Decode Zones* on page 3-11.
- Hold the symbol close for denser symbols.
- In all cases, make sure the scan pattern extends *at least 3/4 inch* beyond each edge of the bar code.
- The PDF417 bar code symbol has multiple rows, but the raster pattern also has multiple scanning rows. For this reason, do three basic things as you scan:
  - Center the aiming pattern on the bar code, as illustrated before.
  - Keep the pattern in the same horizontal plane as the bar code.

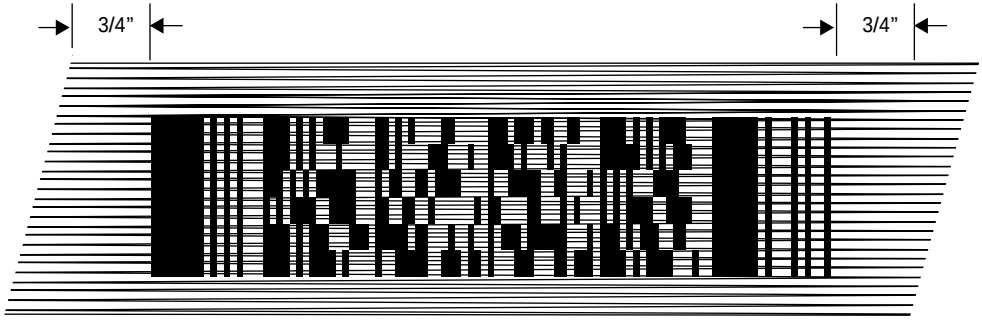



**Figure 3-8. Orienting Scanning Pattern On PDF417 Bar Code**

  - If the vertical scan pattern is not high enough to cover a “tall” PDF417 symbol, move the bar code so the scan pattern moves toward the bottom of the symbol, keeping the beam horizontal to the rows, and then slowly moves back upward toward the top.

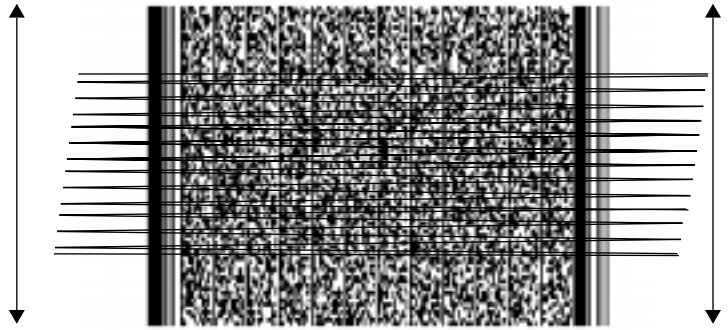


**Figure 3-9. Moving Scan Pattern Upward and Downward on "Tall" PDF Symbol**

- The scan beam does not have to be perfectly parallel with the top and bottom of the symbol (up to a $4^o$ tilt works).

# Specular Reflection

When laser beams reflect directly back into the scanner from the bar code, they can "blind" the scanner and make decoding difficult. This is specular reflection.

To avoid this, scan the bar code at a slight angle so that the beam does not bounce directly back. But don't scan at too oblique an angle; the scanner needs to collect scattered reflections from the scan to make a successful decode. Practice shows what tolerances to work within.

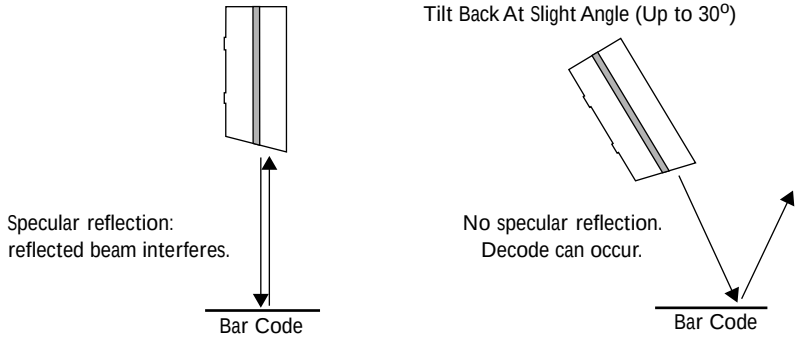



**Figure 3-10. Avoiding Specular Reflection**

For scanning a 1-D bar code, there is only a small specular dead zone to avoid (±2° from the direct laser beam). The specular dead zone is larger for scanning PDF417 (± 9° from the direct laser beam). At the same time, the scanner is not effective if its beams hit the bar code's surface at an angle greater than 30° from the normal to that surface.

# Beeper Definitions

The following table describes beeper indications during standard operation.

**Table 3-1. Beeper Definitions**

| **Standard Use** | |
|---|---|
| **Beeper Sequence** | **Indication** |
| **1 Beep - short high tone (default)** | A bar code symbol was decoded (if decode beeper is enabled). (Beeper tone is programmable.) |
| **1 Beep - long high tone** | Thermal shutdown. |
| **3 Beeps - short high tone** | Power-on or reset. Occurs immediately after the unit is turned on, indicating that the system software is working properly. If three beeps occur during normal operation, it is due to a reset; any work in progress is lost. If this occurs often, contact the Symbol Services Division. |
| **For Linked UCC/EAN-128 Emulation decode beep definitions, see** *Linked UCC/ EAN-128 Emulation* **on page 5-47.** | |
| **Parameter Menu Scanning** | |
| **1 Beep- short high tone** | Correct entry scanned or correct menu sequence performed. |
| **1 Beep- hi/lo/hi/lo tone** | Successful program exit with change in the parameter setting. |
| **2 Beeps - lo/hi tone** | Input error, incorrect bar code, or "Cancel" scanned, wrong entry, incorrect bar code programming sequence; remain in program mode. |
| **Communication** | |
| **4 Beeps - short high tone** | Communication error in the indication field. |
| **4 Beeps - hi/hi/hi/lo** | Receive error. |
| **3 Beeps - lo/hi/lo** | ADF transmit error. |

# Macro PDF Beeper Definitions

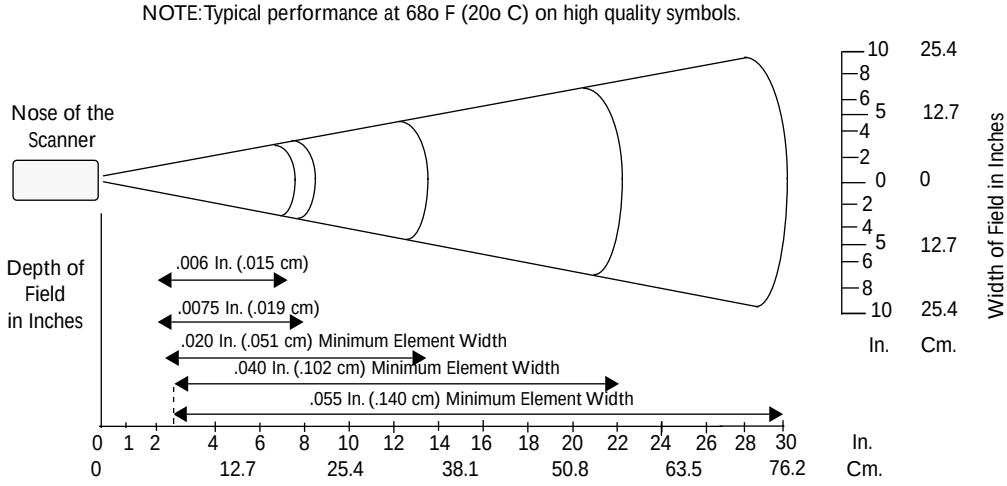
The following table describes beeper indications for Macro PDF.

**Table 3-2. Macro PDF Beeper Indications**

| Error Beep Sequence | Indication |
|---|---|
| **1 Low Long** | Hi-level decode error caused by incorrect symbol.<br>Trying to transmit a Macro PDF field without enabling the *Uses Escape Characters* parameter. |
| **2 Low Long** | File ID error. A bar code not in the current MPDF sequence was scanned. |
| **3 Low Long** | Out of memory. There is not enough buffer space to store the current MPDF symbol. |
| **4 Low Long** | Bad symbology. You scanned a 1-D or 2-D bar code in an MPDF sequence, a duplicate MPDF label, an incorrect sequence, or are trying to transmit an empty or illegal MPDF field. |
| **5 Low Long** | Flushing buffer. |
| **Fast Warble** | Successful parameter scanned. |
| | |
| **Decode Beep Sequence** | **Indication** |
| **1 Short** | Standard decode and transmit beep for all symbols. |
| **2 Short** | MPDF symbol is buffered.<br>Transmission of the buffered data is indicated by the single beep. |

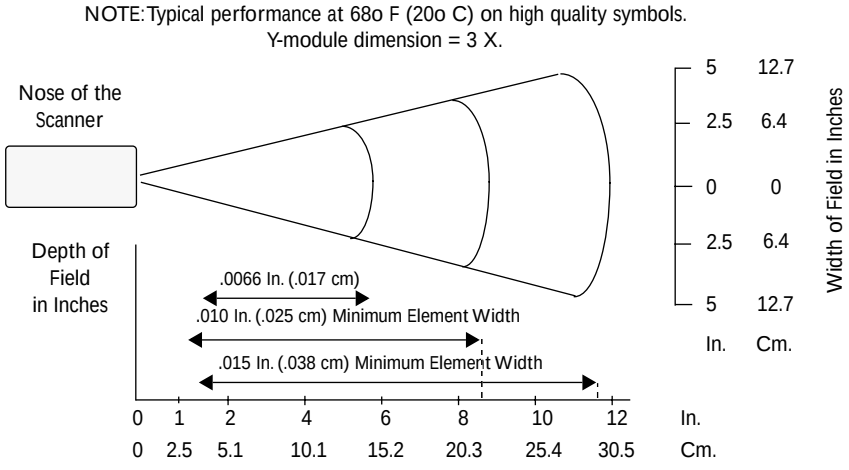
# LS 6800 Decode Zones

## *1-D Bar Codes*



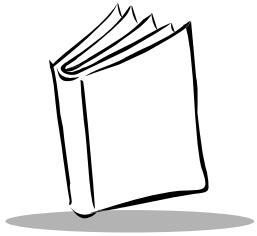

LS 6800 1-D Decode Zone: Depth of field as a function of minimum element width.

## *PDF417 Bar Codes*




LS 6800 2-D Decode Zone: Depth of field as a function of minimum element width.

**Figure 3-11. LS 6800 Decode Zones**



# *Chapter 4*
# *Maintenance and Specifications*

## Maintenance

The LS 6800 scanner and power supply are designed to provide reliable service over an extended period of time with virtually no maintenance.

For the scanner, the only maintenance required is periodic cleaning of the exit window.

- Do not allow any abrasive material to touch the window.
- Remove any dirt particles with a damp cloth.
- Wipe the window using a tissue moistened with ammonia/water.
- Do not spray water or other cleaning liquids directly into the window.

## What If ...

### Nothing happens when you follow the operating instructions?

- Check the system power.
- Make sure you are using the correct interface cable for the host device.
- Check for loose cable connections.
- Make sure the scanner is programmed to decode the symbology you are trying to read. See Chapter 5, *Programming the LS 6800* and Chapter 6, *Advanced Data Formatting*.
- Check the bar code to make sure it is not severely defaced.
- Try scanning a test symbol of the symbology you are trying to read.

### Your terminal operates but scanned data is not displayed correctly?

- Check the system power.
- Check for loose cable connections.
- Check that the communication parameters (baud rate, parity, stop bits, etc.) are set properly for the receiving device.

### Your terminal is operative but scanned data is not displayed? Or is displayed improperly on the screen?

- If you're working with a Synapse cable, refer to your *Synapse Interface Guide.*
- If you're working in RS-232C single-port mode, check that the following parameters have been programmed correctly:

  > Baud Rate  > Stop Bits  > Parity  > Bits per Character

### The laser does not activate, followed by a beep sequence?

- You may be scanning in an inappropriately hot environment. Remove the scanner from the environment, or allow the laser to cool down.

# Accessories

## *Required Accessories*

The following items, required for a complete system, are available through contacting your local Symbol representative or business partner.

| ITEM | Part Number |
|---|---|
| **Host Interface Cables for RS-232 Operation (Available in 14' straight lengths)** | |
| 25-Pin, Male, D Connector (TxD on Pin 3) | 25-17228-XX |
| 25-Pin, Male, D Connector (TxD on Pin 2) | 25-17229-XX |
| 25-Pin, Female, D Connector (TxD on Pin 3) | 25-17230-XX |
| 25-Pin, Female, D Connector (TxD on Pin 2) | 25-17231-XX |
| 9-Pin, Female, D Connector (PC AT: TxD on Pin 3) | 25-17232-XX |
| **Synapse Adapter Cable (10-pin, Male)** | 25-17233-XX |
| **Power Supply (Select One):** | |
| 115 VAC Power Supply | 50-14000-008 |

| | |
|---|---|
| 220/240 VAC Power Supply (Europe) | 50-14000-009 |
| 100 VAC Power Supply (Japan) | 50-14000-010 |

**User Documentation**

| | |
|---|---|
| LS 6800 Product Reference Guide | 70-35933-XX |

## *Optional Accessories*

The following items not included in the standard configuration are available through contacting your local Symbol representative or business partner.

**ITEM**

Wall-Mount Bracket

Desk Stand

# Technical Specifications

The following table provides technical specifications for the LS 6800 scanner.

**Table 4-1. LS 6800 Specifications**

| Item | Description |
|---|---|
| Power Requirements | +5 V, 400 mA Typical |
| Laser Diode Power | 1.0 mW, max. |
| Scan Pattern | |
| Start Time: | 0.065 sec. to 75% of steady state horizontal amplitude;<br>0.50 sec. to 90% of steady state vertical amplitude |
| Pattern Size: | At 9.5 in. from the nose of the scanner, the pattern is 7.2 in. horizontally and 2.4 in. vertically. |
| Scan Rate: | 560 scans/sec. 280 Hz ±15 Hz (horizontal) |
| Frame Rate: | 25 frames/sec. 12.5 Hz ±1 Hz (vertical) |
| Optical Resolution: | Can decode a 6.6 mil (minimum X-dimension) symbol (PDF417); Y-dimension must be 3X for PDF417 and 2X for MicroPDF417. High Density version available (4 mil minimum). |

**Table 4-1. LS 6800 Specifications (Cont'd)**

| Item | Description |
|---|---|
| Angular Orientation Tolerances<br>Pitch Tolerance:<br>Skew:<br>Rotational Tolerance:<br><br>Dead Zone/Optical Throw: | <br>± 30o ("front to back")<br>± 15o from plane parallel to symbol ("side-to-side")<br>± 4o (for scanning benchmark label, assuming 3:1 codeword aspect ratio)<br>± 2o *(1-D symbologies)* or ± 9o*(PDF417)* from beam direction |
| Print Contrast Resolution | 25% *(1-D symbologies)* or 35% *(PDF417)* absolute dark/light reflectance differential, measured at 675 nm. |
| Ambient Light Immunity | Up to 8000 ft-candles of sunlight |
| Humidity | 5 - 95% (non-condensing) |
| Shock | 3-ft drop to concrete |
| Environmental Sealing | NEMA 12 rating; drip-proof, dust-tight |
| Operating Temperature | -20o to 40o C; -4o to 104o F |
| Storage Temperature | -40o to 60o C; -40o to 140o F |
| Scanner Connector | **DB-15 modular** connector<br>**Pins 1, 2**: + 5V power supply<br>**Pin 3**: Transmit (TxD)<br>**Pin 4**: Receive (RxD)<br>**Pin 5**: Request to Send (RTS)<br>**Pin 6**: Clear to Send (CTS)<br>**Pin 7**: Synapse control data<br>**Pin 8**: Synapse control clock<br>**Pin 9**: Data Terminal Ready (DTR)<br>**Pin 10**: Trigger signal output/input<br>**Pin 11**: Ground<br>**Pin 12**: Scan LED (output)<br>**Pin 13**: Beeper (output)<br>**Pin 14**: Decode LED (output)<br>**Pin 15**: Ground |
| Straight Cable Length | 14 ft. 427 cm |
| Weight | 22 oz 624 gm |
| Height | 1.75 in. 4.4 cm |
| Length | 4.5 in. 11.4 cm |

**Table 4-1. LS 6800 Specifications (Cont'd)**

| Item | Description |
|---|---|
| Width | 2.75 in. 7 cm |
| Agency Approvals | CDRH Class II, IEC 825 Class II, FCC Class A, UL, CSA, VDE, CE |
| Decode Capability | 1-D Symbologies: UPC-A, UPC-E, EAN-8, EAN-13, Code 39, Code 39 Full ASCII, Trioptic Code 39, Code 93, Code 128, Interleaved 2 of 5, Discrete 2 of 5, Codabar, UCC/EAN 128, Bookland EAN, UPC/EAN Coupon Code, MSI Plessey. Cannot autodiscriminate between Code 39 and Code 39 Full ASCII.<br>2-D Symbologies: PDF417 (up to 928 codewords at security level 0 - 8), MicroPDF417. |
| Memory | Two models: 64K RAM, 256K Flash; 256K RAM, 256K Flash |

# Cable Pinouts

The following tables provide cable pinout information.

**Table 4-2. Single Port RS-232C, 25-Pin Male, D-Type Connector - p/n 25-17228-01**

| Pin | Signal | Function |
|---|---|---|
| 2 | RxD | Serial data receive input, driven by the serial data transmit output on the device communicating with the scanner. |
| 3 | TxD | Serial data transmit output. This drives the serial data receive input on the device communicating with the scanner. |
| 4 | CTS | Clear-to-send handshaking input line, used optionally by another device to signal the scanner to begin transmitting data. It can be used only in conjunction with the RTS line. |
| 5 | RTS | Request-to-send handshaking output line, used optionally by the scanner to signal another device that data is available to send. It can be used only in conjunction with the CTS line. |
| 6 | DTR | Data Terminal Ready. This signal is hardwired active. |
| 7 | Ground | Power supply input ground pin and reference for both output signals. It must be capable of sinking all return current. |

**Table 4-3. Single-Port RS-232C, 25-Pin Male, D-Type Connector — p/n 25-17229-01**

| Pin | Signal | Function |
|---|---|---|
| 2 | TxD | Serial data transmit output. This drives the serial data receive input on the device communicating with the scanner. |
| 3 | RxD | Serial data receive input, driven by the serial data transmit output on the device communicating with the scanner. |
| 4 | RTS | Request-to-send handshaking output line, used optionally by the scanner to signal another device that data is available to send. It can be used only in conjunction with the CTS line. |
| 5 | CTS | Clear-to-send handshaking input line, used optionally by another device to signal the scanner that it may commence transmitting data. It can be used only in conjunction with the RTS line. |
| 7 | Ground | Power supply input ground pin and reference for both output signals. It must be capable of sinking all return current. |
| 20 | DTR | Data Terminal Ready. This signal is hardwired active. |

**Table 4-4. Single-Port RS-232C, 25-Pin Female, D-Type Connector — p/n 25-17230-01**

| Pin | Signal | Function |
|---|---|---|
| 2 | RxD | Serial data receive input, driven by the serial data transmit output on the device communicating with the scanner. |
| 3 | TxD | Serial data transmit output. This drives the serial data receive input on the device communicating with the scanner. |
| 4 | CTS | Clear-to-send handshaking input line, used optionally by another device to signal the scanner to begin transmitting data. It can be used only in conjunction with the RTS line. |

**Table 4-4. Single-Port RS-232C, 25-Pin Female, D-Type Connector — p/n 25-17230-01 (Cont'd)**

| Pin | Signal | Function |
|---|---|---|
| 5 | RTS | Request-to-send handshaking output line, used optionally by the scanner to signal another device that data is available to send. It can be used only in conjunction with the CTS line. |
| 6 | DTR | Data Terminal Ready. This signal is hardwired active. |
| 7 | Ground | Power supply input ground pin and reference for both output signals. It must be capable of sinking all return current. |

**Table 4-5. Single-Port RS-232C, 25-Pin Female, D-Type Connector — p/n 25-17231-01**

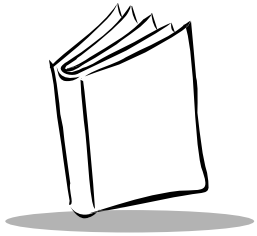
| Pin | Signal | Function |
|---|---|---|
| 2 | TxD | Serial data transmit output. This drives the serial data receive input on the device communicating with the scanner. |
| 3 | RxD | Serial data receive input, driven by the serial data transmit output on the device communicating with the scanner. |
| 4 | RTS | Request-to-send handshaking output line, used optionally by the scanner to signal another device that data is available to send. It can be used only in conjunction with the CTS line |
| 5 | CTS | Clear-to-send handshaking input line, used optionally by another device to signal the scanner to begin transmitting data. It can be used only in conjunction with the RTS line. |
| 7 | Ground | Power supply input ground pin and reference for both output signals. It must be capable of sinking all return current. |
| 20 | DTR | Data Terminal Ready. This signal is hardwired active. |

**Table 4-6. Single-Port RS-232C, 9-Pin Female, D-Type Connector (PC/AT) — p/n 25-17232-01**

| Pin | Signal | Function |
|---|---|---|
| 2 | RxD | Serial data receive input, driven by the serial data transmit output on the device communicating with the scanner. |
| 3 | TxD | Serial data transmit output. This drives the serial data receive input on the device communicating with the scanner. |
| 5 | Ground | Power supply input ground pin and reference for both output signals. It must be capable of sinking all return current. |
| 6 | DTR | Data terminal ready. Hardwired active. |
| 7 | CTS | Clear-to-send handshaking input line, used optionally by another device to signal the scanner to begin transmitting data. It can be used only in conjunction with the RTS line. |
| 8 | RTS | Request-to-send handshaking output line, used optionally by the scanner to signal another device that data is available to send. It can be used only in conjunction with the CTS line |

**Table 4-7. 10-Pin Male Modular Connector for Synapse Cable — p/n 25-17233-01**

| Pin | Signal | Function |
|---|---|---|
| 1 | SYNDATA | Synapse-compatible data line.<br>Bi-directional line. |
| 2 | SYNCLK | Synapse-compatible clock line.<br>Bi-directional line. |
| 3 | VCC | Power. |
| 4 | GND | Power supply ground pin and reference for both output signals. It must be capable of sinking all return current. |



# *Chapter 5*
# *Programming the LS 6800*

Before programming, follow the pertinent setup instructions in Chapter 2, *Setup*.

