

# SE955

## Integration Guide





# ***SE955 Scan Engine Integration Guide***

*72E-72322-03*

*Revision A*

*October 2006*

© **2006** by Symbol Technologies, Inc. All rights reserved.

No part of this publication may be reproduced or used in any form, or by any electrical or mechanical means, without permission in writing from Symbol. This includes electronic or mechanical means, such as photocopying, recording, or information storage and retrieval systems. The material in this manual is subject to change without notice.

The software is provided strictly on an “as is” basis. All software, including firmware, furnished to the user is on a licensed basis. Symbol grants to the user a non-transferable and non-exclusive license to use each software or firmware program delivered hereunder (licensed program). Except as noted below, such license may not be assigned, sublicensed, or otherwise transferred by the user without prior written consent of Symbol. No right to copy a licensed program in whole or in part is granted, except as permitted under copyright law. The user shall not modify, merge, or incorporate any form or portion of a licensed program with other program material, create a derivative work from a licensed program, or use a licensed program in a network without written permission from Symbol. The user agrees to maintain Symbol’s copyright notice on the licensed programs delivered hereunder, and to include the same on any authorized copies it makes, in whole or in part. The user agrees not to decompile, disassemble, decode, or reverse engineer any licensed program delivered to the user or any portion thereof.

Symbol reserves the right to make changes to any software or product to improve reliability, function, or design.

Symbol does not assume any product liability arising out of, or in connection with, the application or use of any product, circuit, or application described herein.

No license is granted, either expressly or by implication, estoppel, or otherwise under any Symbol Technologies, Inc., intellectual property rights. An implied license only exists for equipment, circuits, and subsystems contained in Symbol products.

Symbol, Spectrum One, and Spectrum24 are registered trademarks of Symbol Technologies, Inc. Other product names mentioned in this manual may be trademarks or registered trademarks of their respective companies and are hereby acknowledged.

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

#### Patents

This product is covered by one or more patents listed on the Web site: <http://www.symbol.com/patents>.

# Revision History

Changes to the original manual are listed below:

Change	Date	Description
72E-72322-01	4/2006	Initial release
72E-72322-02	6/2006	Updates for gold connectors
72E-72322-03	10/2006	Updates for RoHS compliance, flex cables, and mechanical drawings



# Contents

## About This Guide

Chapter Descriptions .....	xiv
Configurations .....	xv
Notational Conventions .....	xv
Service Information .....	xv

## Chapter 1. Getting Started

Introduction .....	1-1
Theory of Operation .....	1-1
Scan Engine .....	1-2
Microprocessor .....	1-3
Simple Serial Interface (SSI) .....	1-4
Power Management .....	1-4
Electrical Interface .....	1-6
Beeper Definitions .....	1-7

## Chapter 2. Installation

Introduction .....	2-1
Grounding .....	2-1
ESD .....	2-1
Environment .....	2-1
Mounting .....	2-2
Installing the SE955 .....	2-3
Optical .....	2-4

Housing Design . . . . .	2-4
Wavefront Distortion . . . . .	2-4
Collection Beam Geometry . . . . .	2-5
Laser Clear Aperture . . . . .	2-5
Collection Clear Aperture . . . . .	2-6
Exit Window Materials . . . . .	2-6
Color . . . . .	2-7
Surface Quality . . . . .	2-7
Commercially Available Coatings . . . . .	2-7
Location and Positioning . . . . .	2-9
Symbol Position with Respect to a Fixed-Mount Scan Engine . . . . .	2-9
Exit Window Characteristics . . . . .	2-10
Accessories . . . . .	2-13
Flex Cables . . . . .	2-13
Connectors . . . . .	2-13
Hardware Accessories . . . . .	2-13
Scan Engine Developer Kit . . . . .	2-16
Regulatory Requirements . . . . .	2-17

## **Chapter 3. Replacing Existing Engines**

Introduction . . . . .	3-1
Replacing an SE824 with the SE955 Scan Engine . . . . .	3-2
Mounting . . . . .	3-2
Electrical . . . . .	3-2
Optical . . . . .	3-3
Mechanical . . . . .	3-3
Regulatory . . . . .	3-3
Replacing an SE923 with the SE955 Scan Engine . . . . .	3-4
Mounting . . . . .	3-4
Electrical . . . . .	3-4
Optical . . . . .	3-5
Mechanical . . . . .	3-5
Regulatory . . . . .	3-5
Replacing an SE1223WA with the SE955 Scan Engine . . . . .	3-6
Mounting . . . . .	3-6
Electrical . . . . .	3-7
Optical . . . . .	3-7
Mechanical . . . . .	3-7
Regulatory . . . . .	3-7

## **Chapter 4. SE955-I100R/E100R Specifications**

Introduction . . . . .	4-1
Technical Specifications . . . . .	4-2
Decode Zones . . . . .	4-5

## **Chapter 5. SE955-I105R/E105R Specifications**

Introduction . . . . .	5-1
Technical Specifications . . . . .	5-2



Decode Zones . . . . .	5-5
------------------------	-----

## Chapter 6. Regulatory Requirements

Regulatory Requirements . . . . .	6-1
Required Documentation for Class 1 Laser Products . . . . .	6-2
Required Documentation for Class 2 Laser Products . . . . .	6-2
Required Documentation for all End Products . . . . .	6-3
Required Labeling for Class 1 End Products . . . . .	6-3
1 - Certification Statement from FDA/IEC Label Set, 2005 . . . . .	6-3
2 - Identification . . . . .	6-3
3 - Laser Warning Label . . . . .	6-4
4 - Protective Housing Statement . . . . .	6-4
Required Labeling for Class 2 End Products . . . . .	6-5
1 - Certification Statement from FDA/IEC Label Set, 2005 . . . . .	6-5
2 - Identification . . . . .	6-5
3 - Laser Warning Label . . . . .	6-5
4 - Protective Housing Statements . . . . .	6-6
Recycling . . . . .	6-6
RoHS Compliance . . . . .	6-6

## Chapter 7. Application Notes

Introduction . . . . .	7-1
AC Electrical Characteristics . . . . .	7-1
Timing Waveforms . . . . .	7-3
Explanation Of The AC Symbols . . . . .	7-3

## Chapter 8. Parameter Menus

Introduction . . . . .	8-1
Operational Parameters . . . . .	8-1
Set Default Parameter . . . . .	8-7
Beeper Volume . . . . .	8-8
Beeper Tone . . . . .	8-9
Beeper Frequency Adjustment . . . . .	8-10
Laser On Time . . . . .	8-10
Aim Duration . . . . .	8-11
Scan Angle . . . . .	8-11
Power Mode . . . . .	8-12
Triggering Modes . . . . .	8-13
Time-out Between Same Symbol . . . . .	8-14
Beep After Good Decode . . . . .	8-14
Transmit "No Read" Message . . . . .	8-15
Parameter Scanning . . . . .	8-15
Linear Code Type Security Level . . . . .	8-16
Bi-directional Redundancy . . . . .	8-17
UPC/EAN . . . . .	8-18
Enable/Disable UPC-A . . . . .	8-18

Enable/Disable UPC-E .....	8-18
Enable/Disable UPC-E1 .....	8-19
Enable/Disable EAN-8 .....	8-19
Enable/Disable EAN-13 .....	8-20
Enable/Disable Bookland EAN .....	8-20
Decode UPC/EAN Supplementals .....	8-21
Decode UPC/EAN Supplementals (continued) .....	8-22
Decode UPC/EAN Supplemental Redundancy .....	8-22
Transmit UPC-A Check Digit .....	8-23
Transmit UPC-E Check Digit .....	8-23
Transmit UPC-E1 Check Digit .....	8-24
UPC-A Preamble .....	8-25
UPC-E Preamble .....	8-26
UPC-E1 Preamble .....	8-27
Convert UPC-E to UPC-A .....	8-28
Convert UPC-E1 to UPC-A .....	8-28
EAN Zero Extend .....	8-29
Convert EAN-8 to EAN-13 Type .....	8-29
UPC/EAN Security Level .....	8-30
UCC Coupon Extended Code .....	8-31
Code 128 .....	8-32
Enable/Disable Code 128 .....	8-32
Enable/Disable UCC/EAN-128 .....	8-32
Enable/Disable ISBT 128 .....	8-33
Lengths for Code 128 .....	8-33
Code 39 .....	8-34
Enable/Disable Code 39 .....	8-34
Enable/Disable Trioptic Code 39 .....	8-34
Convert Code 39 to Code 32 (Italian Pharma Code) .....	8-35
Code 32 Prefix .....	8-35
Set Lengths for Code 39 .....	8-36
Code 39 Check Digit Verification .....	8-37
Transmit Code 39 Check Digit .....	8-37
Enable/Disable Code 39 Full ASCII .....	8-38
Code 93 .....	8-39
Enable/Disable Code 93 .....	8-39
Set Lengths for Code 93 .....	8-40
Code 11 .....	8-41
Enable/Disable Code 11 .....	8-41
Set Lengths for Code 11 .....	8-41
Code 11 Check Digit Verification .....	8-43
Transmit Code 11 Check Digits .....	8-43
Interleaved 2 of 5 .....	8-44
Enable/Disable Interleaved 2 of 5 .....	8-44
Set Lengths for Interleaved 2 of 5 .....	8-45
I 2 of 5 Check Digit Verification .....	8-47
Transmit I 2 of 5 Check Digit .....	8-48