This chapter provides parameter descriptions and programming bar codes. The *General Parameter Table* beginning on page 5-3 and *Macro PDF Parameter Table* on page 5-7 show selectable parameters and standard default values for the LS 6800.

If the default values suit your requirements, all you need to do is scan the SET DEFAULT bar code. Parameters other than default values can be set by scanning single bar codes or short bar code sequences. Bar codes are arranged up to four per page in a way that you can present the bar code you want to scan directly to the LS 6800, leading with the appropriate edge of this four-sided, folded-over manual.

Note that all parameter settings are stored in non-volatile memory and so are retained after power down.

## Scanning Sequence Examples

In most cases you need only scan one bar code to set a specific parameter. For example, if you want to set the baud rate to 9600, simply scan the 9600 bar code listed under BAUD RATE. The scanner issues a warble tone, signifying a successful parameter entry.

If you want to set specific code lengths or specify Serial Response Time-Out, you have to scan several bar codes. This procedure is described in this chapter.

## Errors While Scanning

Don't worry if you make an error during a scanning sequence. Just reenter the correct parameter.

# Set Defaults

Scanning the SET DEFAULT bar code returns all parameters to the values listed in Table 5-1 and Table 5-2.

**<FN3>91**

**SET DEFAULTS**

# General Parameter Table

The following table provides general parameter selections and defaults.

**Table 5-1. General Parameter Table**

| Parameter | Selection | Default |
|---|---|---|
| Set Defaults | None | Set Default Values |
| Host Selection | RS-232, Null | RS-232 |
| Parameter Program via RS-232 (Simple Scanner Control) | Enabled, Disabled | Enabled |
| Code Types Enabled | UPC-A, UPC-E, EAN-8, EAN-13, Code 128, Code 39, Code 39 Full ASCII, Trioptic Code 39, Code 93, Codabar, Interleaved 2 of 5, Discrete 2 of 5, UCC/EAN 128, Bookland EAN, UPC/EAN Coupon Code, MSI Plessey, PDF417, MicroPDF417 | UPC-A/E, EAN-8/ 13, Code 128, Code 39, UCC/EAN 128, PDF417 |
| Code 39 Lengths | Any Length, Length Within Range, 1 or 2 Discrete Lengths | Length Within Range: 01-55 |
| Code 93 Lengths | Any Length, Length Within Range, 1 or 2 Discrete Lengths | Length Within Range: 04-55 |
| Codabar Lengths | Any Length, Length Within Range, 1 or 2 Discrete Lengths | Length Within Range: 05-55 |
| Interleaved 2 of 5 Lengths | Any Length, Length Within Range (2 to 54 characters), 1 or 2 Discrete Lengths | 1 Discrete Length 14 |
| Discrete 2 of 5 Lengths | Any Length, Length Within Range (2 to 54 characters), 1 or 2 Discrete Lengths | 1 Discrete Length 12 |
| MSI Plessey Lengths | Any Length, Length Within Range, 1 or 2 Discrete Lengths | Any Length |

**Table 5-1. General Parameter Table (Cont'd)**

| Parameter | Selection | Default |
|---|---|---|
| **Decode Options:** | | |
| Transmit UPC-A Check Digit | Enabled, Disabled | Enabled |
| Transmit UPC-E Check Digit | Enabled, Disabled | Enabled |
| Convert UPC-E to UPC-A | Enabled, Disabled | Disabled |
| Code 39 Check Digit Verification | Enabled, Disabled | Disabled |
| Transmit Code 39 Check Digit | Enabled, Disabled | Disabled |
| I 2 of 5 Check Digit Verification | Enabled, Disabled | Disabled |
| Transmit I 2 of 5 Check Digit | Enabled, Disabled | Disabled |
| Convert I 2 of 5 to EAN-13 | Enabled, Disabled | Disabled |
| MSI Plessey Check Digits | One, Two | One |
| Transmit MSI Plessey Check Digit | Enabled, Disabled | Disabled |
| MSI Plessey Check Digit Algorithm | Mod 10/Mod 10, Mod 10/Mod 11 | Mod 10/Mod 10 |
| Convert EAN-8 to EAN-13 | Enabled, Disabled | Disabled |
| Decode UPC-EAN Supplemental | Enabled, Disabled, Auto | Disabled |
| CLSI Editing | Enabled, Disabled | Disabled |
| NOTIS Editing | Enabled, Disabled | Disabled |
| Transmit Code ID Character | None, Symbol Std., AIM Std. | None |
| Decode Buffering | Enabled, Disabled | Disabled |
| Transmit "No Decode" Message | Enabled, Disabled | Disabled |
| Transmit LRC Checksum | Enabled, Disabled | Disabled |
| UPC-A Preamble | None, System Character, System Character & Country Code | None |
| UPC-E Preamble | None, System Character, System Character & Country Code | None |
| Code 128 Emulation (MicroPDF417 only) | Enabled, Disabled | Enabled |
| UCC/EAN-128 Emulation (MicroPDF417 only) | Decode, Ignore, Auto-discriminate Linked Symbol | Ignore |
| Pause Duration | 0.0 to 9.9 seconds | 0.0 seconds |
| Prefix/Suffix Values | 4-Digit ASCII Value | Enter |
| Scan Data Transmission Format | <Data>; <Data><Suffix>; <Prefix><Data><Suffix>; <Prefix><Data> | <Data> |

**Table 5-1. General Parameter Table (Cont'd)**

| Parameter | Selection | Default |
|---|---|---|
| **Scanning Options:** | | |
| Beep After Good Decode | Enabled, Disabled | Enabled |
| Beeper Tone | High, Medium. Low | High |
| Decode Attempt Duration | 0.0 to 9.9 seconds; Infinite | 5 seconds |
| Trigger Modes | Level, Pulse, Continuous, Blinking | Blinking |
| Scanning Modes | Smart Raster, Always Raster, Programmable Raster, Slab Raster, Line Pattern | Smart Raster |
| Raster Height | 1-15 | 15 |
| Raster Expansion Speed | 1-15 | 11 |
| Aiming Mode | Aiming Dot (Normal or Extended Timeout), Slab Raster | Slab Raster |
| Time Delay to Low Power Mode | 30-second, 1-, 2- , or 3- minutes | 30 seconds |
| Timeout Between Decodes: Same Symbol | 0.0 to 9.9 seconds | 0.6 seconds |
| Timeout Between Decodes: Different Symbol | 0.0 to 9.9 seconds | 0.0 seconds |
| FAT Mode | Enabled, Disabled | Disabled |
| **Security Options:** | | |
| Decode UPC/EAN Supplemental Redundancy | 2 - 20 | 7 |
| Linear Code Type Security Levels | 1 - 4 | 1 |
| Bi-directional Redundancy | Enabled, Disabled | Disabled |
| UPC/EAN Security Levels | 0 - 3 | 1 |
| **Host Port RS-232 Options:** | | |
| Baud Rate | 300, 600, 1200, 2400, 4800, 9600, 19.2K, 38.4 K | 9600 |
| Parity | Even, Odd, Mark, Space, None | None |
| Check Parity of Received Data | Enabled, Disabled | Enabled |
| Stop Bit Select | One, Two | One |
| Hardware Handshaking | None, RTS/CTS Options 1, 2, 3 | None |
| Software Handshaking | None, ENQ, ACK/NAK, ACK/NAK with ENQ, XON/XOFF | None |
| Intercharacter Delay | 00 - 99 ms. | 0 |

**Table 5-1. General Parameter Table (Cont'd)**

| Parameter | Selection | Default |
|---|---|---|
| Host Serial RTS Line State | Low RTS, High RTS | Low RTS |
| Serial Response Time-out | 00 - 9.9 seconds | 2.0 seconds |
| Beep on <BEL> | Enabled, Disabled | Disabled |
| 8-Bit ASCII Format | 7-Bit ASCII, 8-Bit ASCII | 8-Bit ASCII |

# Macro PDF Parameter Table

The following table provides Macro PDF parameter selections and defaults.

**Table 5-2. Macro PDF Parameter Table**

| Parameter | Selection | Default |
|---|---|---|
| Macro PDF Transmit/Decode Mode | Buffer All Symbols / Transmit Macro PDF When Complete<br>Transmit Any Symbol in Set / No Particular Order<br>Scan in Sequence Only / Transmit in Sequence Without Buffering<br>Buffer Scans Out of Order / Transmit Scans in Order | Buffer All Symbols / Transmit Macro PDF When Complete |
| Transmit Each Symbol in Codeword Format | Enabled or Disabled | Disabled |
| Escape Characters | None, GLI Protocol, ECI Protocol | None |
| Character Set ECIs | Delete or Transmit | Delete |
| ECI Decoder | Enabled or Disabled | Enabled |
| Transmit Unknown Codewords | Enabled or Disabled | Disabled |
| Transmit Macro PDF User-Selected Field | | |
| Transmit File Name | Enabled or Disabled | Disabled |
| Transmit Block Count | Enabled or Disabled | Disabled |
| Transmit Time Stamp | Enabled or Disabled | Disabled |
| Transmit Sender | Enabled or Disabled | Disabled |
| Transmit Addresses | Enabled or Disabled | Disabled |
| Transmit File Size | Enabled or Disabled | Disabled |
| Transmit Checksum | Enabled or Disabled | Disabled |
| Transmit Macro PDF Control Header | Enabled or Disabled | Disabled |
| Last Block Marker | Enabled or Disabled | Disabled |

# Host Selection

This determines whether the LS 6800 interfaces with an RS-232C host, a Synapse host, or no host.

**<FN3>2050A37**

**RS-232 HOST**

SYNAPSE HOST

See the *Synapse Interface Guide* for this bar code.

**NULL HOST**

**<FN3>2050AFE**

# Parameter Programming via RS-232C (Simple Scanner Control — Enable and Disable)

Parameters are stored in non-volatile memory, and they control different options for scanner operation. These are usually programmed by scanning special Code 128 bar codes, which begin with **<FN3>**, from this manual. However, by enabling parameter programming via RS-232, you can program the scanner through host command to the scanner's RS-232 port, a capability contained in a feature set called Simple Scanner Control.

**Command Syntax**. When using Simple Scanner Control, program a scanner parameter by sending a string to the scanner in this format:

**13XXXX<parameter bar code>**

where **<parameter bar code>** refers to the string of characters (found under each programming bar code in this manual, minus the **<FN3>** character) encoded in the bar code for the parameter option you wish to set, and **XXXX** is a four-digit hexadecimal number representing the number of characters in the command. For example, to enable Pulse Trigger, send: **13000D2050202**. To disable Pulse Trigger, send **13000D2050200**.

Depending on the application environment, it may be possible to enable LRC checksum and ACK/NAK protocols for this communication. See *LRC Checksum* on page 5-43 and *Software Handshaking* on page 5-86.

**Optional T Character**. An optional 'T' character can be added to the string **<parameter bar code>** to indicate a temporary change to the value of a parameter which lasts only during one power cycle. After that cycle, the value returns to that saved in the unit's EEPROM. The 'T' character option does not apply for numerical keypad entries, Set Defaults, or for other parameters (e.g., Flush Macro PDF Buffer) which perform a single, isolated action rather than change a parameter setting in EEPROM. The 'T' character counts in the total to be entered in the **XXXX** field.

Two parameters which require scanning several bar codes can be changed by sending a single RS-232 command. These parameters are Raster Height and Raster Expansion Rate. Their respective single RS-232 commands are: **13000D20N030X** and **13000D20N020X**, where X is a hexadecimal value between 1 and F inclusive, representing the parameter value. Sending two consecutive ESC characters (ASCII decimal value 27) resets the command parser.

At least 1.25 seconds must elapse after a parameter change in EEPROM before the scanner can accept another parameter change command. With temporary parameter changes, at least 100 ms must elapse between consecutive commands.

Select whether the LS 6800 can be programmed through commands from the host to the scanner's RS-232 port.

**<FN3>1040621**

**ENABLE SIMPLE SCANNER CONTROL**

**DISABLE SIMPLE SCANNER CONTROL**

**<FN3>1040620**

# Code Types

The bar code menu selections enable the scanner to decode any or all of the following symbologies:

- UPC Versions A and E (EAN 8 and 13)
- Code 39
- Code 93
- Interleaved 2 of 5
- UCC/EAN 128
- Bookland EAN
- MSI Plessey
- Codabar
- Trioptic Code 39
- Code 128
- Discrete 2 of 5
- PDF417
- MicroPDF417
- UPC/EAN Coupon Code

The integrated scanner autodiscriminates between all of the above symbologies, except for Code 39 and Code 39 Full ASCII.

## *Code 39 / Code 39 Full ASCII*

The ASCII character set assigns a code to letters, punctuation marks, numerals, and most control keystrokes on the keyboard.

The first 32 codes are non-printable and are assigned to keyboard control characters, such as **BACKSPACE** and **RETURN**. The other 96 are called printable codes because all but **SPACE** and **DELETE** produce visible characters.

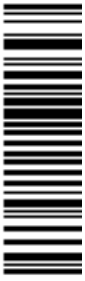
Code 39 Full ASCII interprets the bar code control character ($ + % /) preceding a Code 39 symbol and assigns an ASCII character value. For example, when Code 39 Full ASCII is enabled and a **+B** is scanned, it is interpreted as **b**, %J as ?, and $H emulates the keystroke **BACKSPACE**. Scanning **ABC$M** outputs the keystroke equivalent of **ABC ENTER**.

The LS 6800's integrated decoder autodiscriminates between Code 39 and Code 39 Full ASCII.

# Code Type (Cont'd)

**<FN3>1000011**

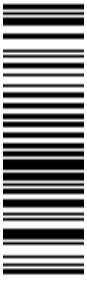
**ENABLE UPC-A**



**<FN3>1000020**

**DISABLE UPC-E**

**DISABLE UPC-A**

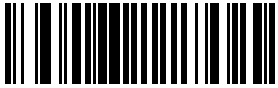


**<FN3>1000010**

**ENABLE UPC-E**

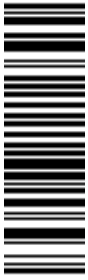
**<FN3>1000021**

# Code Type (Cont'd)



**<FN3>1000041**

**ENABLE EAN-8**

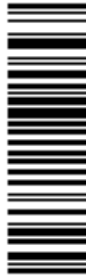


**<FN3>1000030**

**DISABLE EAN-13**

**DISABLE EAN-8**

**<FN3>1000040**



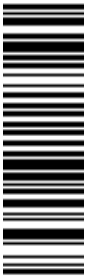
**ENABLE EAN-13**

**<FN3>1000031**

# Code Type (Cont'd)

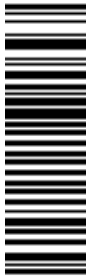
**<FN3>1000001**

**ENABLE CODE 39**



**<FN3>1020110**

**DISABLE CODE 39 Full ASCII**



**<FN3>1000000**

**DISABLE CODE 39**

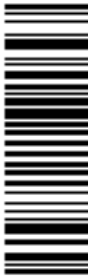
**ENABLE CODE 39 Full ASCII**

**<FN3>1020111**

# Code Type (Cont'd)

**<FN3>1000221**

**ENABLE TRIOPTIC CODE 39**

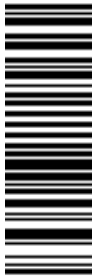


**<FN3>1000110**

**DISABLE CODE 93**

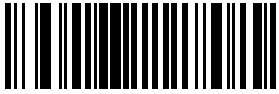
**DISABLE TRIOPTIC CODE 39**

**<FN3>100220**



**ENABLE CODE 93**

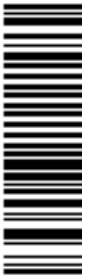
**<FN3>1000111**



# Code Type (Cont'd)

<FN3>1000071

**ENABLE CODABAR**

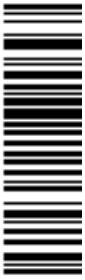


F<FN3>1000100

**DISABLE CODE 128**

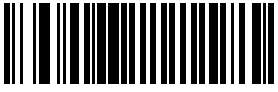
**DISABLE CODABAR**

<FN3>1000070



**ENABLE CODE 128**

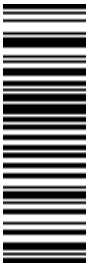
<FN3>10000101



# Code Type (Cont'd)

**<FN3>1000061**

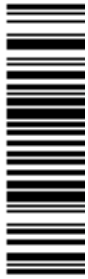
**ENABLE I 2 of 5**



**<FN3>1000050**

**DISABLE D 2 OF 5**

**DISABLE I 2 of 5**

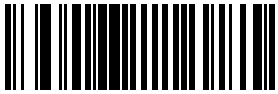


**<FN3>1000060**

**ENABLE D 2 OF 5**

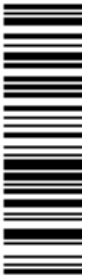
**<FN3>1000051**

# Code Type (Cont'd)



**<FN3>1000231**

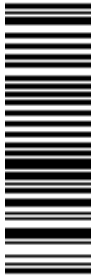
**ENABLE BOOKLAND EAN**



**<FN3>1040330**

**DISABLE UCC/EAN-128**

**DISABLE BOOKLAND EAN**

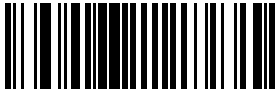


**<FN3>1000230**

**ENABLE UCC/EAN-128**

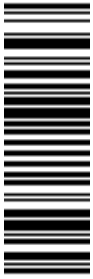
**<FN3>1040331**

# Code Type (Cont'd)



**<FN3>1000141**

**ENABLE MSI PLESSEY**

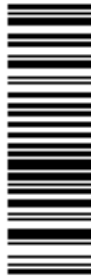


**<FN3>1000240**

**DISABLE UPC/EAN COUPON CODE**

**DISABLE MSI PLESSEY**

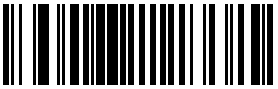
**<FN3>1000140**



**ENABLE UPC/EAN COUPON CODE**

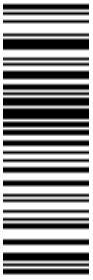
**<FN3>1000241**

# Code Type (Cont'd)



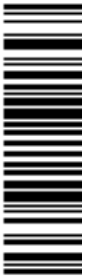
**<FN3>1000161**

**ENABLE PDF417**



**<FN3>1000340**

**DISABLE MICROPDF417**



**<FN3>1000160**

**DISABLE PDF417**

**ENABLE MICROPDF417**

**<FN3>1000341**

# Code Lengths

Code lengths for certain one-dimensional code types (e.g., Code 39, Codabar) may be set for any length, one or two discrete lengths, or lengths within a specific range. The length of a code refers to the number of characters (i.e., human readable characters) the code contains.

- **Length Within Range** - This option allows you to decode a one-dimensional code type within a specified range. For example, to decode Code 39 characters containing between 4 and 12 characters, first scan **Code 39 Length Within Range**. Then scan **0**, **4**, **1** and **2** (single-digit numbers must always be preceded by a leading zero).
- **One Discrete Length** - This option allows you to decode only those one-dimensional codes containing a selected length. For example, if you select **Codabar - One Discrete Length**, then scan **1** and **4**, the only Codabar codes decoded are those containing 14 characters. No discrete lengths can be set for Code 128.
- **Two Discrete Lengths** - This option allows you to decode only those one-dimensional codes containing two selected lengths. For example, if you select **I 2 of 5 Two Discrete Lengths**, then scan **0**, **2**, **1**, **4**, the only Interleaved 2 of 5 codes decoded are those containing 2 or 14 characters. No discrete lengths can be set for Code 128.
- **Any Length** - Scanning this option allows you to decode the selected one-dimensional code type containing any number of characters. For example, if you scan **Codabar Any Length**, you can decode a Codabar symbol containing any number of characters.

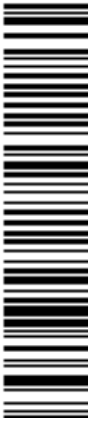
# Code Lengths (Cont'd)

To select lengths for each code type:

1. Scan the desired option.
2. Scan two bar codes from *Numeric Bar Codes* beginning on page 5-96 for each desired length. For example, for a length of "12", scan "1" then "2". For a length of "3", scan "0", then "3". **You must always scan two bar codes for each length**.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98**.**

**<FN3>F0010001013700000**

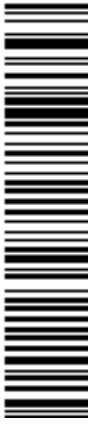
**CODE 39 ANY LENGTH**



**<FN3>F2010001013700000**

**CODE 39 2 DISCRETE LENGTHS**

**CODE 39 1 DISCRETE LENGTH**



**<FN3>F1010001013700000**

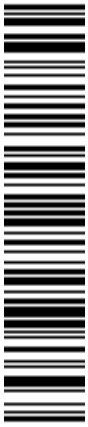
**CODE 39 LENGTH WITHIN RANGE**

**<FN3>F3010001013700000**

# Code Lengths (Cont'd)

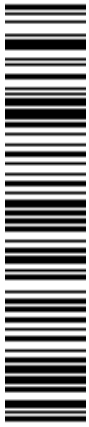
<FN3>F0010203003700007

**CODABAR ANY LENGTH**



<FN3>F2010203013700007

**CODABAR 2 DISCRETE LENGTHS**



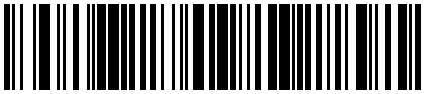
<FN3>F1010203013700007

**CODABAR 1 DISCRETE LENGTH**

**CODABAR LENGTH WITHIN RANGE**

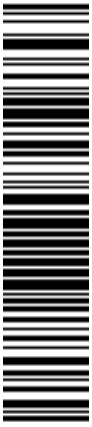
<FN3>F3010203013700007

# Code Lengths (Cont'd)



**<FN3>F0010809003600006**

**I 2 OF 5 - ANY LENGTH**

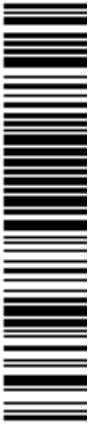


**<FN3>F2010809023600006**

**I 2 OF 5 2 DISCRETE LENGTHS**

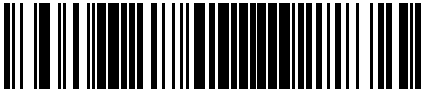
**I 2 OF 5 1 DISCRETE LENGTH**

**<FN3>F1010809023600006**



**I 2 OF 5 - LENGTH WITHIN RANGE**

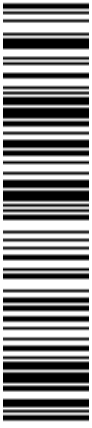
**<FN3>F3010809023600006**



# Code Lengths (Cont'd)

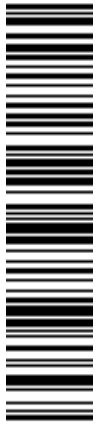
**<FN3>F0010607023700005**

**D 2 OF 5 - ANY LENGTH**



**<FN3>F2010607003700005**

**D 2 OF 5 2 DISCRETE LENGTHS**



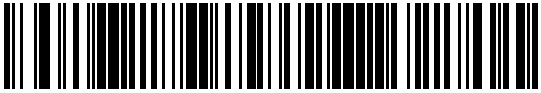
**<FN3>F1010607023700005**

**D 2 OF 5 1 DISCRETE LENGTH**

**D 2 OF 5 - LENGTH WITHIN RANGE**

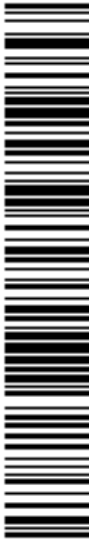
**<FN3>F3010607023700005**

# Code Lengths (Cont'd)



**<FN3>F0010A0B023700011**

**CODE 93 - ANY LENGTH**



**<FN3>F2010A0B023700011**

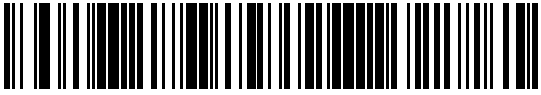
**CODE 93 - 2 DISCRETE LENGTHS**

**CODE 93 - 1 DISCRETE LENGTH**

**<FN3>F1010A0B023700011**

**CODE 93 - LENGTH WITHIN RANGE**

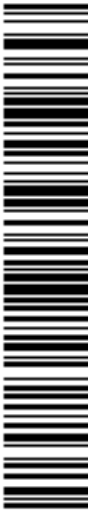
**<FN3>F3010A0B023700011**



# Code Lengths (Cont'd)

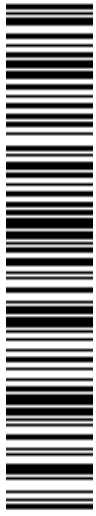
**<FN3>F0010F10013700014**

**MSI PLESSEY - ANY LENGTH**



**<FN3>F2010F10013700014**

**MSI PLESSEY - 2 DISCRETE LENGTHS**



**<FN3>F1010F10013700014**

**MSI PLESSEY - 1 DISCRETE LENGTH**

**MSI PLESSEY - LENGTH WITHIN RANGE**

**<FN3>F3010F10013700014**

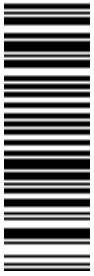
# Decode Options

## *Transmit UPC-E/UPC-A Check Digit*

Select if decoded UPC symbols are transmitted with or without a check digit.

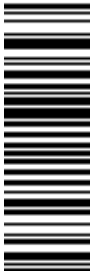
**<FN3>1020021**

**TRANSMIT UPC-A CHECK DIGIT**



**<FN3>1020030**

**DO NOT TRANSMIT UPC-E CHECK DIGIT**



**<FN3>1020020**

**DO NOT TRANSMIT UPC-A CHECK DIGIT**

**TRANSMIT UPC-E CHECK DIGIT**

**<FN3>1020031**

## *Convert UPC-E to UPC-A*

Use this parameter to convert UPC-E (zero suppressed) decoded data to UPC-A format before transmission. After conversion, data follows UPC format and is affected by UPC-A programming selections (e.g., Preamble, Check Digit).

**<FN3>1020001**

**CONVERT UPC-E TO UPC-A**

**DO NOT CONVERT UPC-E TO UPC-A**

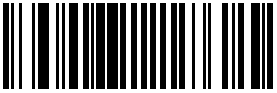
**<FN3>1020000**

## Code 39 Check Digit Verification

When enabled, this parameter checks the integrity of a Code 39 symbol to ensure that it complies with specified algorithms.

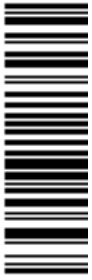
## Transmit Code 39 Check Digit

When enabled, this parameter checks the integrity of a Code 39 symbol to ensure that it complies with specified algorithms.



**<FN3>1020241**

**TRANSMIT CODE 39 CHECK DIGIT**

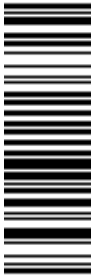


**<FN3>1020040**

**DISABLE CODE 39 CHECK DIGIT**

**DO NOT TRANSMIT CODE 39 CHECK DIGIT**

**<FN3>1020240**

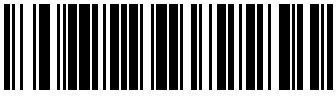


**ENABLE CODE 39 CHECK DIGIT**

**<FN3>1020041**

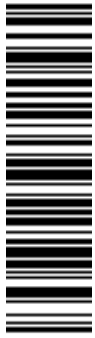
## *I 2 of 5 Check Digit Verification*

When enabled, this parameter checks the integrity of an I 2 of 5 symbol to ensure it complies a specified algorithm, either Uniform Symbology Specification (USS), or Optical Product Code Council (OPCC).



**<FN3>2051E00**

**DISABLE**

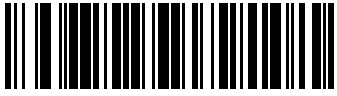


**<FN3>2051E02**

**OPCC CHECK DIGIT**

**USS CHECK DIGIT**

**<FN3>2051E01**



## Transmit I 2 of 5 Check Digit

Scan a bar code below to transmit data with or without the check digit.

**<FN3>1020211**

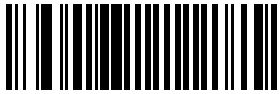
**TRANSMIT I 2 of 5 CHECK DIGIT (ENABLE)**

**DO NOT TRANSMIT I 2 of 5 CHECK DIGIT (DISABLE)**

**<FN3>1020210**

## *Convert I 2 of 5 to EAN-13*

This parameter converts a 14 character I 2 of 5 code into EAN-13, and transmits to the host as EAN-13. In order to accomplish this, the I 2 of 5 code must be enabled, one length must be set to 14, and the code must have a leading zero and a valid EAN-13 check digit.



**<FN3>1020201**

**CONVERT I 2 of 5 to EAN-13 (ENABLE)**

**DO NOT CONVERT I 2 of 5 to EAN-13 (DISABLE)**

**<FN3>1020200**

## MSI Plessey Check Digits

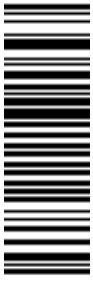
These check digits at the end of the bar code verify the integrity of the data. At least one check digit is always required. Check digits are not automatically transmitted with the data (see below).

## Transmit MSI Plessey Check Digit

Scan the appropriate bar code below to transmit the data with or without the check digit.

**<FN3>1020120**

**ONE MSI PLESSEY CHECK DIGIT**

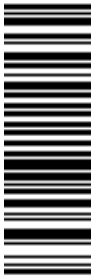


**<FN3>1020130**

**TRANSMIT MSI PLESSEY CHECK DIGIT (DISABLE)**

**TWO MSI PLESSEY CHECK DIGITS**

**<FN3>1020121**



**TRANSMIT MSI PLESSEY CHECK DIGIT (ENABLE)**

**<FN3>1020131**

## *MSI Plessey Check Digit Algorithm*

When the two MSI Plessey check digits option is selected, an additional verification is required to ensure integrity. Either of the two following algorithms may be selected.

**<FN3>1020230**

**MOD 10/MOD11**

**MOD 10/MOD10**

**<FN3>1020231**

## Convert EAN-8 to EAN-13

This parameter adds five leading zeros to decoded EAN-8 symbols to make them compatible in format to EAN-13 symbols.

**<FN3>1020201**

**ENABLE CONVERT EAN-8 TO EAN-13**

**DISABLE CONVERT EAN-8 TO EAN-13**

**<FN3>1020200**

## *Decode UPC/EAN Supplemental*

Select whether UPC/EAN is decoded with or without supplemental characters. Supplementals are additionally appended characters (2 or 5) according to specific code format conventions (e.g., UPC A+2, UPC E+2, EAN 8+2).

- If UPC/EAN with supplemental characters is selected, UPC/EAN symbols without supplemental characters won't be decoded.
- If UPC/EAN without supplemental characters is selected and the scanner is presented with a UPC/EAN plus supplemental symbol, the UPC/EAN is decoded and the supplemental characters ignored.
- If autodiscrimination is chosen, the LS 6800 scanner transmits either, after additional processing to ensure a good decode.

---

**Note:** *To minimize the risk of invalid data transmission, it is recommended that you select whether to read or ignore supplemental characters.*

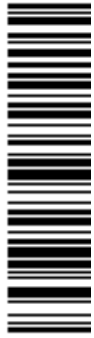
---

**<FN3>2010E01**

**DECODE UPC/EAN SUPPLEMENTALS**

**IGNORE UPC/EAN SUPPLEMENTALS**

**<FN3>2016E00**



**AUTODISCRIMINATE UPC/EAN SUPPLEMENTALS**

**<FN3>201E02**

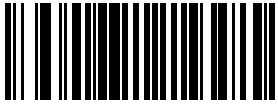
## CLSI Editing

When enabled, this parameter strips the start and stop characters and inserts a space after the first, fifth, and tenth characters of a 14-character Codabar symbol.

**Note:** *Symbol length does not include start and stop characters*

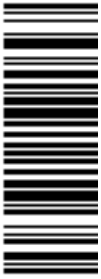
## NOTIS Editing

When enabled, this parameter strips the start and stop characters from decoded Codabar symbol.



**<FN3>1020051**

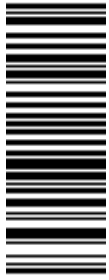
**ENABLE CLSI EDITING**



**<FN3>1020060**

**DISABLE NOTIS EDITING**

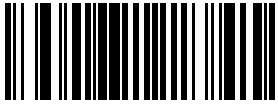
**DISABLE CLSI EDITING**



**<FN3>1020050**

**ENABLE NOTIS EDITING**

**<FN3>1020061**



## Transmit Code ID Character

A code ID character identifies the code type of a scanned bar code, useful when the scanner is decoding more than one code type. If a prefix is selected, the code ID character is sent after the prefix and before the decoded symbol.

You may select no code ID character, a Symbol code ID character, or an AIM ID character. Symbol code ID characters are listed below. AIM ID characters are listed in Appendix A, *Programming Reference*.

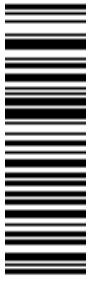
**Table 5-3. Symbol Code Identifiers**

| Code Type | Symbol Identifier |
|---|---|
| UPC-A, UPC-E, EAN-13, EAN-8 | A |
| Code 39 | B |
| Codabar | C |
| Code 128 | D |
| Code 93 | E |
| Interleaved 2 of 5 | F |
| Discrete 2 of 5, D 2 of 5 IATA | G |
| MSI Plessey | J |
| UCC/EAN 128 | K |
| Bookland EAN | L |
| Trioptic Code 39 | M |
| PDF417, MicroPDF417 | X |

## *Transmit Code ID Character (Cont'd)*

**<FN3>2051702**

**TRANSMIT SYMBOL CODE ID CHARACTER**



**<FN3>2051701**

**TRANSMIT AIM CODE ID CHARACTER**

**DO NOT TRANSMIT CODE ID CHARACTER**

**<FN3>2051700**

## Decode Buffering

When this option is enabled the scanner stores decode data until the host is ready to receive it. If the scanner reaches its storage capacity before the host is ready, subsequent trigger pulls have no effect until a buffer is available.

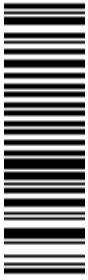
## Transmit “No Decode” Message

When this feature is enabled, “FR” is transmitted if a PDF symbol was detected but not decoded, and “NR” if no PDF symbol is detected and a 1D bar code was not decoded. Prefixes and suffixes enabled are appended around this message.

## *Decode Buffering / "No Decode" Message (Cont'd)*

**<FN3>2051401**

**ENABLE DECODE BUFFERING**

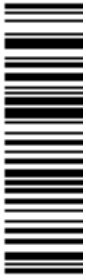


**<FN3>1020100**

**DO NOT TRANSMIT "NO DECODE" MESSAGE**

**<FN3>2051400**

**DISABLE DECODE BUFFERING**



**TRANSMIT "NO DECODE" MESSAGE**

**<FN3>1020101**

## *LRC Checksum*

This option allows appending an LRC checksum character to the end of a decode transmission.

When an LRC checksum is enabled, the format of output data is as follows:

**<STX> <DATA...> <ETX> <LRC>**.

If **<DATA...>** contains the special characters STX, ETX, and DLE, a DLE character is added before each of the special characters to inform the receiving side not to interpret the special characters as control characters. The LRC character is the exclusive OR of all characters except for the LRC character itself.

**<FN3>1040401**

**ENABLE LRC CHECKSUM**

**DISABLE LRC CHECKSUM**

**<FN3>1040400**

## UPC-A and -E Preamble

Three options are given for the lead-in characters of decoded UPC-A or UPC-E symbols transmitted to the host. Select one preamble for UPC-A decodes and one for UPC-E decodes. These lead-in characters are considered part of the symbol itself. The three options are:

- a system character only
- the country code and system character
- no preamble

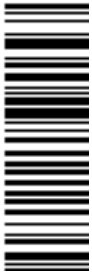
The system character is the digit printed to the extreme left of a UPC symbol. The country code for UPC is always zero, and cannot be transmitted without the system character.

## UPC-A Preamble

Select one option for UPC-A preamble by scanning the appropriate bar code.

**<FN3>2030100**

**NONE**



**SYSTEM CHARACTER**

**<FN3>2030101**

**SYSTEM CHARACTER**
**&**
**COUNTRY CODE**

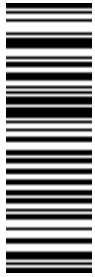
**<FN3>2030102**

## *UPC-E Preamble*

Select one option for UPC-E preamble by scanning the appropriate bar code.

**<FN3>203000**

**NONE**



**<FN3>2030001**

**SYSTEM CHARACTER**

**SYSTEM CHARACTER**
**&**
**COUNTRY CODE**

**<FN3>2030002**

## Code 128 Emulation

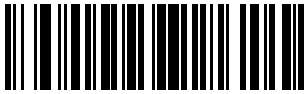
When this parameter is enabled, the scanner transmits data from certain MicroPDF417 symbols as if it was encoded in Code 128 symbols. Transmit AIM Symbology Identifiers must be enabled for this parameter to work.

If Code 128 Emulation is enabled, these MicroPDF417 symbols are transmitted with one of the following prefixes:

**]C1** if the first codeword is 903-907, 912, 914, or 915
**]C2** if the first codeword is 908 or 909
**]C0** if the first codeword is 910 or 911

If disabled, they are transmitted with one of the following prefixes:

**]L3** if the first codeword is 903-907, 912, 914, or 915
**]L4** if the first codeword is 908 or 909
**]L5** if the first codeword is 910 or 911

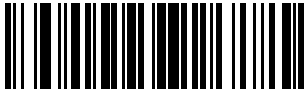


**<FN3>10M0171**

**ENABLE CODE 128 EMULATION**

**ENABLE CODE 128 EMULATION**

**<FN3>10M0170**



## Linked UCC/EAN-128 Emulation

Certain MicroPDF417 symbols can be "linked" with a 1-D symbol during transmission as if they were one symbol. The MicroPDF417 symbol provides supplemental data to the 1-D symbol. Three scanning options are offered for these symbols:

- If Decode Linked Symbol is selected, Code 128, ITF-14 and UPC/EAN symbols are not decoded unless a MicroPDF417 symbol beginning with codeword 906, 907, 912, 914, or 915 is present.
- If Ignore Linked Symbol is selected, MicroPDF417 symbols beginning with codeword 906, 907, 912, 914, or 915 are ignored.
- If Autodiscriminate Linked Symbol is selected, the scanner looks for a MicroPDF417 symbol when scanning a 1-D symbol. If a MicroPDF417 symbol is not detected within the timeout period, the 1-D symbol data is transmitted.

---

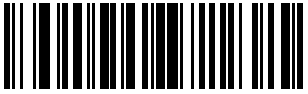
**Note:** *The Timeout Between Decodes, Different Symbols parameter must be set to 0 before Ignore Linked Symbol or Autodiscriminate Linked Symbol can be selected.*

*When scanning a linked symbol, if the 1-D symbol is scanned first the scanner issues a low beep indicating that the MicroPDF417 symbol must still be scanned. Move the laser up until this symbol is scanned. The scanner issues two high beeps: the first indicates the MicroPDF417 symbol was decoded, the second indicates the entire linked symbol was decoded.*

*If the MicroPDF417 symbol is scanned first, the laser opens to a raster pattern and automatically scans the 1-D symbol. This decode is indicated by a high-low-high beep; the first high issues when the MicroPDF417 symbol is decoded, the low issues when the 1-D symbol is decoded, and the second high indicates the entire linked symbol has been decoded.*

---

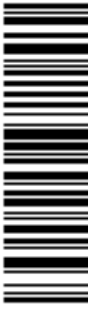
## *Linked UCC/EAN-128 Emulation (Cont'd)*



**<FN3>20N0501**

**DECODE LINKED SYMBOL**

**AUTODISCRIMINATE LINKED SYMBOL**



**<FN3>20N0502**

**IGNORE LINKED SYMBOL**

**<FN3>20N0500**

## *Pause Duration*

This parameter, used in data editing, allows a pause to be inserted at any point in the data transmission. Pauses are set by scanning a two-digit number
(i.e., two bar codes), and are measured in 1/10 second intervals. For example, scanning bar codes “0” and “1” inserts a 1/10 (0.1) second pause; “0” and “5” inserts a 1/2 (0.5) second delay.

To set a pause duration:

1. Scan the **PAUSE DURATION** bar code below.
2. Scan two bar codes from *Numeric Bar Codes* beginning on page 5-96 which represent the desired pause. You must always scan two bar codes.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98.

**<FN3>30C0D20063**

**PAUSE DURATION**

## Prefix/Suffix Values

A prefix/suffix may be appended to scan data for use in data editing. These values are set by scanning a four digit number (i.e., four bar codes) that correspond to keycodes for various terminals. See *Prefix / Suffix Values* on page A-5.

To set a **PREFIX/SUFFIX** value:

1. Scan the option bar code you wish to set.
2. Scan four bar codes from *Numeric Bar Codes* beginning on page 5-96 which correspond to the ASCII value or keystroke you wish to assign. The **ENTER** key is the default for all options.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98**.**

**<FN3>50C0006**

**SCAN SUFFIX**
**(VALUE 1)**

**SCAN PREFIX**
**(VALUE 2)**

**<FN3>50C0107**

## Scan Data Transmission Format

The scan data format options listed below can be selected by the user.

| | |
|---|---|
| Standard: | <data> |
| Option 1: | <data> <SUFFIX> |
| Option 2: | <PREFIX> <data> <SUFFIX> |
| Option 3: | <PREFIX> <data> |

<data> = scanned bar code data
<PREFIX> and <SUFFIX> as selected by the user.

To select a scan data transmission format:

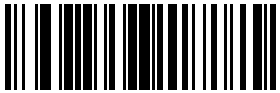
1. Scan the **SCAN OPTIONS** bar code.
2. Scan the bar code corresponding to the desired converted data format.
3. Scan **ENTER.**
4. If you make a mistake, scan **CANCEL** on page 5-98.

---

**Note:** *RS-232C hosts treat the extended keypad default suffix (7013) as ASCII data.*

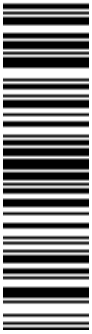
---

## *Scan Data Transmission Format (Cont'd)*



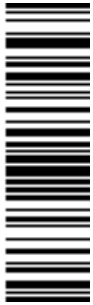
**<FN3>7B1210**

**SCAN OPTIONS**



**<FN3>6A157207 1**

**<PREFIX> <DATA> <SUFFIX>**



**<FN3>6A13071**

**<DATA> <SUFFIX>**

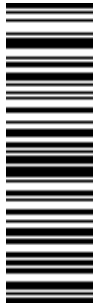
**DATA AS IS**

**<FN3>6A110**

## *Scan Data Transmission Format (Cont'd)*

**<FN3>4**

**ENTER**



**<FN3>6A13720**

**<PREFIX> <DATA>**

**CANCEL**

**<FN3>6Q**

# Scanning Options

## *Beep After Good Decode*

Determine if the scanner's beeper sounds during normal scanning. In most cases the scanner operates with the beeper enabled. The beeper always operates during parameter menu scanning and indicates error conditions. See *Beeper Definitions* on page 3-9.

Select whether or not to beep after a successful decode. If you select Beep After Good Decode, set the beeper tone or frequency.

**<FN3>1040021**

**BEEP AFTER GOOD DECODE**

**DO NOT BEEP AFTER GOOD DECODE**

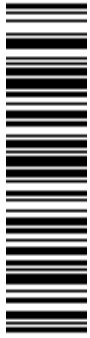
**<FN3>1040020**

## *Beeper Tone/Frequency*

This parameter sets the decode beep frequency/tone — low, medium, or high.

**<FN3>2050E02**

**LOW FREQUENCY**



**<FN3>2050E01**

**MEDIUM FREQUENCY**

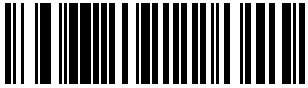
**HIGH FREQUENCY**

**<FN3>2050E00**

## Decode Attempt Duration

This sets the maximum time decode processing continues during a scan attempt. This is programmable in 0.1 second increments from 0.1 to 9.9 seconds. The recommended interval is between 1.0 and 5.0 seconds. If DECODE ATTEMPT DURATION INFINITE is selected, no timeout occurs; the scanner continues to attempt to decode.