Convert I 2 of 5 to EAN-13 .....	8-48
Discrete 2 of 5 .....	8-49
Enable/Disable Discrete 2 of 5 .....	8-49
Set Lengths for Discrete 2 of 5 .....	8-50
Chinese 2 of 5 .....	8-51
Enable/Disable Chinese 2 of 5 .....	8-51
Codabar .....	8-52
Enable/Disable Codabar .....	8-52
Set Lengths for Codabar .....	8-53
CLSI Editing .....	8-54
NOTIS Editing .....	8-54
MSI .....	8-55
Enable/Disable MSI .....	8-55
Set Lengths for MSI .....	8-56
MSI Check Digits .....	8-57
Transmit MSI Check Digit .....	8-57
MSI Check Digit Algorithm .....	8-58
RSS .....	8-59
Enable/Disable RSS-14 .....	8-59
Enable/Disable RSS-Limited .....	8-59
Enable/Disable RSS-Expanded .....	8-60
Convert RSS to UPC/EAN .....	8-60
Transmit Code ID Character .....	8-61
Prefix/Suffix Values .....	8-62
Scan Data Transmission Format .....	8-63
Scan Data Transmission Format (continued) .....	8-64
Serial Parameters .....	8-65
Baud Rate .....	8-65
Parity .....	8-67
Software Handshaking .....	8-68
Decode Data Packet Format .....	8-69
Host Serial Response Time-out .....	8-69
Stop Bit Select .....	8-70
Intercharacter Delay .....	8-70
Host Character Time-out .....	8-70
Event Reporting .....	8-71
Decode Event .....	8-71
Boot Up Event .....	8-72
Parameter Event .....	8-72
Numeric Bar Codes .....	8-73
Cancel .....	8-74

## Chapter 9. Remote Scanner Management

Introduction .....	9-1
Hardware Signals .....	9-2
Protocol Commands .....	9-2
Attribute Support .....	9-3

RMD Protocol Over RS232 (SSI) .....	9-4
Encapsulation of RMD Commands/Responses over SSI .....	9-4
Example Transaction .....	9-5
RSM Protocol (Management Payload) .....	9-6
Error Commands .....	9-6
RMS_GET_PACKETSIZE .....	9-7
ATTRIBUTE_GETALL .....	9-8
ATTRIBUTE_GET .....	9-10
ATTRIBUTE_GETNEXT .....	9-14
ATTRIBUTE_GET_OFFSET .....	9-15
ATTRIBUTE_SET .....	9-16
ATTRIBUTE_STORE .....	9-17
Remote Monitoring Commands .....	9-18
Time to Decode .....	9-18
Internal Temperature .....	9-18
Motor Frequency .....	9-19
Part Number .....	9-19
Serial Number .....	9-20
Date of Manufacture .....	9-20
Date of Last Service .....	9-20
Firmware Version .....	9-20
Engine ID .....	9-21
Hardware Version .....	9-21