1. Scan a bar code below.
2. If the first bar code was scanned, scan two bar codes from *Numeric Bar Codes* beginning on page 5-96 which correspond to the desired interval, in 0.1 second increments.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98**.**



**<FN3>3050020163**

**DECODE ATTEMPT DURATION**

**DECODE ATTEMPT DURATION INFINITE**

**<FN3>20500FF**

## Trigger Modes

This parameter controls scanner triggering. The following options are available:

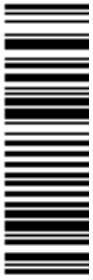
- In Level trigger mode, a trigger pull (external hardware or scanner) activates the laser and decode processing. The laser remains on and decode processing continues until a trigger release, a valid decode, or the Decode Attempt Timeout is reached.
- In Pulse trigger mode, a trigger pull (external hardware or scanner) activates the laser and decode processing. The laser remains on and decode processing continues until a valid decode or the Decode Attempt Timeout is reached.
- Blinking Laser mode is an internal trigger mode that cycles the scanning laser at approximately a 50% duty cycle after no decode attempts have occurred for 30 seconds. This extends laser life.
- Continuous mode is an internal trigger mode that operates the scanner at a 100% duty cycle. The scanner continuously attempts to decode a bar code even if none are present. This mode is suited for time-critical applications where bar codes move past the scanner.

## *Trigger Modes (Cont'd)*

Select the preferred trigger mode.

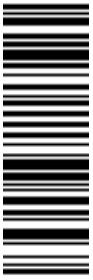
**<FN3>2050200**

**LEVEL**



**<FN3>2050202**

**PULSE**



**<FN3>2050207**

**BLINKING LASER MODE**

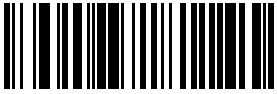
**CONTINUOUS LASER MODE**

**<FN3>2050204**

## *Scanning Modes*

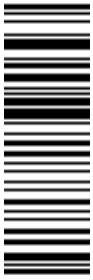
This parameter is used to select the preferred scanning mode. See *LS 6800 Scanning Mode Options* on page 3-1 for descriptions of these modes.

Select a scanning mode.



**<FN3>2050901**

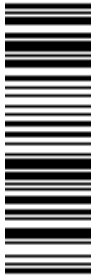
**SMART RASTER**



**<FN3>2050904**

**SLAB ONLY RASTER**

**PROGRAMMABLE RASTER**



**<FN3>2050903**

**ALWAYS RASTER**

**<FN3>2050902**

## Scanning Modes (Cont'd)

**<FN3>2050905**

**LINE ONLY PATTERN**

## *Programmable Raster Height and Raster Expansion Speed*

The height of the laser raster pattern and the rate at which the raster expands to that height can be programmed by using the two associated bar codes, **Raster Height** and **Raster Expansion Speed**. Note that the actual height and expansion rate are directly, though not linearly, proportional to their respective parameter values.

This parameter is intended for very specific applications, and is usually not necessary. Note also that these parameters are in effect only when Programmable Raster or Always Raster is enabled.

Select the laser pattern's height and/or rate of expansion.

1. Scan the bar code for either RASTER HEIGHT or RASTER EXPANSION SPEED below.
2. To represent a two-digit value, scan two bar codes from *Numeric Bar Codes* beginning on page 5-96. Valid values are between 01 and 15.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98.

**<FN3>30N032010F**

**RASTER HEIGHT (Default 15)**

**RASTER EXPANSION SPEED (Default 11)**

**<FN3>30N022010F**

## *Aiming Modes*

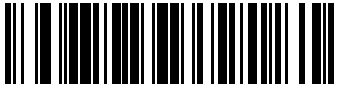
Select an aiming mode: aiming with a dot pattern, or with a slab raster pattern. Note that aiming modes do not work with the Always Raster scanning option.

### Aiming Dot Option

A trigger pull creates the single dot aiming pattern, which lasts for a fixed interval. This dot can easily be seen in outdoor or high ambient light environments. A slab raster pattern or an open raster pattern appears next, depending on the programmed scanning option. There are two programmable timeout periods for this option — normal and extended.

### Slab Raster Option

A trigger pull creates the slab raster pattern. If the target is a 1-D bar code, the pattern never gets beyond a slab raster. If the target bar code is PDF417, the pattern opens up to an optimized raster pattern as soon as the scanner is properly aligned over the bar code.

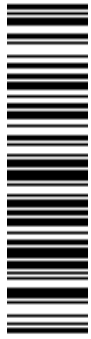


**<FN3>2051B01**

**AIMING DOT**
**(Normal Timeout)**

**SLAB RASTER**

**<FN3>2051B00**



**AIMING DOT**
**(Extended Timeout)**

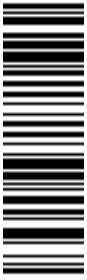
**<FN3>2051B02**

## *Time Delay to Low Power Mode*

Select the time the scanner remains continually on after any scanning attempt. If another bar code is not presented within the programmed time, the laser blinks to save power. This parameter only applies to Continuous and Blinking modes.

**<FN3>2051000**

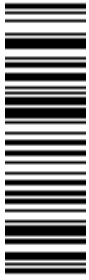
**30 SECOND DELAY**



**<FN3>2051002**

**2 MINUTE DELAY**

**1 MINUTE DELAY**



**<FN3>2051001**

**3 MINUTE DELAY**

**<FN3>2051003**

# Timeout Options

## Timeout Between Decodes, Same Symbol
## Timeout Between Decodes, Different Symbols

Select the timeouts between decodes for the same or different symbols. This is programmable in 0.1 second increments from 0.0 to 9.9 seconds. The recommended interval for the same symbol option is 0.6 seconds.

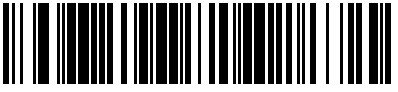
1. Scan the option bar code you wish to set.
2. Scan two bar codes from *Numeric Bar Codes* beginning on page 5-96 which correspond to the desired interval, in 0.1 second increments.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98**.**

**<FN3>3050120063**

**TIMEOUT BETWEEN DECODES - THE SAME SYMBOL**

**TIMEOUT BETWEEN DECODES - DIFFERENT SYMBOLS**

**<FN3>3050D20063**

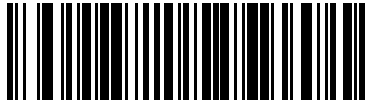


# Security Options

## *Decode UPC/EAN Supplemental Redundancy*

With Autodiscriminate UPC/EAN Supplementals selected, this option adjusts the number of times a symbol without supplementals is decoded before transmission. The range is from two to twenty times. Five or above is recommended when decoding a mix of UPC/EAN symbols with and without supplementals, and the autodiscriminate option is selected.

1. Scan the bar code below to select a decode redundancy value.
2. Scan two numeric bar codes from *Numeric Bar Codes* beginning on page 5-96. Single digit numbers must have a leading zero.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98.



**<FN3>305182021E**

**DECODE UPC/EAN SUPPLEMENTAL REDUNDANCY**

# Linear Code Type Security Level

**Note:** *Does not apply to Code 128*

The LS 6800 offers four levels of decode security for linear code types (e.g. Code 39, Interleaved 2 of 5). Higher security levels are selected for decreasing levels of bar code quality. As security levels increase, the scanner's aggressiveness decreases.

Select the security level appropriate for your bar code quality.

## Linear Security Level 1

The following code types must be successfully read twice before being decoded:

| Code Type | Length |
|---|---|
| Codabar | All |
| MSI Plessey | 4 or less |
| D 2 of 5 | 8 or less |
| I 2 of 5 | 8 or less |

## Linear Security Level 2

All supported code types must be successfully read twice before being decoded.

## Linear Security Level 3

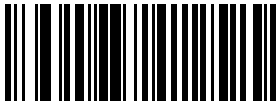
Code types other than the following must be successfully read twice before being decoded. The following codes must be read three times:

| Code Type | Length |
|---|---|
| MSI Plessey | 4 or less |
| D 2 of 5 | 8 or less |
| I 2 of 5 | 8 or less |
| Codabar | 8 or less |

## *Linear Code Type Security Level*

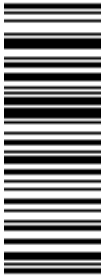
### Linear Security Level 4

All supported code types must be successfully read three times before being decoded.



**<FN3>2051501**

**LINEAR SECURITY LEVEL 1**

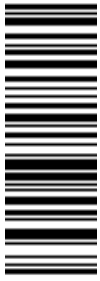


**<FN3>2051504**

**LINEAR SECURITY LEVEL 4**

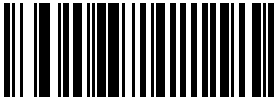
**LINEAR SECURITY LEVEL 3**

**<FN3>2051503**



**LINEAR SECURITY LEVEL 2**

**<FN3>2051502**



## Bi-directional Redundancy

This parameter is only valid when a *Linear Code Type Security Level* has been enabled. When this parameter is enabled, a bar code must be successfully scanned in both directions (forward and reverse) before being decoded.

**<FN3>1040311**

**ENABLE BI-DIRECTIONAL REDUNDANCY**

**DISABLE BI-DIRECTIONAL REDUNDANCY**

**<FN3>1040310**

## *UPC/EAN Security Level*

The LS 6800 offers four levels of decode security for UPC/EAN bar codes. Increasing levels of security are provided for decreasing levels of bar code quality. There is an inverse relationship between security and scanner aggressiveness, so be sure to choose only that level of security necessary for any given application.

### UPC/EAN Security Level 0

This is the default setting which allows the scanner to operate in its most aggressive state, while providing sufficient security in decoding "in-spec" UPC/EAN bar codes.

### UPC/EAN Security Level 1

As bar code quality levels diminish, certain characters become prone to mis-decodes before others (i.e., 1, 2, 7, 8). If you are experiencing mis-decodes of poorly printed bar codes, and the mis-decodes are limited to these characters, select this security level.

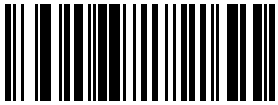
### UPC/EAN Security Level 2

If you are experiencing mis-decodes of poorly printed bar codes, and the mis-decodes are not limited to characters 1, 2, 7, and 8, select this security level.

### UPC/EAN Security Level 3

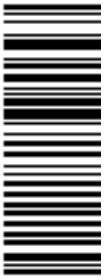
If you have tried Security Level 2, and are still experiencing misdecodes, select this security level. Be advised, selecting this option is an extreme measure against mis-decoding severely out of spec bar codes. Selection of this level of security significantly impairs the decoding ability of the scanner. If this level of security is necessary, you should try to improve the quality of your bar codes.

# UPC/EAN Security Level (Cont'd)



**<FN3>2051100**

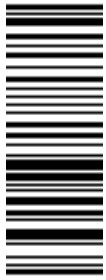
**UPC/EAN SECURITY LEVEL 0**



**<FN3>2051102**

**UPC/EAN SECURITY LEVEL 2**

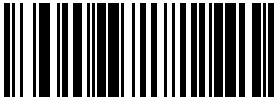
**UPC/EAN SECURITY LEVEL 3**



**<FN3>2051103**

**UPC/EAN SECURITY LEVEL 1**

**<FN3>2051101**



# RS-232C Options

**Note:** *If using a Synapse Cable, consult the Synapse Interface Guide.*

In Single Port mode, the host device is connected to the **HOST** port.

## RS-232C Host Types

This parameter selects an RS-232C host interface option. When you select a specific host, this automatically sets RS-232C parameters to those of that host (see Table 5-4 and Table 5-5).

**Table 5-4. Standard RS-232, Nixdorf Host Types**

| Parameter | Std. RS-232 | Nixdorf Mode A | Nixdorf Beetle |
|---|---|---|---|
| Xmit Code ID | No | Yes | Yes |
| Data Trans. Error | Data | Data/Suffix | Data/Suffix |
| Suffix | None | CR(1013) | CR(1013) |
| Baud Rate | 9600 | 9600 | 9600 |
| Parity | None | Odd | Odd |
| H/W Handshaking | None | None | RTS/CTS Opt. 3 |
| S/W Handshaking | None | None | None |
| Serial Resp. T/O | 2.0 Sec. | 9.9 Sec. | 9.9 Sec. |
| Stop Bit Select | One | One | One |
| ASCII Format | 8-Bit | 8-Bit | 8-Bit |
| Beep on <Bel> | Disabled | Disabled | Disabled |
| RTS Line State | Low Always | Low Always | Low Always |
| Special Notes | None | None | Output 0x60 for all CR except last one |

## RS-232C Host Types (Cont'd)

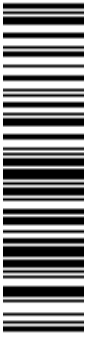
**Table 5-5. Fujitsu, ICL, PDT 3300 Host Types**

| Parameter | Fujitsu | ICL | PDT 3300 |
|---|---|---|---|
| Xmit Code ID | Yes | Yes | Yes |
| Data Trans. Error | Data/Suffix | Data/Suffix | STX/Data/ETX/ LRC |
| Suffix | CR(1013) | CR(1013) | None |
| Baud Rate | 9600 | 9600 | 19200 |
| Parity | None | Even | Even |
| H/W Handshaking | None | RTS/CTS Opt. 3 | RTS/CTS |
| S/W Handshaking | None | None | ACK/NAK |
| Serial Resp. T/O | 2.0 Sec. | 9.9 Sec. | 9.9 Sec. |
| Stop Bit Select | One | One | One |
| ASCII Format | 8-Bit | 8-Bit | 7-Bit |
| Beep on <Bel> | Disabled | Disabled | Disabled |
| RTS Line State | Low Always | High Always | Low Always |
| Special Notes | None | None | None |

## *RS-232C Host Types (Cont'd)*

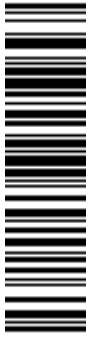
**<FN3>2050A37**

**STANDARD RS-232C**



**<FN3> 2050A8A**

**NIXDORF RS-232C**



**<FN3>2050A89**

**ICL RS-232C**

**FUJITSU RS-232C**

**<FN3>2050A8B**

## RS-232C Host Types (Cont'd)

**<FN3>2050A8C**

**BEETLE RS-232C**

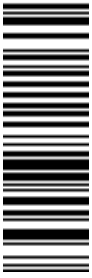
## *Baud Rate*

Baud rate is the number of bits of data transmitted per second. The scanner's respective baud rate settings should match the data rate settings of the host and auxiliary devices. If not, data may not reach the devices or may reach them in distorted form.

Set the baud rate for RS-232C transmission.

**<FN3>2090100**

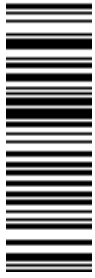
**110**



**<FN3>2090102**

**600**

**300**



**<FN3>2090101**

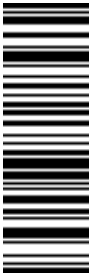
**1200**

**<FN3>2090103**

## *Baud Rate (Cont'd)*

**<FN3>2090104**

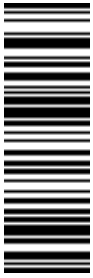
**2400**



**<FN3>2090106**

**9600**

**4800**



**<FN3>2090105**

**19200**

**<FN3>2090107**

## *Baud Rate (Cont'd)*

**<FN3>2090108**

**38400**

## Parity

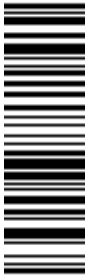
A parity check bit is the most significant bit of each ASCII coded character and is helpful in detecting transmission errors.

If you select **ODD** parity, the parity bit has a value 0 or 1, based on data, to ensure that an odd number of 1 bits are contained in the coded character. If you select **EVEN** parity, the parity bit has a value 0 or 1, to ensure that an even number of 1 bits are contained in the coded character. If **MARK** is selected, the parity bit is always 1, while if **SPACE** is selected, the parity bit is always 0.

Select the parity type according to host requirements.

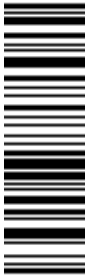
**<FN3>2090300**

**ODD**



**<FN3>2090302**

**MARK**



**<FN3>2090303**

**SPACE**

**EVEN**

**<FN3>2090301**

## *Parity (Cont'd)*

**<FN3>2090304**

**NONE**

## Check Parity

Select whether or not the parity of received characters is checked. The type of parity used is selectable through the **PARITY** parameter.

Select whether or not to check parity for RS-232C transmissions.

**<FN3>1080011**

**CHECK PARITY**

**DO NOT CHECK PARITY**

**<FN3>1080010**

## *Stop Bit Select*

The stop bit(s) at the end of each transmitted character marks the end of transmission of one character and prepares the receiving device for the next character in the serial data stream. The number of stop bits (one or two) selected depends on the number the receiving terminal is programmed to accommodate. Set the number of stop bits to match host requirements.

Select the desired number of stop bits for RS-232C communications.

**<FN3>2090201**

**1 STOP BIT**

**2 STOP BITS**

**<FN3>2090202**

## Hardware Handshaking

The LS 6800 scanner uses an RS-232C data interface. The RS-232C port is designed to operate either with or without the hardware handshaking lines **RTS**, *Request to Send*, and **CTS**, *Clear to Send*.

If **RTS**/**CTS** handshaking is selected, scan data is transmitted according to the following sequence:

- The scanner reads the **CTS** line for activity. If **CTS** is asserted, the scanner waits up to one second for the host to negate the **CTS** line. After two seconds, if the **CTS** line is still asserted, the scanner sounds a transmit error and any scanned data is lost.
- When the **CTS** line is negated, the scanner asserts the **RTS** line and waits for two seconds for the host to assert **CTS**. When the host asserts **CTS**, data is transmitted. If the **CTS** line is not asserted after two seconds, the scanner sounds a transmission error and discards the data.
- When data transmission is complete, the scanner negates **RTS** 10 msec after sending the last character.
- The host should respond by negating **CTS**. The scanner checks for a negated **CTS** upon the next transmission of data.

During the transmission of data, the **CTS** line should be asserted. If **CTS** is asserted for less than 50 ms between characters, the transmission is aborted. The scanner sends a transmit error, and the data is discarded.

- If the above communications sequence fails, the scanner issues an error beep. Data is lost and must be rescanned.

---

**Note:** *The DTR signal is jumpered active.*

---

## *Hardware Handshaking (Cont'd)*

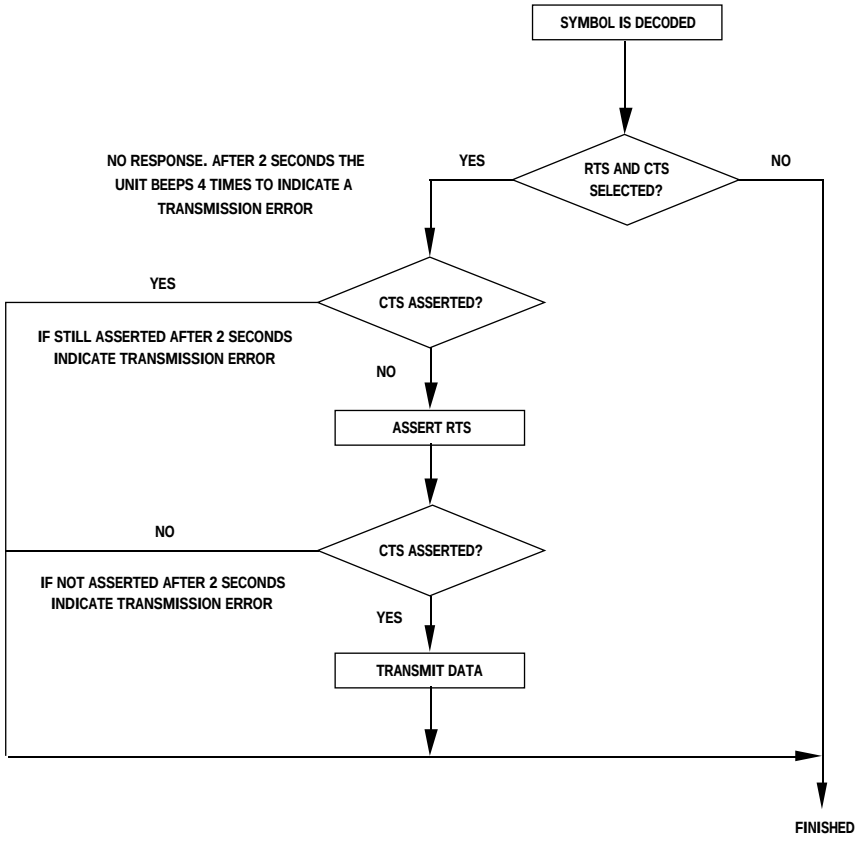



**Figure 5-1. Hardware Handshaking**

## Hardware Handshaking (Cont'd)

Hardware handshaking allows you to check the readiness of the receiving device before data is transmitted. If the receiving device is periodically occupied with other tasks, hardware handshaking is needed to prevent loss of transmitted data. Hardware handshaking may not be used in conjunction with software handshaking.

Select whether the scan data is to be transmitted as soon as it is available or whether transmission follows the RTS/CTS procedure. For RTS/CTS, also select the appropriate option.

- **RTS/CTS Option 1**. The scanner asserts **RTS** before transmitting and ignores the state of **CTS**. The scanner deasserts **RTS** when the transmission is incomplete.
- **RTS/CTS Option 2**. **RTS** is always high or low (user-programmed logic level). However, the scanner waits for **CTS** to be asserted before transmitting data. If **CTS** is not asserted within two seconds, the scanner issues an error indication and discards the data.
- **RTS/CTS Option 3**. The scanner asserts **RTS** prior to any data transmission, regardless of the state of **CTS**. The scanner waits up to two seconds for **CTS** to be asserted. If **CTS** is not asserted during this time, the scanner issues an error indication and discards the data. The scanner deasserts **RTS** when transmission is incomplete.

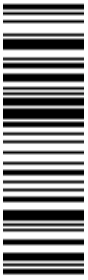
**<FN3>2090500**

**NONE**

## *Hardware Handshaking (Cont'd)*

**RTS/CTS**

**<FN3>2090501**

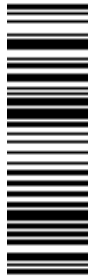


**<FN3>2090503**

**RTS/CTS Option 2**

**<FN3>2090502**

**RTS/CTS Option 1**



**RTS/CTS Option 3**

**<FN3>2090504**

## Software Handshaking

This parameter offers control of the data transmission process. It may be used instead of, but not in conjunction with, hardware handshaking. **ACK**/**NAK** handshaking may be combined with **ENQ** handshaking.

- No Software Handshaking
- **ACK/NAK Only**
  The **ACK**/**NAK** option checks the success or failure of transmission. The scanner expects one of the following host responses after a data transmission:
  <**ACK**> - Acknowledges a valid and successful transmission.
  <**NAK**> - Indicates a problem with the transmission.
  Whenever a <**NAK**> is received, the unit retransmits the same data and awaits an **ACK**/**NAK** response. After three unsuccessful attempts to transmit the same data, the scanner aborts any further communication attempts on that message. Transmission error is indicated by the unit sounding 4 short beeps.
- **ENQ ONLY**
  To ensure that data transmission occurs only when the host is ready to receive, the **ENQ** option requires the host to request data before it is transmitted there.
  When the **ENQ** option is enabled, the scanner waits for an **ENQ** (Enquire character) from the host before it transmits data. The scanner must receive an **ENQ** from the host within 2 seconds after the last activity or 4 short beeps sound to indicate a transmission error. The unit is now ready to scan again.
- **ACK/NAK with ENQ**
  This combines both handshaking options.
- **XON/XOFF**

## *Software Handshaking (Cont'd)*

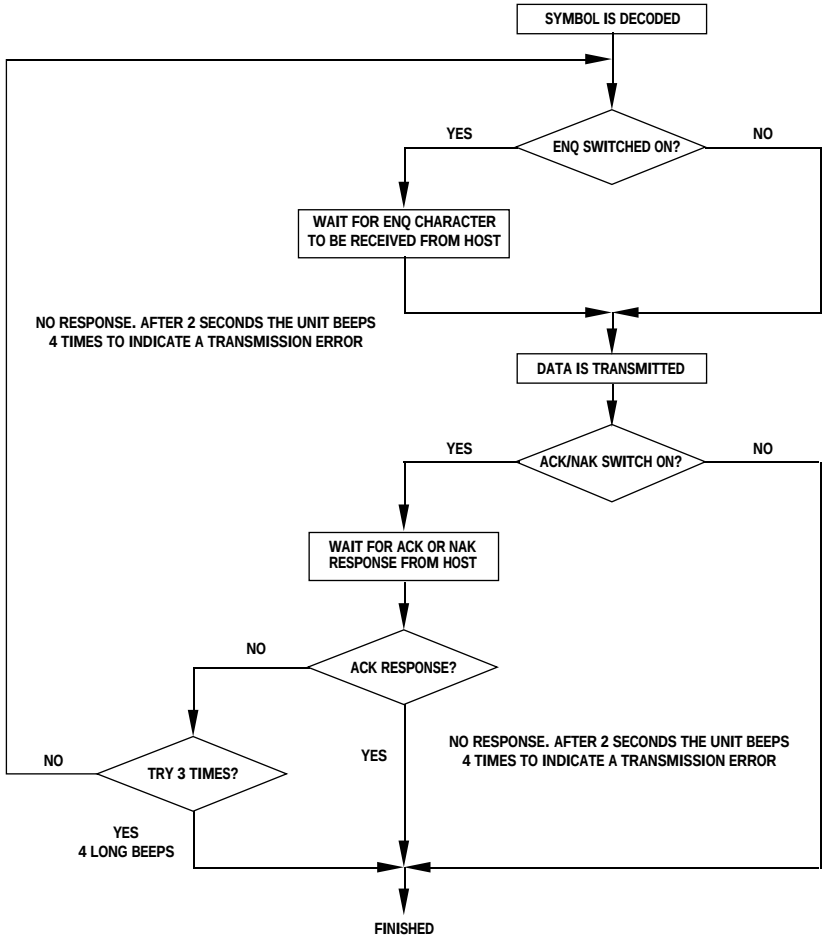


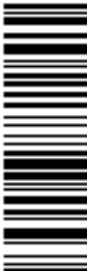

**Figure 5-2. Software Handshaking**

## *Software Handshaking (Cont'd)*

Select the type of RS-232C software handshaking protocol.

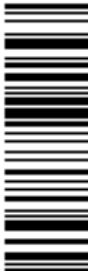
**<FN3>2090400**

**NONE**



**<FN3>2090401**

**ACK/NAK**



**<FN3>2090403**

**ENQ ONLY**

**ACK/NAK with ENQ**

**<FN3>2090402**

## *Software Handshaking (Cont'd)*

**<FN3>2090404**

**XON/XOFF**

## Data Transmission (7 or 8-Bit ASCII Format)

This parameter determines whether data transmits in 7-bit or 8-bit ASCII format. Select this parameter according to the requirement of the receiving device. The default value is 8-bit ASCII.

**<FN3>2090707**

**7-BIT**

**8-BIT**

**<FN3>2090708**

## *Intercharacter Delay*

Select the intercharacter delay option matching host device requirements. The intercharacter delay gives the host system time to service its receiver and perform other tasks between characters. Select from no delay to a 99 ms delay between the transmission of each character.

To set a host communications intercharacter delay:

1. Scan the **INTERCHARACTER DELAY** bar code below.
2. Scan two bar codes from *Numeric Bar Codes* beginning on page 5-96 which represent the desired delay. You must always scan two bar codes.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98**.**

**<FN3>30C0C0063**

**INTERCHARACTER DELAY**

## Host Serial RTS Line State

Scan a bar code below to set the host serial RTS line state to either high or low.

**<FN3>1080040**

**HOST: LOW RTS**

**HOST: HIGH RTS**

**<FN3>1080041**

## *Serial Response Timeout*

This parameter determines the maximum period allowed to elapse before the scanner assumes end of transmission. The delay period can range from 0 to 9.9 seconds.

To set a serial (RS-232C) response time-out:

1. Scan the **SERIAL RESPONSE TIME-OUT** bar code below.
2. Scan two bar codes from *Numeric Bar Codes* beginning on page 5-96 which represent the desired time-out. You must always scan two bar codes.
3. If you make an error, or wish to change your selection, scan **CANCEL** on page 5-98**.**

**<FN3>3090020063**

**SERIAL RESPONSE TIME-OUT**

## Beep on <BEL>

When this parameter is enabled, the scanner issues a beep when a <**BEL**> character is detected on the RS-232C serial data line. <**BEL**> indicates an illegal entry or other important event. If this parameter is enabled and the system is in battery mode, a 100 ms delay is placed at the end of the decode transmission.

Scan a bar code below to enable or disable this option.

**<FN3>1080001**

**BEEP ON <BEL> CHARACTER**

**DO NOT BEEP ON <BEL> CHARACTER**

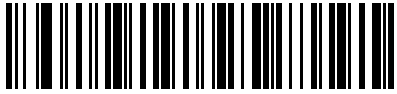
**<FN3>1080000**

## *Flash Memory Programming*

The LS 6800 series scanners are equipped with flash memory, which can be updated by a field service technician or at a repair depot. Through this process, software revisions and features can be added easily to existing units.

## *FAT (Factory Acceptance Test) Mode*

Scanning this bar code reveals information about the scanner's software configuration that may be useful during diagnostics. You must recycle power to return to normal functionality.

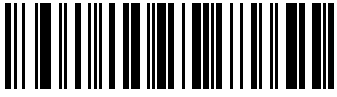


**<FN3>FLASHMODE**

**FLASH MODE**

**FINAL ACCEPTANCE TEST MODE**

**<FN3>FATMODE**

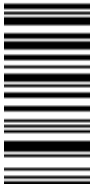


# Numeric Bar Codes

Use these bar codes when programming the following parameters: **Code Lengths, Pause Duration, Decode Attempt Duration, Prefix/Suffix Values**, **Timeout Between Decodes, Intercharacter Delay, Serial Response Timeout**.

**<FN3>A0**

**0**

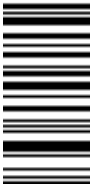


**<FN3>A3**

**3**

**1**

**<FN3>A1**



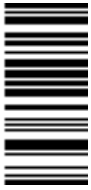
**2**

**<FN3>A2**

# Numeric Bar Codes (Cont'd)

**<FN3>A4**

**4**

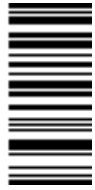


**<FN3>A7**

**7**

**5**

**<FN3>A5**



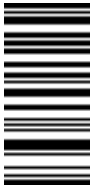
**6**

**<FN3>A6**

# Numeric Bar Codes (Cont'd)

<FN3>A8

8



<FN3>A9

9

CANCEL

<FN3>A-

# Special Macro PDF Features

This section discusses programmable Macro PDF features supported by the LS 6800. Macro PDF concatenates multiple PDF symbols into one file. The LS 6800 decodes symbols that are encoded with this feature, however the 64K version cannot buffer large sets of Macro PDF symbols.

*Caution*

When printing, keep each Macro PDF sequence separate, for each sequence has unique identifiers. Do not mix bar codes from several Macro PDF sequences, even if they encode the same data. When scanning Macro PDF sequences, scan the entire Macro PDF sequence without interruption. If you scan a mixed sequence, you get two long low beeps (Lo Lo) for inconsistent file ID or inconsistent symbology error.

Before programming these special features, follow the physical setup instructions in Chapter 2, *Setup*, then program the required generic decode and data transmission parameters from Chapter 6, *Advanced Data Formatting*. Use the same programming method for setting Macro PDF features as used for standard features.

Note that all parameter settings are stored in non-volatile memory and so are retained after power down.

## Macro PDF Transmission / Decode Modes

Select only one of the four options below for handling the decoding of Macro PDF. The decoder can handle up to 64K bytes of decoded data, including any optional field information, except in the mode *Transmit Each Macro PDF Symbol Decoded in Sequential Order*, where there is no limit to the size of the Macro PDF set. Select one of these four options for decoding Macro PDF sequences.

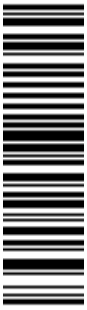
- **Buffer All Symbols / Transmit Macro PDF When Complete:** This activates transmission of all decode data from an entire Macro PDF sequence. Transmission does not occur until the entire Macro PDF sequence is scanned and decoded. This is the default option.

  If the decode data exceeds the limit of 64K bytes, there is no transmission because the entire sequence has not been scanned. Use the parameter Flush Macro PDF Buffer to purge the buffer.
- **Transmit Any Symbol in Set / No Particular Order**: This causes transmission of data from each Macro PDF417 symbol as decoded, whether it is in sequential order or not.
- **Scan in Sequence Only / Transmit in Sequence Without Buffering**: This causes transmission of data from each symbol within the Macro PDF sequence as decoded, provided the Macro PDF symbols are scanned in order. If you do not scan the symbols in order, an error occurs.
- **Buffer Scans Out of Order / Transmit Scans in Order**: When enabled, decode data from each symbol within the Macro PDF sequence is transmitted when decoded, provided that the Macro PDF symbols are scanned in order. Decode data from symbols out of order in the Macro PDF sequence is buffered. The decoder can buffer up to 64K bytes.

## *Macro PDF Transmission / Decode Modes (Cont'd)*

**<FN3>20N0000**

**BUFFER ALL SYMBOLS / TRANSMIT WHEN COMPLETE - DEFAULT**



**<FN3>20N0002**

**SCAN IN SEQUENCE ONLY / TRANSMIT IN SEQUENCE WITHOUT BUFFERING**

**TRANSMIT ANY SYMBOL IN SET / NO PARTICULAR ORDER**

**<FN3>20N0001**

**BUFFER SCANS OUT OF ORDER/ TRANSMIT SCANS IN ORDER**

**<FN3>20N0003**

## Transmit Each Symbol in Codeword Format

Enabling this activates transmission of each PDF symbol as directly decoded data codewords, whether that symbol is part of a macro PDF sequence or not. Note that data is output as *codeword values — not as interpreted data*.

"Codeword values" is an ASCII representation of a number from 000 to 928 for each codeword, preceded by an escape character. This escape character is a backslash by default, but the user may change this value. For example, the codeword value 005 is sent to the host in the form of **\005** for GLIs, and **\C005C** for ECIs. This output format is based on the *AIM USA Uniform Symbology Specification* for PDF417 (1994).

All output codewords take up exactly 4 characters for GLIs and 6 characters for ECIs. However, there may be nondecodable characters in the PDF symbol, such as a GLI sequence. This special codeword sequence activates a certain kind of interpretation to the encoded data. Non-decodable codewords like GLIs are embedded in the output stream just like any other codeword, e.g., \927\001.

Because GLIs are indistinguishable from other codewords in the output data stream, the host must intelligently recognize them as GLIs and process their interpretations.

Note that when a macro PDF sequence is transmitted, the last character in the last block of data transmitted is always**\922** (if selected). This indicates the end of that macro PDF transmission.

## *Transmit Each Symbol in Codeword Format (Cont'd)*

Scan the appropriate bar code to enable or disable this option.

**<FN3>10M0001**

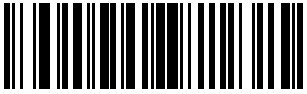
**ENABLE TRANSMIT IN CODEWORD FORMAT**

**DISABLE TRANSMIT IN CODEWORD FORMAT**

**<FN3>10M0000**

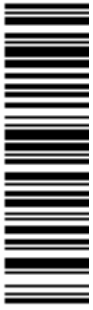
## Escape Characters

This enables the backslash (\) character as an Escape character for systems that can process transmissions containing special data sequences. Scan a bar code below to either format special data (e.g., GLI escapes, MacroPDF417 Control Block optional fields) according to the GLI (Global Label Identifier) protocol or the ECI (Extended Channel Interpretation) protocol, or to disable this parameter.



**<FN3>20N0401**

**ECI PROTOCOL**



**<FN3>20N0400**

**NONE**

**GLI PROTOCOL**

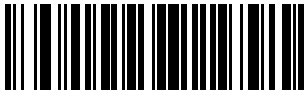
**<FN3>20N0402**

## *Delete Character Set ECIs*

This parameter enables the scanner to delete any escape sequences representing Character Set ECIs (also known as GLIs) from its buffer before transmission. In many receiving systems, Character Set ECIs can be removed without affecting the way data is displayed or processed.

When deletion is selected, the scanner transmits data from PDF417 and MicroPDF417 bar codes containing Character Set ECIs, even when the ECI Protocol is disabled.

Scan a bar code to delete or transmit character set ECIs.

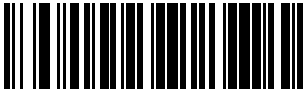


**<FN3>10M0151**

**DELETE CHARACTER SET ECIs**

**TRANSMIT CHARACTER SET ECIs**

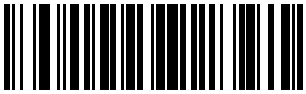
**<FN3>10M0150**



## ECI Decoder

This parameter enables the scanner to interpret any Extended Channel Interpretations (ECIs) that are supported by the scanner firmware. This parameter has no effect on symbols that were not encoded using ECIs. This verison of the product supports ECIs 000900 through 000913, used for efficient encoding of Common Data Syntax Format 00-99. If this parameter is disabled, and a symbol is scanned that was encoded using an ECI escape, the scanner transmits the ECI escape followed by the uninterpreted data.

Scan a bar code to enable or disable this option.

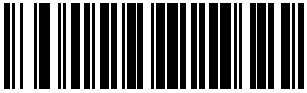


**<FN3>10M0161**

**ENABLE ECI DECODER**

**DISABLE ECI DECODER**

**<FN3>10M0160**



## *Transmit Unknown Codewords*

This enables using the output codeword format for transmitting any non-GLI or non-Macro PDF codeword. If this is *not* enabled and an unknown codeword is found, a decode error beep sounds.

**<FN3>10M0131**

**TRANSMIT UNKNOWN CODEWORDS**

**<FN3>10M0130**

**DO NOT TRANSMIT UNKNOWN CODEWORDS**

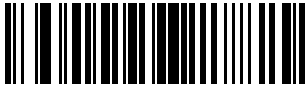
## Transmit Macro PDF User-Selected Fields

When enabled, the following parameters cause transmission of the specified field in subsequently scanned Macro PDF417 symbols. Unless transmission of a specific field is enabled, it is not transmitted. The options cannot be changed in the middle of a Macro PDF set entry. All user-selected fields are prefixed by \923 for GLIs, and \C923C for ECIs. Tags and examples in the following parameters demonstrate GLI protocol, but the ECI tag (\C923C) can be used instead if ECI protocol is enabled.

- **Transmit File Name**: Activates transmission of the file name field. The field character tag is \923\000. For example, the filename MANHOURS.WK1 is sent as: \923\000MANHOURS.WK1.
- **Transmit Block Count**: This activates transmission of the block count field. The field character tag is \923\001. For example, the field may be: \923\0011856.
- **Transmit Time Stamp**: This activates transmission of the time stamp field. The field character tag is \923\002. For example, the field may be: \923\0022123443243234.
- **Transmit Sender**: Activates transmission of the sender field. The field character tag is \923\003. For example, the field may be: \923\003Symbol TechnologiesHoltsville, NY.
- **Transmit Addressee**: Activates transmission of the addressee field. The field character tag is \923\004. For example, the field may be: \923\004AIM USA.
- **Transmit File Size**: Activates transmission of the file size field. The field character tag is \923\005. For example, the field may be: \923\005179234.
- **Transmit Checksum**: Activates transmission of the checksum field. The field character tag is \923\006. For example, the field may be: \923\00663823.
- **Transmit Macro PDF Control Header**: Activates transmission of the control header, which contains the segment index and the file ID. For example, the field may be: \92800000\725\120\343. The five digits after the \928 are the segment index (or block index), and \725\120\343 is the file ID.
- **Enable / Disable Last Block Marker**: This enables marking the last block in the set by the codeword \922.

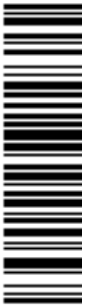
## *Transmit Macro PDF User-Selected Fields (Cont'd)*

Enable or disable by scanning the appropriate bar code.



**<FN3>10M0011**

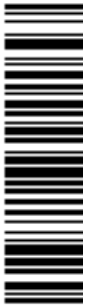
**ENABLE FILE NAME TRANSMIT**



**<FN3>10M0021**

**ENABLE BLOCK COUNT TRANSMIT**

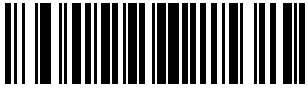
**DISABLE FILE NAME TRANSMIT**



**<FN3>10M0010**

**DISABLE BLOCK COUNT TRANSMIT**

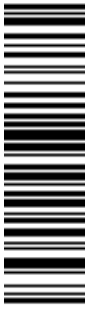
**<FN3>10M0020**



## *Transmit Macro PDF User-Selected Fields (Cont'd)*

**<FN3>10M0031**

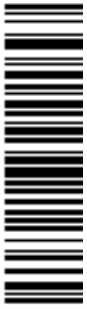
**ENABLE TIME STAMP TRANSMIT**



**<FN3>10M0041**

**ENABLE SENDER TRANSMIT**

**DISABLE TIME STAMP TRANSMIT**



**<FN3>10M0030**

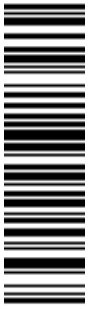
**DISABLE SENDER TRANSMIT**

**<FN3>10M0040**

## *Transmit Macro PDF User-Selected Fields (Cont'd)*

**<FN3>10M0051**

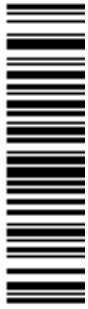
**ENABLE ADDRESSEE TRANSMIT**



**<FN3>10M0061**

**ENABLE FILE SIZE TRANSMIT**

**DISABLE ADDRESSEE TRANSMIT**

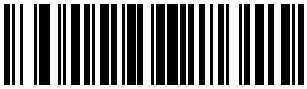


**<FN3>10M0050**

**DISABLE FILE SIZE TRANSMIT**

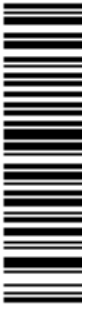
**<FN3>10M0060**

# Transmit Macro PDF User-Selected Fields (Cont'd)



**<FN3>10M0071**

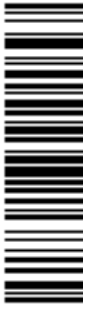
**ENABLE CHECKSUM TRANSMIT**



**<FN3>10M0101**

**ENABLE MACRO PDF CONTROL HEADER TRANSMIT**

**DISABLE CHECKSUM TRANSMIT**



**<FN3>10M0070**

**DISABLE MACRO PDF CONTROL HEADER TRANSMIT**

**<FN3>10M0100**

## *Transmit Macro PDF User-Selected Fields (Cont'd)*

**<FN3>10M0121**

**ENABLE LAST BLOCK MARKER**

**DISABLE LAST BLOCK MARKER**

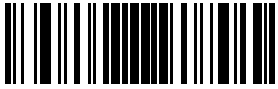
**<FN3>10M0120**

## Abort Macro PDF Entry

This clears all currently-stored Macro PDF data in the buffer without transmission and aborts from Macro PDF mode.

## Flush Macro PDF Entry

This flushes the buffer of all decoded Macro PDF data stored to that point, transmits it to the host, and aborts from Macro PDF mode.

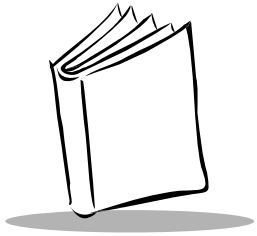


**<FN3>FLUSH**

**FLUSH MACRO PDF BUFFER**

**ABORT MACRO PDF ENTRY**

**<FN3>ABORT**



# Chapter 6
# Advanced Data Formatting

## Introduction

Advanced Data Formatting (ADF) is a means of customizing data before transmission to your host device. Scan data can be edited to suit your particular requirements.

ADF can be implemented through scanning a related series of bar codes, which appear in Chapter 7, *ADF Bar Codes*.

## Rules: Criteria Linked to Actions

In ADF, data is customized through **rules**. These rules perform detailed actions when the data meets certain criteria. One rule may consist of single or multiple criteria applied to single or multiple actions.

For instance, a data formatting rule could be the following:

**Criteria:** *When scan data is Code 39, length 12,*
*and data at the start position is the string "129",*

**Actions:** *pad all sends with zeros to length 8,*
*send all data up to X,*
*send a space.*

If a Code 39 bar code of 1299X1559828 is scanned, the following is transmitted: 00001299<space>. If a Code 39 bar code of 1299X15598 is scanned, this rule is ignored.

The rule specifies the editing conditions and requirements before data transmission occurs.

# Using ADF Bar Codes

When you program a rule, make sure the rule is logically correct. Plan ahead before you start scanning.

To program each data formatting rule:

- **Start the Rule**. Scan the **BEGIN NEW RULE** bar code on page 7-1.
- **Criteria**. Scan the bar codes for all pertinent criteria. Criteria can include code type (e.g., Code 128), code length, or data that contains a specific character string (e.g., the digits "129"). These options are described in *Criteria* on page 6-4.
- **Actions**. Scan all actions related to, or affecting, these criteria. The actions of a rule specify how to format the data for transmission. These options are described in *Actions* on page 6-5.
- **Save the Rule**. Scan the **SAVE RULE** bar code on page 7-2. This places the rule in the "top" position in the rule buffer.
- If you make errors during this process, some special-purpose bar codes may be useful: **Erase Criteria and Start Again**, **Erase Actions and Start Again**, **Erase Previously Saved Rule**, etc.