## Chapter 10. Simple Serial Interface

Introduction .....	10-1
Communication .....	10-1
SSI Message Formats .....	10-4
Engine Control Commands .....	10-4
SSI Transactions .....	10-27
General data transactions .....	10-27
Transfer of Decode Data .....	10-28
Communication Summary .....	10-30
RTS/CTS Lines .....	10-30
ACK/NAK Option .....	10-30
Number of Data Bits .....	10-30
Serial Response Time-out .....	10-30
Retries .....	10-30
Baud Rate, Stop Bits, Parity, Response Time-out, ACK/NAK Handshake .....	10-30
Errors .....	10-30
SSI Communication Notes .....	10-31

## Appendix A. Serial Interface Specification

Introduction .....	A-1
Terms and Definitions .....	A-1
Systems .....	A-1
Inactive .....	A-1

The Decoder and the Host ..... A-1

A Character ..... A-2

Data ..... A-2

Tolerances ..... A-2

Common Attributes ..... A-2

    The Decoder ..... A-3

    The Host ..... A-5

Transaction Examples ..... A-6

**Appendix B. Miscellaneous Code Information**

Introduction ..... B-1

UCC/EAN-128 ..... B-1

AIM Code Identifiers ..... B-3

Setting Code Lengths Via Serial Commands ..... B-6

Setting Prefixes and Suffixes Via Serial Commands ..... B-7

**Glossary**

**Index**

**Tell Us What You Think...**



## ***About This Guide***

The SE955 is a high performance miniature scan engine offering best in class, size, quality, reliability, durability and performance. The SE955 is built upon Symbol Technologies' long heritage of high-performance scan engines, and is a superior miniature scan engine, replacing the industry benchmark, the SE923. The SE955 has more features than any other scan engine available and delivers a new level of performance giving your products a competitive advantage.

The SE955 features include:

- Superior working range on all bar code densities.
- Steady and crisp easy to view scan line.
- 104 scans/second nominal.
- Fast decode time: typical 40 msec.
- Integration Flexibility - small size and lightweight to maximize customer's design.
- Low power consumption that maximizes battery life in portable devices.
- AIM mode for long range scanning.
- Blink mode.
- Flash upgradeable.
- Two different scan angles provide flexibility to customized application
- Custom default settings.
- Remote scan engine diagnostics/status reporting capability built in.
- Drop shocks of 2,000G.
- RoHS compliant upon product release.

The SE955 delivers a new level of performance in miniature scan engines and sets your product apart from the competition. With over 8 million scan engines installed worldwide, Symbol scan engines are unmatched for reliability, performance, durability and size.

The *SE955 Series Integration Guide* provides general instructions for mounting and set up of the SE955-I100R, SE955-E100R, SE955-I105R and 955-E105R scan engines as well as instruction for replacing existing Symbol SE824, SE923 or SE1223WA scan engine with an SE955.



This guide provides general instructions for the installation of the scan engine into a customer's device. It is recommended that an opto-mechanical engineer perform an opto-mechanical analysis prior to integration.

## Chapter Descriptions

The following is a description of each chapter in this guide.

- [Chapter 1, Getting Started](#) provides an overview, theory of operation, and power management information for the engine and decoder.
- [Chapter 2, Installation](#) describes how to install the engine, and provides considerations for ESD, optical, and positioning aspects.
- [Chapter 3, Replacing Existing Engines](#) provides information for replacing existing scan engines with the SE955.
- [Chapter 4, SE955-I100R/E100R Specifications](#) provides the technical specifications for the SE955 3.3 volt engine.
- [Chapter 5, SE955-I105R/E105R Specifications](#) provides the technical specifications for the SE955 5 volt engine.
- [Chapter 6, Regulatory Requirements](#) provides regulatory guidelines for properly marking product for regulatory approvals.
- [Chapter 7, Application Notes](#) describes the electrical characteristics of the imaging system and provides timing waveforms.
- [Chapter 8, Parameter Menus](#) provides the bar codes necessary to program the scan engine system.
- [Chapter 9, Remote Scanner Management](#) provides the hardware signals, protocol commands and attribute support for Remote Scanner Management.
- [Chapter 10, Simple Serial Interface](#) describes the system requirements of the Simple Serial Interface (SSI), which provides a communications link between Symbol Technologies decoders and a serial host.
- [Chapter A, Serial Interface Specification](#) describes the requirements for digital systems to exchange asynchronous serial data, and provides transaction examples.
- [Chapter B, Miscellaneous Code Information](#) provides information on AIM code identifiers and prefix/suffix values.