Criteria, actions, and entire rules may be erased by scanning the appropriate bar code (see *Erase* on page 7-2).

*Beeper Definitions* on page 6-13 help guide you through the programming steps.

# Special Commands

## Begin New Rule

Scan this bar code first when programming a data formatting rule.

## Save Rule

Scan this bar code to complete a data formatting rule.

## Erase

Use these bar codes to erase criteria, actions, and rules.

## Quit Entering Rules

Scan this bar code to quit entering rules.

## Disable Rule Set

These bar codes allow you to disable particular rule sets.

# Criteria

## *Code Types*

Select any number of code types to be affected. All selected codes must be scanned in succession, prior to selecting other criteria. *If you don't select a code type, all code types will be affected.*

## *Code Lengths*

Define the number of characters the selected code type must contain. *If you don't select a code length, selected code types of any length will be affected.*

## *Message Containing A Specific Data String*

Select whether the formatting affects data that begins with a specific character or data string, or contains a specific character or data string.

### Specific String at Start

Scan this bar code, then scan the bar codes representing the desired character or characters (up to a total of 8) in the *Alphanumeric Keyboard* beginning on page 7-50.

### Specific String, Any Location

Scan this bar code, then, using the *Numeric Keypad* on page 7-12, scan a two-digit number representing the *position* (use a leading "zero" if necessary). Then scan the desired character or characters (up to a total of 8) on the *Alphanumeric Keyboard* beginning on page 7-50, followed by the **END OF MESSAGE** bar code on page 7-56.

### Any Message OK

By not scanning any bar code, all selected code types are formatted, regardless of information contained.

## *Rule Belongs To Set*

Select the set a rule belongs to.

# Actions

Select how to format the data for transmission.

## *Send Data*

Send all data that remains, send all data up to a specific character selected from the *Alphanumeric Keyboard* on page 7-50, or send the next N characters. N = any number from 1 to 254, selected from the *Alphanumeric Keyboard* on page 7-50.

## *Setup Field(s)*

Define fields as follows:

### Move Cursor To a Character

Scan the **MOVE CURSOR TO CHARACTER** bar code on page 7-17, then any printable ASCII character from the *Alphanumeric Keyboard* on page 7-50. When this is used, the cursor moves to the position after the matching character. If the character is not there, the rule fails and ADF tries the next rule.

### Move Cursor to Start of Data

Scan this bar code to move cursor to the beginning of the data.

### Move Cursor Past a Character

This parameter moves the cursor past all sequential occurrences of a selected character. Scan the **MOVE CURSOR PAST CHARACTER** bar code on page 7-17, then select a character from the *Alphanumeric Keyboard* on page 7-50. If the character is not there, the cursor does not move (i.e., has no effect).

### Skip Ahead "N" Characters

Scan one of these bar codes to select the number of positions ahead you wish to move the cursor.

### Skip Back "N" Characters

Scan one of these bar codes to select the number of positions back you wish to move the cursor.

## Send Preset Value

Send Values 1 through 6 by scanning the appropriate bar code. These values must be set using *Prefix / Suffix Values* on page A-5.

Value 1 = Scan Suffix

Value 2 = Scan Prefix

## Modify Data

Modify data in the ways listed. The following actions work for all send commands that follow it within a rule. If you program *pad zeros to length 6, send next 3 characters, stop padding, send next 5 characters,* three zeros are added to the first send, and the next send is unaffected by the padding. These options do not apply to the **Send Keystroke** or **Send Preset Value** options.

### Remove All Spaces

To remove all spaces in the send commands that follow, scan this bar code.

### Crunch All Spaces

To leave one space between words, scan this bar code. This also removes all leading and trailing spaces.

### Stop Space Removal

Scan this bar code to disable space removal.

### Remove Leading Zeros

Scan this bar code to remove all leading zeros.

### Stop Zero Removal

Scan this bar code to disable the removal of zeros.

## Pad Data With Spaces

To pad data to the left, scan the bar code containing the desired number of spaces. This parameter is activated by Send commands.

## Pad Data With Zeros

To pad data to the left, scan the bar code containing the desired number of zeros. This parameter is activated by Send commands.

## Beeps

Select a beep sequence for each ADF rule.

## Send Keystroke (Control Characters and Keyboard Characters)

Scan the “Send __” bar code for the keystroke you wish to send.

## Turn On/Off Rule Sets

Use these bar codes to turn rule sets on and off.

# ADF Bar Code Menu Example

This section provides an example of how ADF rules are entered and used for scan data.

An auto parts distribution center wants to encode manufacturer ID, part number, and destination code into their own Code 128 bar codes. The distribution center also has products that carry UPC bar codes, placed there by the manufacturer. The Code 128 bar codes have the following format:

**MMMMMPPPPPDD**

Where: M = Manufacturer ID

P = Part Number

D = Destination Code

The distribution center uses a PC with dedicated control characters for manufacturer ID <CTRL M>, part number <CTRL P>, and destination code <CTRL D>. At this center the UPC data is treated as manufacturer ID code.

The following rules need to be entered:

When scanning data of code type Code 128, send the next 5 characters, send the manufacturer ID key <CTRL M>, send the next 5 characters, send the part number key <CTRL P>, send the next 2 characters, send the destination code key <CTRL D>.

When scanning data of code type UPC/EAN, send all data, send the manufacturer ID key <CTRL M>.

To enter these rules, follow the steps below:

## Rule 1: The Code 128 Scanning Rule

| Step | Bar Code | On Page | Beep Indication |
|---|---|---|---|
| 1 | Begin New Rule | 7-1 | High High |
| 2 | Code 128 | 7-5 | High High |
| 3 | Send next 5 characters | 7-14 | High High |
| 4 | Send <CTRL M> | 7-34 | High High |
| 5 | Send next 5 characters | 7-14 | High High |
| 6 | Send <CTRL P> | 7-35 | High High |
| 7 | Send next 2 characters | 7-14 | High High |
| 8 | Send <CTRL D> | 7-33 | High High |
| 9 | Save Rule | 7-2 | High Low High Low |

## Rule 2: The UPC Scanning Rule

| Step | Bar Code | On Page | Beep Indication |
|---|---|---|---|
| 1 | Begin New Rule | 7-1 | High High |
| 2 | UPC/EAN | 7-6 | High High |
| 3 | Send all remaining data | 7-14 | High High |
| 4 | Send <CTRL M> | 7-34 | High High |
| 5 | Save Rule | 7-2 | High Low High Low |

If you made any mistakes while entering this rule, scan the **QUIT ENTERING RULES** bar code on page 7-3. If you already saved the rule, scan the **ERASE PREVIOUSLY SAVED RULE** bar code on page 7-2.

## Alternate Rule Sets

ADF rules may be grouped into one of four alternate sets which can be turned on and off when needed. This is useful when you want to format the same message in different ways. For example, a Code 128 bar code contains the following information:

**Class (2 digits), Stock Number (8) digits, Price (5 digits)**

This bar code might look like this:

**245671243701500**

where:

**Class = 24**

**Stock Number = 56712437**

**Price = 01500**

Ordinarily you would send this data as follows:

**24 (class key)**

**56712437 (stock key)**

**01500 (enter key)**

But, when there is a sale, you may want to send only the following:

**24 (class key)**

**56712437 (stock key)**

and the cashier will key the price manually.

To implement this, you would first enter an ADF rule that applies to the normal situation. This rule may look like this:

**When scanning a bar code of length 15, send the next 2 characters, send the class key, send the next 8 characters, send the stock key, send the data that remains, send the Enter key.**

The “sale” rule may look like this:

**When scanning a bar code of length 15, send the next 2 characters, send the class key, send the next 8 characters, send the stock key.**

To switch between the two sets of rules, a “switching rule” must be programmed. This rule specifies what type of bar code must be scanned to switch between the rule sets. For example,

in the case of the "sale" rule above, the rule programmer wants the cashier to scan the bar code "M" before a sale. To do this, a rule can be entered as follows:

**When scanning a bar code of length 1 that begins with "M", select rule set number 1**.

Another rule could be programmed to switch back.

**When scanning a bar code of length 1 that begins with "N", turn off rule set number 1.**

The switching back to normal rules can also be done in the "sale" rule. For example, the rule may look like this:

**When scanning a bar code of length 15, send the next 2 characters, send the class key, send the next 8 characters, send the stock key, turn off rule set 1.**

It is recommended that you scan the **Disable All Rule Sets** bar code after programming a rule belonging to an alternate rule set.

In addition to enabling and disabling rule sets within the rules, you can enable or disable them by scanning the appropriate bar codes on *Code Lengths* on page 7-7.

## Rules Hierarchy (in Bar Codes)

The order of programming individual rules is important.The most general rule should be programmed last.

All programmed rules are stored in a buffer. As they are programmed, they are stored at the "top" of a rules list. If three rules have been created, the list would be configured as follows:

Third Rule

Second Rule

First Rule

When data is scanned, the rules list is checked from top to bottom to determine if the criteria matches (and therefore, if the actions should occur). Input is modified into the data format specified by the first matching set of criteria it finds. Be sure that your most general rule is the last one programmed.

For example, if the THIRD rule states:

**When scanning a bar code of any length, send all data, then send the ENTER key.**

And the SECOND rule states:

**When scanning a Code 128 bar code of length 12, send the first four characters, then send the ENTER key, then send all remaining data.**

If a Code 128 bar code of length 12 were scanned, the THIRD rule would be in effect. The SECOND rule would appear to not function.

Note also that ADF rules are actually created when you use the standard data editing functions. Scan options are entered as ADF rules, and the hierarchy mentioned above also applies to them. For the LS 6800, this applies to prefix/suffix programming in the parameter *Scan Data Transmission Format*.

These rules reside in the same "rule list" as ADF Rules, so the order of their creation is also important.

## Default Rules

Every unit has a default rule to send all scan data. Units with custom software may have one or more default rules burned in. The rules hierarchy checks user programmable rules first, then the default rules. Default rules can be disabled by entering the following general rule in the user programmable buffer:

**When receiving scan data, send all data.**

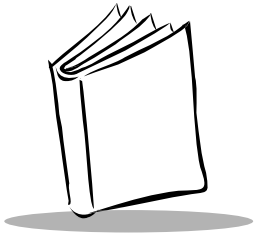
Since this rule always applies, ADF will never go into the default rules.

# Beeper Definitions

The following table defines beep sequences which occur during data entry.

**Table 6-1. Beeper Definitions**

| **Normal Data Entry. Duration of tones are short.** | |
|---|---|
| **Beeper Sequence** | **Indication** |
| **High-Low** | Entry of a number is expected. Enter another digit. Add leading zeros to the front if necessary. |
| **Low-Low** | Entry of an alphabetic character is expected. Enter another character or scan the End of Message bar code. |
| **High-High** | Entry of Criterion/Action is expected. Enter another criterion or action, or scan the Save Rule bar code. |
| **High-Low-High-Low** | Rule saved. Rule entry mode exited. |
| **High-Low-Low** | All criteria or actions were cleared for rule currently being entered; continue entry of rule. |
| **Low** | Last saved rule was successfully deleted. The rule presently being entered is left intact. |
| **Low-High-High** | All rules are now deleted. The rule presently being entered is left intact. (This beep sequence has a different meaning outside of ADF.) |
| | |
| **Error Indications. Duration of tones are very long.** | |
| **Beeper Sequence** | **Indication** |
| **Low-High-Low-High** | Out of rule memory. Erase some existing rules, then try to save rule again. (The current rule need not be re-entered.) |
| **Low-High-Low** | Cancel rule entry. Rule entry mode exited because of an error or the user asked to exit rule entry. |
| **Low-High** | Entry error, wrong bar code scanned. Re-enter criterion or action. All previously entered criteria and actions are retained. Criteria or action list is too long for a rule. |



# *Chapter 7*
# *ADF Bar Codes*

Due to the large number of Advanced Data Formatting bar codes, the page layout of this chapter is not as in Chapter 5. When scanning a bar code in this chapter, cover up unwanted bar codes in the scanning area. If you have difficulty programming ADF using this method, contact 1-800 - 92SYMBOL, extension 4417.

## Special Commands

### *Begin New Rule*

Scan this bar code to start entering a new rule.

**<FN3>7B1211**

**BEGIN NEW RULE**

## Save Rule

Scan this bar code to save the rule you entered.

**<FN3>4**

**SAVE RULE**

## Erase

Use these bar codes to erase criteria, actions, or rules.

**<FN3>6C-**

**ERASE CRITERIA AND START AGAIN**

**<FN3>6A-**

**ERASE ACTIONS AND START AGAIN**

**<FN3>81**

**ERASE PREVIOUSLY SAVED RULE**

**<FN3>80**

**ERASE ALL RULES**

## Quit Entering Rules

Scan this bar code to quit entering rules.

**<FN3>6Q**

**QUIT ENTERING RULES**

## Disable Rule Set

Use these bar codes to disable rule sets.

**<FN3>01**

**DISABLE RULE SET 1**

**<FN3>02**

**DISABLE RULE SET 2**

**<FN3>03**

**DISABLE RULE SET 3**

**<FN3>04**

**DISABLE RULE SET 4**

**<FN3>00**

**DISABLE ALL RULE SETS**

# Criteria

## *Code Types*

Scan the bar codes for all code types desired before selecting other criteria.

**<FN3>6C13D01**

**CODE 39**

**<FN3>6C13D02**

**CODABAR**

**<FN3>6C13D03**

**CODE 128**

**<FN3>6C13D06**

**I 2 OF 5**

**<FN3>6C13D0F**

**EAN 128**

## Code Types (Continued)

**<FN3>6C13D08**

**UPC-A**

**<FN3>6C13D09**

**UPC-E**

**<FN3>6C13D0A**

**EAN-8**

**<FN3>6C13D0B**

**EAN-13**

**<FN3>6C13D00**

**PDF417**

**<FN3>6C13D1A**

**ADF for Macro PDF417**
**(Applies rule to each block in MPDF set.)**

## Code Lengths

Scan these bar codes to define the number of characters the selected code types must contain. Select one length per rule only.

---

**Note:** *This is not a keypad.*

---

**<FN3>6C13701**

**1**

**<FN3>6C13702**

**2**

**<FN3>6C13703**

**3**

**<FN3>6C13704**

**4**

**<FN3>6C13705**

**5**

**<FN3>6C13706**

**6**

**<FN3>6C13707**

**7**

**<FN3>6C13708**

**8**

## *Code Lengths (Continued)*

**<FN3>6C13709**

**9**

**<FN3>6C1370A**

**10**

**<FN3>6C1370B**

**11**

**<FN3>6C1370C**

**12**

**<FN3>6C1370D**

**13**

**<FN3>6C1370E**

**14**

**<FN3>6C1370F**

**15**

**<FN3>6C13710**

**16**

## *Code Lengths (Continued)*

**<FN3>6C13711**

**17**

**<FN3>6C13712**

**18**

**<FN3>6C13713**

**19**

**<FN3>6C13714**

**20**

**<FN3>6C13715**

**21**

**<FN3>6C13716**

**22**

**<FN3>6C13717**

**23**

**<FN3>6C13718**

**24**

## *Code Lengths (Continued)*

**<FN3>6C13719**

**25**

**<FN3>6C1371A**

**26**

**<FN3>6C1371B**

**27**

**<FN3>6C1371C**

**28**

**<FN3>6C1371D**

**29**

**<FN3>6C1371E**

**30**

## Specific Data String

After scanning the following bar code:

1. Enter a string using the *Alphanumeric Keyboard* beginning on page 7-50.
2. Scan **END OF MESSAGE** on page 7-56.

**<FN3>6C201**

**SPECIFIC STRING AT START**

After scanning the following bar code:

1. Enter a location using the *Numeric Keypad* on page 7-12.
2. Enter a string using the *Alphanumeric Keyboard* beginning on page 7-50.
3. Scan **END OF MESSAGE** on page 7-56.

**<FN3>6C200**

**SPECIFIC STRING ANY LOCATION**

## Numeric Keypad

Bar codes on this page should not be confused with those on the alphanumeric keyboard.

**<FN3>A0**
**0**

**<FN3>A1**
**1**

**<FN3>A2**
**2**

**<FN3>A3**
**3**

**<FN3>A4**
**4**

**<FN3>A5**
**5**

**<FN3>A6**
**6**

**<FN3>A7**
**7**

**<FN3>A8**
**8**

**<FN3>A9**
**9**

**<FN3>A-**
**CANCEL**

## Rule Belongs To Set

Scan a bar code below to select which set a rule belongs to.

**<FN3>6C12A1**

**RULE BELONGS TO SET 1**

**<FN3>6C12A2**

**RULE BELONGS TO SET 2**

**<FN3>6C12A3**

**RULE BELONGS TO SET 3**

**<FN3>6C12A4**

**RULE BELONGS TO SET 4**

# Actions

## *Send Data*

Use these bar codes to send data.

**<FN3>6A5211**

**SEND DATA UP TO CHARACTER**

**<FN3>6A110**

**SEND ALL DATA THAT REMAINS**

**<FN3>6A141001**

**SEND NEXT CHARACTER**

**<FN3>6A141002**

**SEND NEXT 2 CHARACTERS**

**<FN3>6A141003**

**SEND NEXT 3 CHARACTERS**

**<FN3>6A141004**

**SEND NEXT 4 CHARACTERS**

**<FN3>6A141005**

**SEND NEXT 5 CHARACTERS**

**<FN3>6A141006**

**SEND NEXT 6 CHARACTERS**

**<FN3>6A141007**

**SEND NEXT 7 CHARACTERS**

## *Send Data (Continued)*

**<FN3>6A141008**

**SEND NEXT 8 CHARACTERS**

**<FN3>6A141009**

**SEND NEXT 9 CHARACTERS**

**<FN3>6A14100A**

**SEND NEXT 10 CHARACTERS**

**<FN3>6A14100B**

**SEND NEXT 11 CHARACTERS**

**<FN3>6A14100C**

**SEND NEXT 12 CHARACTERS**

**<FN3>6A14100D**

**SEND NEXT 13 CHARACTERS**

**<FN3>6A14100E**

**SEND NEXT 14 CHARACTERS**

**<FN3>6A14100F**

**SEND NEXT 15 CHARACTERS**

**<FN3>6A141010**

**SEND NEXT 16 CHARACTERS**

**<FN3>6A141011**

**SEND NEXT 17 CHARACTERS**

## *Send Data (Continued)*

**<FN3>6A141012**

**SEND NEXT 18 CHARACTERS**

**<FN3>6A141013**

**SEND NEXT 19 CHARACTERS**

**<FN3>6A141014**

**SEND NEXT 20 CHARACTERS**

**<FN3>6A141032**

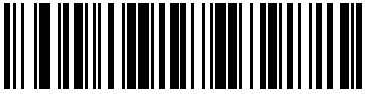
**SEND NEXT 50 CHARACTERS**

**<FN3>6A141064**

**SEND NEXT 100 CHARACTERS**

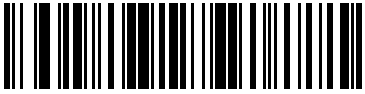
**<FN3>6A141096**

**SEND NEXT 150 CHARACTERS**



**<FN3>6A1410C8**

**SEND NEXT 200 CHARACTERS**



**<FN3>6A1410FA**

**SEND NEXT 250 CHARACTERS**

## Setup Fields

Scan a bar code below to move the cursor in relation to a specified character. Then enter a character by scanning a bar code from the *Alphanumeric Keyboard* beginning on page 7-50.

---

**Note:** *If there is no match when the rule is interpreted and the rule fails, the next rule is checked.*

---

**<FN3>6A5230**

**MOVE CURSOR TO CHARACTER**

**<FN3>6A1231**

**MOVE CURSOR TO START**

**<FN3>6A5235**

**MOVE CURSOR PAST CHARACTER**

## Skip Ahead

Use the following bar codes to skip ahead characters.

**<FN3>6A143301**

**SKIP AHEAD 1 CHARACTER**

**<FN3>6A143302**

**SKIP AHEAD 2 CHARACTERS**

**<FN3>6A143303**

**SKIP AHEAD 3 CHARACTERS**

**<FN3>6A143304**

**SKIP AHEAD 4 CHARACTERS**

**<FN3>6A143305**

**SKIP AHEAD 5 CHARACTERS**

**<FN3>6A143306**

**SKIP AHEAD 6 CHARACTERS**

**<FN3>6A143307**

**SKIP AHEAD 7 CHARACTERS**

**<FN3>6A143308**

**SKIP AHEAD 8 CHARACTERS**

## Skip Ahead (Continued)

**<FN3>6A143309**

**SKIP AHEAD 9 CHARACTERS**

**<FN3>6A14330A**

**SKIP AHEAD 10 CHARACTERS**

**<FN3>6A143332**

**SKIP AHEAD 50 CHARACTERS**

**<FN3>6A143364**

**SKIP AHEAD 100 CHARACTERS**

**<FN3>6A143396**

**SKIP AHEAD 150 CHARACTERS**

**<FN3>6A1433C8**

**SKIP AHEAD 200 CHARACTERS**

**<FN3>6A1433FA**

**SKIP AHEAD 250 CHARACTERS**

## Skip Back

Use the following bar codes to skip back characters.

**<FN3>6A143401**

**SKIP BACK 1 CHARACTERS**

**<FN3>6A143402**

**SKIP BACK 2 CHARACTERS**

**<FN3>6A143403**

**SKIP BACK 3 CHARACTERS**

**<FN3>6A143404**

**SKIP BACK 4 CHARACTERS**

**<FN3>6A143405**

**SKIP BACK 5 CHARACTERS**

**<FN3>6A143406**

**SKIP BACK 6 CHARACTERS**

**<FN3>6A143407**

**SKIP BACK 7 CHARACTER**

**<FN3>6A143408**

**SKIP BACK 8 CHARACTERS**

## Skip Back (Continued)

**<FN3>6A143409**

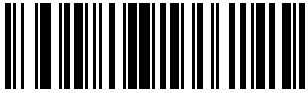
**SKIP BACK 9 CHARACTERS**

**<FN3>6A14340A**

**SKIP BACK 10 CHARACTERS**

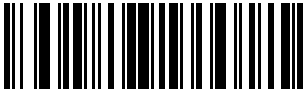
**<FN3>6A143432**

**SKIP BACK 50 CHARACTERS**



**<FN3>6A143464**

**SKIP BACK 100 CHARACTERS**



**<FN3>6A143496**

**SKIP BACK 150 CHARACTERS**

**<FN3>6A1434C8**

**SKIP BACK 200 CHARACTERS**

**<FN3>6A1434FA**

**SKIP BACK 250 CHARACTERS**

## Send Preset Value

Use these bar codes to send preset values.

**<FN3>6A1271**

**SEND VALUE 1**

**<FN3>6A1272**

**SEND VALUE 2**

## Modify Data

Use the bar codes below to modify data.

**<FN3>6A1260**

**REMOVE ALL SPACES**

**<FN3>6A1261**

**CRUNCH ALL SPACES**

**<FN3>6A1262**

**STOP SPACE REMOVAL**

**<FN3>6A1264**

**REMOVE LEADING ZEROS**

**<FN3>6A1265**

**STOP ZERO REMOVAL**

## Pad Data with Spaces

Use these bar codes to pad data with spaces.

**<FN3>6A146301**

**PAD SPACES TO LENGTH 1**

**<FN3>6A146302**

**PAD SPACES TO LENGTH 2**

**<FN3>6A146303**

**PAD SPACES TO LENGTH 3**

**<FN3>6A146304**

**PAD SPACES TO LENGTH 4**

**<FN3>6A146305**

**PAD SPACES TO LENGTH 5**

**<FN3>6A146306**

**PAD SPACES TO LENGTH 6**

**<FN3>6A146307**

**PAD SPACES TO LENGTH 7**

**<FN3>6A146308**

**PAD SPACES TO LENGTH 8**

## Pad Data with Spaces (Continued)

**<FN3>6A146309**

**PAD SPACES TO LENGTH 9**

**<FN3>6A14630A**

**PAD SPACES TO LENGTH 10**

**<FN3>6A14630B**

**PAD SPACES TO LENGTH 11**

**<FN3>6A14630C**

**PAD SPACES TO LENGTH 12**

**<FN3>6A14630D**

**PAD SPACES TO LENGTH 13**

**<FN3>6A14630E**

**PAD SPACES TO LENGTH 14**

**<FN3>6A14630F**

**PAD SPACES TO LENGTH 15**

**<FN3>6A146310**

**PAD SPACES TO LENGTH 16**

# Pad Data with Spaces (Continued)

**<FN3>6A146311**

**PAD SPACES TO LENGTH 17**

**<FN3>6A146312**

**PAD SPACES TO LENGTH 18**

**<FN3>6A146313**

**PAD SPACES TO LENGTH 19**

**<FN3>6A146314**

**PAD SPACES TO LENGTH 20**

**<FN3>6A146315**

**PAD SPACES TO LENGTH 21**

**<FN3>6A146316**

**PAD SPACES TO LENGTH 22**

**<FN3>6A146317**

**PAD SPACES TO LENGTH 23**

**<FN3>6A146318**

**PAD SPACES TO LENGTH 24**

## Pad Data with Spaces (Continued)

**<FN3>6A146319**

**PAD SPACES TO LENGTH 25**

**<FN3>6A14631A**

**PAD SPACES TO LENGTH 26**

**<FN3>6A14631B**

**PAD SPACES TO LENGTH 27**

**<FN3>6A14631C**

**PAD SPACES TO LENGTH 28**

**<FN3>6A14631D**

**PAD SPACES TO LENGTH 29**

**<FN3>6A14631E**

**PAD SPACES TO LENGTH 30**

**<FN3>6A146300**

**STOP PAD SPACES**

## Pad Data with Zeros

Use these bar codes to pad data with zeros.

**<FN3>6A146601**

**PAD ZEROS TO LENGTH 1**

**<FN3>6A146602**

**PAD ZEROS TO LENGTH 2**

**<FN3>6A146603**

**PAD ZEROS TO LENGTH 3**

**<FN3>6A146604**

**PAD ZEROS TO LENGTH 4**

**<FN3>6A146605**

**PAD ZEROS TO LENGTH 5**

**<FN3>6A146606**

**PAD ZEROS TO LENGTH 6**

**<FN3>6A146607**

**PAD ZEROS TO LENGTH 7**

**<FN3>6A146608**

**PAD ZEROS TO LENGTH 8**

## *Pad Data With Zeros (Continued)*

**<FN3>6A146609**

**PAD ZEROS TO LENGTH 9**

**<FN3>6A14660A**

**PAD ZEROS TO LENGTH 10**

**<FN3>6A14660B**

**PAD ZEROS TO LENGTH 11**

**<FN3>6A14660C**

**PAD ZEROS TO LENGTH 12**

**<FN3>6A14660D**

**PAD ZEROS TO LENGTH 13**

**<FN3>6A14660E**

**PAD ZEROS TO LENGTH 14**

**<FN3>6A14660F**

**PAD ZEROS TO LENGTH 15**

**<FN3>6A146610**

**PAD ZEROS TO LENGTH 16**

## *Pad Data With Zeros (Continued)*

**<FN3>6A146611**

**PAD ZEROS TO LENGTH 17**

**<FN3>6A146612**

**PAD ZEROS TO LENGTH 18**

**<FN3>6A146613**

**PAD ZEROS TO LENGTH 19**

**<FN3>6A146614**

**PAD ZEROS TO LENGTH 20**

**<FN3>6A146615**

**PAD ZEROS TO LENGTH 21**

**<FN3>6A146616**

**PAD ZEROS TO LENGTH 22**

**<FN3>6A146617**

**PAD ZEROS TO LENGTH 23**

**<FN3>6A146618**

**PAD ZEROS TO LENGTH 24**

## Pad Data With Zeros (Continued)

**<FN3>6A146619**

**PAD ZEROS TO LENGTH 25**

**<FN3>6A14661A**

**PAD ZEROS TO LENGTH 26**

**<FN3>6A14661B**

**PAD ZEROS TO LENGTH 27**

**<FN3>6A14661C**

**PAD ZEROS TO LENGTH 28**

**<FN3>6A14661D**

**PAD ZEROS TO LENGTH 29**

**<FN3>6A14661E**

**PAD ZEROS TO LENGTH 30**

**<FN3>6A146600**

**STOP PAD ZEROS**

## Beeps

Select one beep sequence per ADF rule.

**<FN3>6A13A01**

**BEEP ONCE**

**<FN3>6A13A02**

**BEEP TWICE**

**<FN3>6A13A03**

**BEEP THREE TIMES**

## Control Characters

Scan these bar codes to send control characters.

**<FN3>6A144100**

**SEND CONTROL 2**

**<FN3>6A144101**

**SEND CONTROL A**

**<FN3>6A144102**

**SEND CONTROL B**

**<FN3>6A144103**

**SEND CONTROL C**

**<FN3>6A144104**

**SEND CONTROL D**

**<FN3>6A144105**

**SEND CONTROL E**

**<FN3>6A144106**

**SEND CONTROL F**

**<FN3>6A144107**

**SEND CONTROL G**

## Control Characters (Continued)

**<FN3>6A144108**

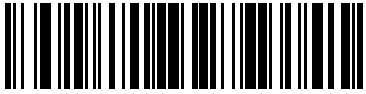
**SEND CONTROL H**

**<FN3>6A144109**

**SEND CONTROL I**

**<FN3>6A14410A**

**SEND CONTROL J**



**<FN3>6A14410B**

**SEND CONTROL K**

**<FN3>6A14410C**

**SEND CONTROL L**

**<FN3>6A14410D**

**SEND CONTROL M**

**<FN3>6A14410E**

**SEND CONTROL N**

**<FN3>6A14410F**

**SEND CONTROL O**

## *Control Characters (Continued)*

**<FN3>6A144110**

**SEND CONTROL P**

**<FN3>6A144111**

**SEND CONTROL Q**

**<FN3>6A144112**

**SEND CONTROL R**

**<FN3>6A144113**

**SEND CONTROL S**

**<FN3>6A144114**

**SEND CONTROL T**

**<FN3>6A144115**

**SEND CONTROL U**

**<FN3>6A144116**

**SEND CONTROL V**

**<FN3>6A144117**

**SEND CONTROL W**

## Control Characters (Continued)

**<FN3>6A144118**

**SEND CONTROL X**

**<FN3>6A144119**

**SEND CONTROLY**

**<FN3>6A14411A**

**SEND CONTROL Z**

**<FN3>6A14411B**

**SEND CONTROL [**

**<FN3>6A14411C**

**SEND CONTROL **

**<FN3>6A14411D**

**SEND CONTROL ]**

**<FN3>6A14411E**

**SEND CONTROL 6**

**<FN3>6A14411F**

**SEND CONTROL -**

## Keyboard Characters

Use these bar codes to send keyboard characters.

**<FN3>6A144120**

**SEND SPACE**

**<FN3>6A144121**

**SEND !**

**<FN3>6A144122**

**SEND “**

**<FN3>6A144123**

**SEND #**

**<FN3>6A144124**

**SEND $**

**<FN3>6A144125**

**SEND %**

**<FN3>6A144126**

**SEND &**

**<FN3>6A144127**

**SEND ‘**

## Keyboard Characters (Continued)

**<FN3>6A144128**

**SEND (**

**<FN3>6A144129**

**SEND )**

**<FN3>6A14412A**

**SEND ***

**<FN3>6A14412B**

**SEND +**

**<FN3>6A14412C**

**SEND ,**

**<FN3>6A14412D**

**SEND -**

**<FN3>6A14412E**

**SEND .**

**<FN3>6A14412F**

**SEND /**

## *Keyboard Characters (Continued)*

**<FN3>6A144130**

**SEND 0**

**<FN3>6A144131**

**SEND 1**

**<FN3>6A144132**

**SEND 2**

**<FN3>6A144133**

**SEND 3**

**<FN3>6A144134**

**SEND 4**

**<FN3>6A144135**

**SEND 5**

**<FN3>6A144136**

**SEND 6**

**<FN3>6A144137**

**SEND 7**

## Keyboard Characters (Continued)

**<FN3>6A144138**

**SEND 8**

**<FN3>6A144139**

**SEND 9**

**<FN3>6A14413A**

**SEND :**

**<FN3>6A14413B**

**SEND ;**

**<FN3>6A14413C**

**SEND <**

**<FN3>6A14413D**

**SEND =**

**<FN3>6A14413E**

**SEND >**

**<FN3>6A14413F**

**SEND ?**

## *Keyboard Characters (Continued)*

**<FN3>6A144140**

**SEND @**

**<FN3>6A144141**

**SEND A**

**<FN3>6A144142**

**SEND B**

**<FN3>6A144143**

**SEND C**

**<FN3>6A144144**

**SEND D**

**<FN3>6A144145**

**SEND E**

**<FN3>6A144146**

**SEND F**

**<FN3>6A144147**

**SEND G**

## Keyboard Characters (Continued)

**<FN3>6A144148**

**SEND H**

**<FN3>6A144149**

**SEND I**

**<FN3>6A14414A**

**SEND J**

**<FN3>6A14414B**

**SEND K**

**<FN3>6A14414C**

**SEND L**

**<FN3>6A14414D**

**SEND M**

**<FN3>6A14414E**

**SEND N**

**<FN3>6A14414F**

**SEND O**

## *Keyboard Characters (Continued)*

**<FN3>6A144150**

**SEND P**

**<FN3>6A144151**

**SEND Q**

**<FN3>6A144152**

**SEND R**

**<FN3>6A144153**

**SEND S**

**<FN3>6A144154**

**SEND T**

**<FN3>6A144155**

**SEND U**

**<FN3>6A144156**

**SEND V**

**<FN3>6A144157**

**SEND W**

## Keyboard Characters (Continued)

**<FN3>6A144158**

**SEND X**

**<FN3>6A144159**

**SEND Y**

**<FN3>6A14415A**

**SEND Z**

**<FN3>6A14415B**

**SEND [**

**<FN3>6A14415C**

**SEND **

**<FN3>6A14415D**

**SEND ]**

**<FN3>6A14415E**

**SEND ^**

**<FN3>6A14415F**

**SEND _**

## *Keyboard Characters (Continued)*

**<FN3>6A144160**

**SEND ‘**

**<FN3>6A144161**

**SEND a**

**<FN3>6A144162**

**SEND b**

**<FN3>6A144163**

**SEND c**

**<FN3>6A144164**

**SEND d**

**<FN3>6A144165**

**SEND e**

**<FN3>6A144166**

**SEND f**

**<FN3>6A144167**

**SEND g**

## Keyboard Characters (Continued)

**<FN3>6A144168**

**SEND h**

**<FN3>6A144169**

**SEND i**

**<FN3>6A14416A**

**SEND j**

**<FN3>6A14416B**

**SEND k**

**<FN3>6A14416C**

**SEND l**

**<FN3>6A14416D**

**SEND m**

**<FN3>6A14416E**

**SEND n**

**<FN3>6A14416F**

**SEND o**

## *Keyboard Characters (Continued)*

**<FN3>6A144170**

**SEND p**

**<FN3>6A144171**

**SEND q**

**<FN3>6A144172**

**SEND r**

**<FN3>6A144173**

**SEND s**

**<FN3>6A144174**

**SEND t**

**<FN3>6A144175**

**SEND u**

**<FN3>6A144176**

**SEND v**

**<FN3>6A144177**

**SEND w**

## Keyboard Characters (Continued)

**<FN3>6A144178**

**SEND x**

**<FN3>6A144179**

**SEND y**

**<FN3>6A14417A**

**SEND z**

**<FN3>6A14417B**

**SEND {**

**<FN3>6A14417C**

**SEND |**

**<FN3>6A14417D**

**SEND }**

**<FN3>6A14417E**

**SEND ~**

## *Turn On/Off Rule Set*

Use these bar codes to turn rule sets on and off.

**<FN3>6A13911**

**TURN ON RULE SET 1**

**<FN3>6A13921**

**TURN ON RULE SET 2**

**<FN3>6A13931**

**TURN ON RULE SET 3**

**<FN3>6A13941**

**TURN ON RULE SET 4**

**<FN3>6A13910**

**TURN OFF RULE SET 1**

**<FN3>6A13920**

**TURN OFF RULE SET 2**

**<FN3>6A13930**

**TURN OFF RULE SET 3**

**<FN3>6A13940**

**TURN OFF RULE SET 4**

# Alphanumeric Keyboard

**<FN3>B20**

**SPACE**

**<FN3>B23**

**#**

**<FN3>B24**

**$**

**<FN3>B25**

**%**

**<FN3>B2A**

*****

**<FN3>B2B**

**+**

**<FN3>B2D**

**-**

**<FN3>B2E**

**.**

**<FN3>B2F**

**/**

**<FN3>B21**

**!**

# Alphanumeric Keyboard (Continued)

**<FN3>B22**

**“**

**<FN3>B26**

**&**

**<FN3>B27**

**‘**

**<FN3>B28**

**(**

**<FN3>B29**

**)**

**<FN3>B3A**

**:**

**<FN3>B3B**

**;**

**<FN3>B3C**

**<**

**<FN3>B3D**

**=**

**<FN3>B3E**

**>**

# Alphanumeric Keyboard (Continued)

**<FN3>B3F**

**?**

**<FN3>B40**

**@**

**<FN3>B5B**

**[**

**<FN3>B5C**

****

**<FN3>B5D**

**]**

**<FN3>B5E**

**^**

**<FN3>B5F**

**_**

**<FN3>B60**

**`**

# Alphanumeric Keyboard (Continued)

Bar codes on this page should not be confused with those on the numeric keypad.

**<FN3>B30**

**0**

**<FN3>B31**

**1**

**<FN3>B32**

**2**

**<FN3>B33**

**3**

**<FN3>B34**

**4**

**<FN3>B35**

**5**

**<FN3>B36**

**6**

**<FN3>B37**

**7**

**<FN3>B38**

**8**

**<FN3>B39**

**9**

# Alphanumeric Keyboard (Continued)

**<FN3>B41**

**A**

**<FN3>B42**

**B**

**<FN3>B43**

**C**

**<FN3>B44**

**D**

**<FN3>B45**

**E**

**<FN3>B46**

**F**

**<FN3>B47**

**G**

**<FN3>B48**

**H**

**<FN3>B49**

**I**

**<FN3>B4A**

**J**

# Alphanumeric Keyboard (Continued)

**<FN3>B4B**

**K**

**<FN3>B4C**

**L**

**<FN3>B4D**

**M**

**<FN3>B4E**

**N**

**<FN3>B4F**

**O**

**<FN3>B50**

**P**

**<FN3>B51**

**Q**

**<FN3>B52**

**R**

**<FN3>B53**

**S**

**<FN3>B54**

**T**

# Alphanumeric Keyboard (Continued)

**<FN3>B55**

**U**

**<FN3>B56**

**V**

**<FN3>B57**

**W**

**<FN3>B58**

**X**

**<FN3>B59**

**Y**

**<FN3>B5A**

**Z**

**<FN3>B-**

**CANCEL**

**<FN3>B+**

**END OF MESSAGE**

# Alphanumeric Keyboard (Continued)

**<FN3>B61**

**a**

**<FN3>B62**

**b**

**<FN3>B63**

**c**

**<FN3>B64**

**d**

**<FN3>B65**

**e**

**<FN3>B66**

**f**

**<FN3>B67**

**g**

**<FN3>B68**

**h**

**<FN3>B69**

**i**

**<FN3>B6A**

**j**

# Alphanumeric Keyboard (Continued)

**<FN3>B6B**

**k**

**<FN3>B6C**

**l**

**<FN3>B6D**

**m**

**<FN3>B6E**

**n**

**<FN3>B6F**

**o**

**<FN3>B70**

**p**

**<FN3>B71**

**q**

**<FN3>B72**

**r**

**<FN3>B73**

**s**

**<FN3>B74**

**t**

# Alphanumeric Keyboard (Continued)

**<FN3>B75**

**u**

**<FN3>B76**

**v**

**<FN3>B77**

**w**

**<FN3>B78**

**x**

**<FN3>B79**

**y**

**<FN3>B7A**

**z**

**<FN3>B7B**

**{**

**<FN3>B7C**

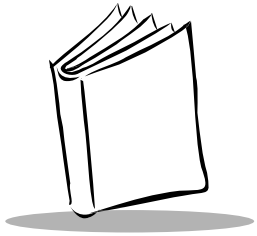
**|**

**<FN3>B7D**

**}**

**<FN3>B7E**

**~**



# *Appendix A*
# *Programming Reference*

## AIM Code Identifiers

Each AIM Code Identifier contains the three-character string **]cm** where:

] = Flag Character (ASCII 93)
c = Code Character
m = Modifier Character

**Table A-1. AIM Code Identifiers**

| Code Character | Code Type |
|---|---|
| A | Code 39 |
| C | Code 128 |
| E | UPC/EAN |
| F | Codabar |
| G | Code 93 |
| I | Interleaved 2 of 5 |
| L | PDF417, MicroPDF417 |
| M | MSI Plessey |
| S | D 2 of 5, IATA 2 of 5 |
| X | Code 39 Trioptic, Bookland EAN |

The modifier character is the sum of the applicable option values based on Table A-2.

## Table A-2. Modifier Characters

| Code Type | Option Value | Option |
|---|---|---|
| **Code 39** | 0 | No Check character or Full ASCII processing. |
| | 1 | Reader has checked one check character. |
| | 3 | Reader has checked and stripped check character. |
| | 4 | Reader has performed Full ASCII character conversion. |
| | 5 | Reader has performed Full ASCII character conversion and checked one check character. |
| | 7 | Reader has performed Full ASCII character conversion and checked and stripped check character. |
| | | Example: A Full ASCII bar code with check character W, **A+I+MI+DW**, can be transmitted as **]A7**AimId where 7 = (1+2+4). |
| **Trioptic Code 39** | 0 | No option specified at this time. Always transmit 0. |
| | | Example: A Trioptic bar code 412356 is transmitted as **]X0**412356. |
| **Code 93** | 0 | No option specified at this time. Always transmit 0. |
| | | Example: A Code 39 bar code 012345678905 is transmitted as **]G0**012345678905. |
| **Code 128** | 0 | Standard data packet, No Function code 1 in first symbol position. |
| | 1 | Function code 1 in first symbol character position. |
| | 2 | Function code 1 in second symbol character position. |
| | | Example: A Code (EAN) 128 bar code with Function 1 character in the first position, **Fcnt1**AimId is transmitted as **]C1**AimId. |
| **Codabar** | 0 | No check digit processing. |
| | 1 | Reader has checked check digit. |
| | 3 | Reader has stripped check digit before transmission. |
| | | Example: A Codabar bar code without check digit, 4123, is transmitted as **F0**4123. |

**Table A-2. Modifier Characters (Continued)**

| Code Type | Option Value | Option |
|---|---|---|
| **EAN/UPC** | 0 | Standard packet in full EAN country code format, which is 13 digits for UPC-A and UPC-E (not including supplemental data). |
| | 1 | Two digit supplement data only. |
| | 2 | Five digit supplement data only. |
| | 4 | EAN-8 data packet. |
| | | Example:A UPC-A bar code 012345678905 is transmitted as **]E0**0012345678905. |
| **Interleaved 2 of 5** | 0 | No check digit processing. |
| | 1 | Reader has validated check digit. |
| | 3 | Reader has validated and stripped check digit. |
| | | Example: An I 2 of 5 bar code without check digit, 4123, will be transmitted as **]I0**4123. |
| **Discrete 2 of 5** | 0 | No option specified at this time. Always transmit 0. |
| | | Example: A D 2 of 5 bar code 4123 is transmitted as **]S0**4123. |
| **MSI Plessey** | 0 | Single check digit checked. |
| | 1 | Two check digits checked. |
| | 2 | Single check digit verified and stripped before transmission. |
| | 3 | Two check digits verified and stripped before transmission. |
| | | Example: An MSI Plessey bar code 4123, with a single check digit checked, is transmitted as **]M0**4123. |
| **Bookland EAN** | 0 | No option specified at this time. Always transmit 0. |
| | | Example: A Bookland EAN bar code 123456789X is transmitted as **]X0**123456789X. |

**Table A-2. Modifier Characters (Continued)**

| Code Type | Option Value | Option |
|---|---|---|
| **PDF417, MicroPDF417** | 0 | Reader set to conform to protocol defined in 1994 PDF417 symbology specifications. **Note:** When this option is transmitted, the receiver cannot reliably determine whether ECIs have been invoked or whether data byte $92_{DEC}$ has been doubled in transmission. |
| | 1 | Reader set to follow the ECI protocol (Extended Channel Interpretation). All data characters $92_{DEC}$ are doubled. |
| | 2 | Reader set for Basic Channel operation (no escape character transmission protocol). Data characters $92_{DEC}$ are not doubled. **Note:** When decoders are set to this mode, unbuffered Macro symbols and symbols requiring the decoder to convey ECI escape sequences cannot be transmitted. |
| | 3 | The bar code contains a UCC/EAN-128 symbol, and the first codeword is 903-907, 912, 914, 915. |
| | 4 | The bar code contains a UCC/EAN-128 symbol, and the first codeword is in the range 908-909. |
| | 5 | The bar code contains a UCC/EAN-128 symbol, and the first codeword is in the range 910-911. |
| | | Example: A PDF417 bar code ABCD, with no transmission protocol enabled, is transmitted as ]L2ABCD |

According to AIM standards, a UPC with supplemental bar code is transmitted in one of the following formats:

**]E0** (UPC chars) (terminator) **]E2** (supplemental) (terminator)

In the LS 6800, however, the format is changed to:

**]E0** (UPC chars) **]E2** (supplemental)

Therefore, a UPC with two supplemental characters, 01234567890510, is transmitted to the host as a 21-character string, **]E0**0012345678905**]E1**10.