## Configurations

Available versions of the SE955 scan engine include:

- SE955-I100R - Class 2, 3.3v decoded engine; gold connector
- SE955-E100R - Class 1, 3.3v decoded engine; gold connector.
- SE955-I105R - Class 2, 5.0v decoded engine; gold connector.
- SE955-E105R - Class 1, 5.0v decoded engine; gold connector.

## Notational Conventions

The following conventions are used in this document:

- *Italics* are used to highlight specific items in the general text, and to identify chapters and sections in this and related documents.
- 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.

## Service Information

If you have a problem with your equipment, contact the Symbol Global Customer Interaction Center for your region (see below for contact information). Before calling, have the model number, serial number, and several of your bar code symbols at hand.

Call the Global Customer Interaction 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 Global Customer Interaction 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.

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.

For service information, warranty information or technical assistance contact or call the Symbol Global Customer Interaction Center. For contact information, go to: [www.symbol.com/contactsupport](http://www.symbol.com/contactsupport).

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

For the latest version of this guide go to: <http://www.symbol.com/manuals>.



## **Introduction**

The SE955 is a miniaturized, high performance laser based, single line, decoded bar code scan engine.

## **Theory of Operation**

The SE955 is a scan engine combined with a microprocessor to control the functionality of the engine, perform software decoding of the bar code information and provide a communication link to the host computer.

The scan engine provides the following functions:

- laser drive circuit controlling a 650 nm laser diode
- scan element drive circuit controlling a resonant single line scan element
- analog receiver with circuitry to identify the bar and space locations in the received waveform
- temperature sensor
- power on reset functionality.

The microprocessor section provides the following functions:

- non-volatile memory for storing user preferences for decoder capability parameters
- runs the bar code decoder software
- watchdog timer.

A host Simple Serial Interface (SSI) provides the following functions:

- low current beeper line (BPR\*) to provide beep signals
- decode LED output line (DLED\*) to indicate a successful decode

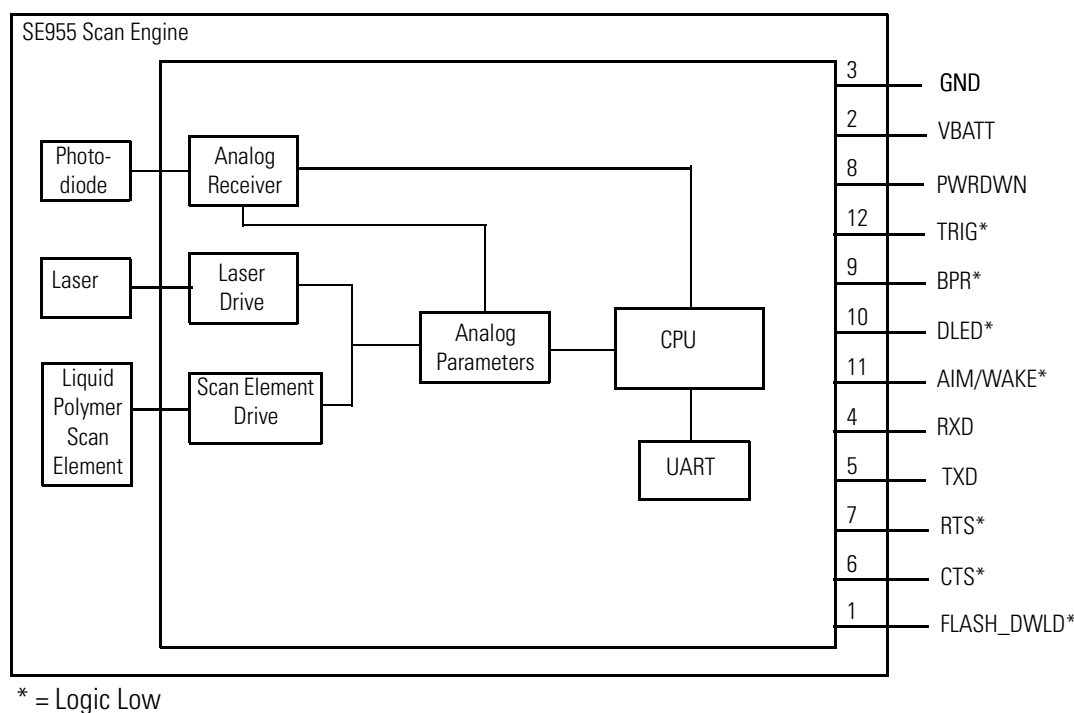
- signal to indicate that the unit can be powered down (PWRDWN)
- two serial I/O lines (RXD and TXD)
- two hardware handshaking lines (CTS\* and RTS\*)
- hardware trigger line (TRIG\*) and a hardware Aim/wake-up line (AIM/WKUP\*)
- line (FLASH\_DWLD\*) to support re-flashing the product software through the SSI interface
- power and ground.