# Prefix / Suffix Values

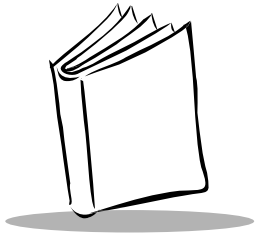
The following values can be assigned as prefixes or suffixes for ASCII character data transmission. If you're using a keyboard interface, refer to the Synapse "Smart Cable" Interface Guide for keystroke prefix/suffix values.

**Table A-3. Prefix/Suffix Values**

| Prefix/Suffix Value | Full ASCII Code 39 Encode Char. | ASCII Character | Prefix/Suffix Value | Full ASCII Code 39 Encode Char. | ASCII Character |
|---|---|---|---|---|---|
| 1000 | %U | NUL | 1030 | %D | RS |
| 1001 | $A | SOH | 1031 | %E | US |
| 1002 | $B | STX | 1032 | Space | Space |
| 1003 | $C | ETX | 1033 | /A | ! |
| 1004 | $D | EOT | 1034 | /B | " |
| 1005 | $E | ENQ | 1035 | /C | # |
| 1006 | $F | ACK | 1036 | /D | $ |
| 1007 | $G | BELL | 1037 | /E | % |
| 1008 | $H | BCKSPC | 1038 | /F | & |
| 1009 | $I | HORIZ TAB | 1039 | /G | ‘ |
| 1010 | $J | LF/NW LN | 1040 | /H | ( |
| 1011 | $K | VT | 1041 | /I | ) |
| 1012 | $L | FF | 1042 | /J | * |
| 1013 | $M | CR/ENTER | 1043 | /K | + |
| 1014 | $N | SO | 1044 | /L | , |
| 1015 | $O | SI | 1045 | - | - |
| 1016 | $P | DLE | 1046 | . | . |
| 1017 | $Q | DC1 | 1047 | / | / |
| 1018 | $R | DC2 | 1048 | 0 | 0 |
| 1019 | $S | DC3 | 1049 | 1 | 1 |
| 1020 | $T | DC4 | 1050 | 2 | 2 |
| 1021 | $U | NAK | 1051 | 3 | 3 |
| 1022 | $V | SYN | 1052 | 4 | 4 |
| 1023 | $W | ETB | 1053 | 5 | 5 |
| 1024 | $X | CAN | 1054 | 6 | 6 |
| 1025 | $Y | EM | 1057 | 7 | 7 |
| 1026 | $Z | SUB | 1056 | 8 | 8 |
| 1027 | %A | ESC | 1057 | 9 | 9 |
| 1028 | %B | FS | 1058 | /Z | : |
| 1029 | %C | GS | 1059 | %F | ; |

## Table A-3. Prefix/Suffix Values (Cont'd)

| Prefix/Suffix Value | Full ASCII Code 39 Encode Char. | ASCII Character | Prefix/Suffix Value | Full ASCII Code 39 Encode Char. | ASCII Character |
|---|---|---|---|---|---|
| 1060 | %G | < | 1095 | %O | _ |
| 1061 | %H | = | 1096 | %W | ` |
| 1062 | %I | > | 1097 | +A | a |
| 1063 | %J | ? | 1098 | +B | b |
| 1064 | %V | @ | 1099 | +C | c |
| 1065 | A | A | 1100 | +D | d |
| 1066 | B | B | 1101 | +E | e |
| 1067 | C | C | 1102 | +F | f |
| 1068 | D | D | 1103 | +G | g |
| 1069 | E | E | 1104 | +H | h |
| 1070 | F | F | 1105 | +I | i |
| 1071 | G | G | 1106 | +J | j |
| 1072 | H | H | 1107 | +K | k |
| 1073 | I | I | 1108 | +L | l |
| 1074 | J | J | 1109 | +M | m |
| 1075 | K | K | 1110 | +N | n |
| 1076 | L | L | 1111 | +O | o |
| 1077 | M | M | 1112 | +P | p |
| 1078 | N | N | 1113 | +Q | q |
| 1079 | O | O | 1114 | +R | r |
| 1080 | P | P | 1115 | +S | s |
| 1081 | Q | Q | 1116 | +T | t |
| 1082 | R | R | 1117 | +U | u |
| 1083 | S | S | 1118 | +V | v |
| 1084 | T | T | 1119 | +W | w |
| 1085 | U | U | 1120 | +X | x |
| 1086 | V | V | 1121 | +Y | y |
| 1087 | W | W | 1122 | +Z | z |
| 1088 | X | X | 1123 | %P | { |
| 1089 | Y | Y | 1124 | %Q | \| |
| 1090 | Z | Z | 1125 | %R | } |
| 1091 | %K | [ | 1126 | %S | ~ |
| 1092 | %L | \ | 1127 | | Undefined |
| 1093 | %M | ] | | | |
| 1094 | %N | ^ | 7013 | | ENTER |



# *Glossary*

| | |
|---|---|
| **Aperture** | The opening in an optical system defined by a lens or baffle that establishes the field of view. |
| **ASCII** | American Standard Code for Information Interchange. A 7 bit-plus-parity code representing 128 letters, numerals, punctuation marks, and control characters. It is a standard data transmission code in the U.S. |
| **Asymmetric Width Growth** | Non-uniform growth of elements in a printed symbol. |
| **Auto-discrimination** | The ability of an interface controller to determine the code type of a scanned bar code. After this determination is made, the information content can be decoded. |
| **Average Bar Width Growth** | Average deviation of bars from nominal widths over the entire symbol. |
| **Bad Check Digit** | Error message resulting from failure of the check digit to calculate properly. |
| **Bad Data Character** | Error message caused by failure of one or more data characters to decode properly. |
| **Bad Print Contrast** | Error message due to lack of contrast between the background and the bars of the symbol. |
| **Bar** | The dark element in a printed bar code symbol. |
| **Bar Code Density** | The number of characters represented per unit of measurement (e.g., characters per inch in one-dimensional symbologies, characters per square inch in PDF417). |

| | |
|---|---|
| **Bar Height** | The dimension of a bar measured perpendicular to the bar width. |
| **Bar Width** | Thickness of a bar measured from the edge closest to the symbol start character to the trailing edge of the same bar. |
| **Bar Width Deviation** | Increase or decrease in bar width as compared with nominal bar width. |
| **Baud Rate** | A measure of the data flow or number of signaling events occurring per second. When one bit is the standard "event," this is a measure of bits per second (bps). For example, a baud rate of 50 means transmission of 50 bits of data per second. |
| **Bidirectional Reading Capability** | The ability to decode a symbol successfully by reading in complementary (opposite) directions across bars and spaces. |
| **Bit** | Binary digit. One bit is the basic unit of binary information. Generally, eight consecutive bits compose one byte of data. The pattern of 0 and 1 values within the byte determines its "meaning." |
| **Buffer** | An area of memory allocated for data storage. In this context, a buffer's data storage capacity is needed when data can flow into the device more quickly than the device can process that data. Buffering the data preserves it until it can be processed. |
| **Byte** | On an addressable boundary, eight adjacent binary digits (0 and 1) combined in a pattern to represent a specific character or numeric value. Bits are numbered from the right, 0 through 7, with bit 0 the low-order bit. One byte in memory can be used to store one ASCII character. |
| **Character** | A pattern of bars and spaces which either directly represents data or indicates a control function, such as a number, letter, punctuation mark, or communications control contained in a message. |
| **Character Set** | Those characters available for encodation in a particular bar code symbology. |
| **Check Digit** | A digit used to verify a correct symbol decode. The scanner inserts the decoded data into an arithmetic formula and checks that the resulting number matches the encoded check digit. Check digits are required for UPC and Code 128 but are optional for other symbologies. Using check digits decreases the chance of substitution errors when a symbol is decoded. |
| **Cluster** | One of three subsets of mututally exclusive codeword definitions within PDF417. |

| | |
|---|---|
| **Codabar** | A discrete self-checking code with a character set consisting of start/stop characters (A B C D or * T N E), digits 0 to 9, and these additional characters: ( - $ : / , +). |
| **Code** | Set of unambiguous rules specifying the way in which data may be represented. |
| **Codeword** | In PDF417, a single group of bars and spaces (4 bars and 4 spaces, for a total of 17 module widths) which represents one or more numbers, letters, or other symbols. |
| **Codeword Pd (Codeword Percent Decode)** | Within a PDF417 symbol, the percentage of codewords which decoded successfully; the number of good codewords divided by the total number of codewords (data codewords plus error correction codewords). |
| **Code Length** | Number of data characters in a bar code between the start and stop characters, not including those characters. |
| **Code 128** | A high density symbology which allows the interface controller to encode all 128 ASCII characters without adding extra symbol elements. |
| **Code 3 Of 9 (Code 39)** | A versatile and widely used alphanumeric bar code symbology with a set of 43 character types, including all uppercase letters, numerals from 0 to 9, and 7 special characters (- . / + % $ and space). The code name is derived from the fact that 3 of 9 elements representing a character are wide, while the remaining 6 are narrow. |
| **Continuous Code** | A bar code or symbol in which all spaces within the symbol are parts of characters. There are no intercharacter gaps in a continuous code. The absence of gaps allows for greater information density. |
| **Country Flag** | In EAN-8 and EAN-13 codes, two or three digits which appear immediately following the left guard bar pattern. |
| **Dead Zone** | An area within a scanner's field of view, in which specular reflection may prevent a successful decode. |
| **Decode** | To recognize a bar code symbology (e.g., UPC/EAN) and then analyze the content of the specific bar code scanned. To translate the bar/space pattern into defined characters within a defined symbology. |
| **Decode Algorithm** | A decoding scheme that converts pulse widths into data representation of the letters or numbers encoded within a bar code symbol. |

**Depth Of Field** The range between minimum and maximum distances at which a scanner can read a symbol with a certain minimum element width.

**Discrete Code** Abar code or symbol in which the spaces between characters (intercharacter gaps) are not part of the code (e.g., Code 39).

**Discrete 2 Of 5** A binary bar code symbology representing each character by a group of five bars, two of which are wide. The location of wide bars in the group determines which character is encoded; spaces are insignificant. Only numeric characters (0 to 9) and START/STOP characters may be encoded.

**EAN** European Article Number. This European/International version of the UPC provides its own coding format and symbology standards. Element dimensions are specified metrically. EAN is used primarily in retail. Main variants are EAN-8 and EAN-13.

**Edge Roughness** Edge irregularities as compared with a nominal bar edge.

**Element** Generic term for a bar or space.

**Encoded Area** Total linear dimension occupied by all characters of a code pattern, including start/stop characters and data.

**Error Correction** In addition to error detection, the recovery capability of PDF417 over missing, destroyed, or misdecoded codewords. Error correction capability is based on the level of security (0 - 8) selected when the PDF417 label is printed.

**Extraneous Ink** Ink in a scan area not intended to be there (i.e., tracking and splatter).

**First Read Rate** Percentage of correct readings obtainable by one pass of a scanning device over a bar code.

**Flash** Derived from EEPROM, this is a type of memory that holds its content without power but must be erased in bulk — or in a "flash." Typically, these memory chips are less expensive and provide higher bit densities.

**Guard Bars** The start, stop, and center delimiting bars of UPC and EAN symbols.

**Host Computer** A computer that serves other terminals in a network, providing such services as computation, database access, supervisory programs, and network control.

**Intercharacter Gap** The space between two adjacent bar code characters in a discrete code.

**Interleaved Bar Code** — A bar code in which characters are paired together, using bars to represents the first character and the intervening spaces to represent the second.

**Interleaved 2 Of 5** — A binary bar code symbology representing character pairs in groups of five bars and five interleaved spaces. Interleaving provides for greater information density. The location of wide elements (bar/spaces) within each group determines which characters are encoded. This continuous code type uses no intercharacter spaces. Only numeric (0 to 9) and START/STOP characters may be encoded.

**Laser** — An acronym for Light Amplification by Stimulated Emission of Radiation. The laser is an intense light source. Light from a laser is all the same frequency, unlike the output of an incandescent bulb. Laser light is typically coherent and has a high energy density.

**Laser Spot Size** — The diameter of the spot of laser light scanning the bar code, as measured at a given distance from the bar code. Smaller spot sizes yield higher resolution but poorer depth of focus.

**LED Indicator** — A semiconductor diode (LED - Light Emitting Diode) used as an indicator, often in digital displays. The semiconductor uses applied voltage to produce light of a certain frequency determined by the semiconductor's particular chemical composition.

**Mil** — 1 mil = 1 thousandth of an inch.

**Minimum Reflectance Difference (Mrd)** — The difference in percentage between light reflected from spaces ($\mathbf{R_S}$) and light reflected from bars ($\mathbf{R_B}$). $\mathbf{MRD = \%R_S - \%R_B}$.

**Misread (Misdecode)** — A condition which occurs when the data output of a reader or interface controller does not agree with the data encoded within a bar code symbol.

**Module** — The narrowest bar or space (unit of measure) in a code. The term is used by the Uniform Code Council in its description of UPC/EAN code; it is also used in the description of Code 128. Contiguous modules are used to form bars or spaces which are wider than one unit.

**Module Aspect Ratio** — The ratio of height to width of the narrowest bar or space, or unit of measure, in a bar code.

**Nanometre** — A unit of measure used to define the wavelength of light. Equal to $10^{-9}$ metre.

**Nominal** The exact (or ideal) intended value for a specified parameter. Tolerances are specified as positive and negative deviations from this value.

**Nominal Size** Standard size for a bar code symbol. Most UPC/EAN codes can be used over a range of magnifications (e.g., from 0.80 to 2.00 of nominal).

**Number System Character** In the UPC/EAN code used in a retail application, the mandatory, first encoded character, after the left guard bars. The corresponding human readable character identifies the coded character and appears at the bottom left-hand margin of the symbol. The assigned system number corresponds to a usage category for the bar coded item.

**One-dimensional Symbology** Symbologies which encode data only in a linear or horizontal dimension (X-dimension); the symbol's vertical height (Y-dimension) is redundant (e.g., UPC/EAN, Code 39).

**Opacity** The capacity for material to interfere with transmission of light.

**Overhead** The number of characters required for start, stop, and checking for a given symbol (in PDF417, also left and right row indicators and error correction codewords). For example, a one-dimensional symbol requiring start/stop and two check characters contains four characters of overhead. Thus, to encode three data characters, seven characters are required.

**Parameter** A variable that can have different values assigned to it.

**Parity Type** A parity check bit is the most significant bit of each ASCII coded character. The parity should be set to help detect transmission errors. The parity should be set to match that of the receiving device. If even parity is selected, the parity bit has a value (0 or 1) to ensure that an even number of 1 bits are contained in the coded character. If odd parity is selected, the parity bit will have a value (0 or 1) to ensure that an odd number of 1 bits are contained in the coded character. If 0 parity is selected, the parity bit always will be set to 0. If 1 parity is selected, the parity bit always will be set to 1.

**PDF417** A two-dimensional, or stacked, bar code symbology which can encode over one kilobyte of data per label and which represents data in the form of codewords (values 0 - 928). Each codeword consists of four bars and four spaces, for a total of 17 module widths; modules vary in width from one to six element widths. The symbology permits encoding up to 30 data columns and from 3 to 90 data rows. For ease of reading while still maintaining high data density, codewords are encoded in three mutually-exclusive encodation sets, or clusters, with the same cluster repeating sequentially each third row.

**Percent Decode** The average probability that a single scan of a bar code would result in a successful decode. In a well-designed bar code scanning system, that probability should approach near 100%.

**Print Contrast Signal (PCS)** Measurement of the contrast (brightness difference) between the bars and spaces of a symbol. A minimum PCS value is needed for a bar code symbol to be scannable. PCS = (RL - RD) / RL, where RL is the reflectance factor of the background and RD the reflectance factor of the dark bars.

**Prom** Acronym for Programmable Read Only Memory. An integrated circuit which can be programmed through special processes and accessed at random during normal operation. Reprogramming is possible, but only through processes such as ultraviolet light erasing and electrical rewriting of data.

**Protocol** For a specific signaling type, a set of recognized rules governing the format and timing of message exchange. Between data communications devices, this includes an exchange of predetermined signals arranged for both establishing connection and for disconnecting.

**Quiet Zone** A clear space, containing no dark marks, which precedes the start character of a bar code symbol and follows the stop character.

**Reflectance** Amount of light returned from an illuminated surface.

**Resolution** The narrowest element dimension which can be distinguished by a particular reading device or printed with a particular device or method.

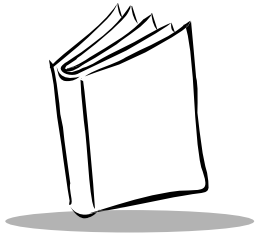
**Row Indicators** To help synchronize a PDF417 symbol's structure, codewords which collectively indicate which row a particular one is, which is the left and right side of that row, how many rows are in the symbol, what security level is encoded in the symbol, and how many data columns are in the rows. Left Row Indicators occur in each row immediately after the Start pattern; Right Row Indicators occur in each row immediately before the Stop pattern.

**Scan** Search for a symbol to be optically recognized.

**Scan Area** Area intended to contain a symbol.

| | |
|---|---|
| **Scanner** | An electronic device used to scan bar code symbols and produce a digitized pattern that corresponds to the bars and spaces of the symbol. Its three main components are:<br>Light Source (Laser Or Photoelectric Cell) - Illuminates A Bar Code.<br>Photodetector - Registers The Difference In Reflected Light (More Light Reflected From Spaces).<br>Signal Conditioning Circuit - Transforms Optical Detector Output Into A Digitized Bar Pattern. |
| **Self-checking Code** | A symbology that uses a checking algorithm to detect encoding errors within the characters of a bar code symbol. |
| **Show-through** | The generally undesirable property of a substrate that permits underlying markings to be seen. |
| **Space** | The lighter element of a bar code formed by the background between bars. |
| **Specular Reflection** | The mirror-like reflection of light from a surface, which can “blind” a scanner. |
| **Spot Size** | Size of the scanning aperture. |
| **Spots** | The presence of ink in a bar code's spaces or clear areas. These generally reduce the percent decode. |
| **Start/stop Character** | A pattern of bars and spaces that provides the scanner with start and stop reading instructions and scanning direction. The start and stop characters are the first and last encoded characters of a bar code. |
| **Substrate** | A foundation material on which a substance or image is placed. |
| **Substrate Scattering** | Optical phenomenon which causes bars to appear larger and spaces narrower than they are actually printed. It is caused by the scattering of incident light rays within the medium. |
| **Symbol** | A scannable unit that encodes data within the conventions of a certain symbology, usually including start/stop characters, quiet zones, data characters, and check characters. |
| **Symbol Aspect Ratio** | The ratio of symbol height to symbol width. |
| **Symbol Height** | The distance between the outside edges of the quiet zones of the first row and the last row. |

| | |
|---|---|
| **Symbol Length** | Length of symbol measured from the beginning of the quiet zone (margin) adjacent to the start character to the end of the quiet zone (margin) adjacent to a stop character. |
| **Symbology** | The structural rules and conventions for representing data within a particular bar code type (e.g. UPC/EAN, Code 39). |
| **Symmetric Bar Width Growth** | Uniform growth of bars evenly distributed. |
| **Tolerance** | Allowable Deviation From The Nominal Bar Or Space Width. |
| **Two-dimensional Symbology** | Designed for high information density and higher encoding capability than one-dimensional bar codes, a symbology which encodes data in both the horizontal (X-dimension) and vertical dimensions, usually in a “stacked” or multi-row arrangement. |
| **Upc** | Universal Product Code. A relatively complex numeric symbology. Each character consists of two bars and two spaces, each of which can be any of four widths. The standard symbology for retail food packages in the United States. |
| **Visible Laser Diode (Vld)** | A solid state device which produces visible laser light. Laser light emitted from the diode has a wavelength of 670 to 680 nanometers. |
| **Void** | Absence of ink within printed bars. |
| **X-dimension** | Width of the narrowest element (bar of space) in a bar code symbol. |
| **Y-dimension** | Element height, as applied to a two-dimensional symbology, which must equal or exceed a required minimum. |
| **Zero-suppressed Code** | A version of UPC/EAN which reduces the number of characters in the code. The resulting code combines the manufacturer's code and the product's code of Version A in a retail application. |



# Index

## C



## D



## E



## F



## H



## I



## L



## M

## N



## P

## R



## S



## T



## U

# Tell Us What You Think...

We'd like to know what you think about this Manual. Please take a moment to fill out this questionaire and fax this form to: (516) 738-3318, or mail to:

Symbol Technologies, Inc.
One Symbol Plaza M/S B-4
Holtsville, NY 11742-1300
Attn: Technical Publications Manager

IMPORTANT: If you need product support, please call the appropriate customer support number provided. Unfortunately, we cannot provide customer support at the fax number above.

User's Manual Title: _____
(please include revision level)

How familiar were you with this product before using this manual?

☐ Very familiar ☐ Slightly familiar ☐ Not at all familiar

Did this manual meet your needs? If not, please explain. _____

What topics need to be added to the index?, if applicable _____

What topics do you feel need to be better discussed? Please be specific. _____

What can we do to further improve our manuals? _____

Thank you for your input—We value your comments.