## ***Scan Engine***

The basic functionality of a scan engine is outlined below:

- A laser diode emits a coherent beam of light focused to a diameter appropriate for the bar code densities to be read.
- The laser beam strikes the mirror of the scan element. This mirror oscillates about its vertical axis and causes the beam to be deflected, forming the outgoing scan line.
- As the laser spot is swept across the bar code it is either reflected off the white spaces or absorbed by the black bars.
- A collection mirror tracks the location of the laser spot on the bar code, collects the reflected light and focuses it onto the receiver photodiode.
- The photodiode is a transducer that converts optical energy to electrical current. This current is fed into the analog signal processing circuitry.
- The analog signal processing circuitry amplifies, filters and edge enhances the signal returned from the bar code. These edges represent the place when the laser transitioned between a bar and a space, and represents the information contained in the bar code.
- The digitizer circuitry generates a digital waveform whose ones and zeros represent the widths of the bars and spaces in the bar code. This waveform is called the Digital Bar Pattern (DBP).

- The DBP is sent to the local microprocessor to be decoded.



**Figure 1-1. SE955 Scan Engine Block Diagram**

The laser drive uses multiple forms of feedback (optical and electrical) to control the diode laser to emit constant optical power, and to ensure compliance with the laser regulatory standards, described in [Chapter 6, Regulatory Requirements](#).

The scan element is a mirror and magnet assembly cantilevered on a spring. This is a resonant system with a natural frequency of 52 Hz resulting in 104 scans per second. Alternating current forced through a drive coil mounted adjacent to the magnet causes the mirror to deflect to either side of its steady state position. This deflection causes the laser spot to be scanned across the bar code. A feedback coil coaxial with the drive measures the amplitude of the scan element and is used to set the scan amplitude. The SE955 is factory calibrated to generate two user selectable scan angles, 35° and 47° (default).

## Microprocessor

The SE955 utilizes a microprocessor to drive the SSI host interface, to control the laser scanning functional blocks, and to perform general decoder maintenance. A new feature being introduced by the SE955 decoded scan engine is support for Remote Scanner Management (RSM). For example, through SSI commands, the host can poll the SE955 for a measurement of temperature, as measured by circuitry on the PCB. For a full listing of the information that is available, see [Chapter 9, Remote Scanner Management](#) and [Chapter 10, Simple Serial Interface](#).

The micro-controller contains a watchdog timer. The enabling/disabling and maintenance of this watchdog are internal to the SE955; the host cannot configure the watchdog. The decoder's reset circuitry holds the micro-controller in reset after power-up to allow sufficient time for hardware initialization. This reset period is 23 msec. A reset can occur upon power up, or power supply voltage falling below 2.7 V.

## Simple Serial Interface (SSI)

The SE955 scan engine is host controlled through the Simple Serial Interface (see [Table 1-3](#)), and supports various triggering modes of operation (see [Triggering Modes on page 8-13](#)), including:

- Scan mode
- Aim mode which provides a laser aim dot
- Blink mode for presentation scanning
- Continuous mode.



In Scan mode the pre-selected scan amplitude is used. I.e., 35° and 47° (default). Also see, [Scan Angle on page 8-11](#).

The Aim mode is used to provide a laser aim dot, which can be used to pre-align the scan engine to a bar code before scanning. To aim then scan, the host would control the engine using the hardware AIM/WKUP\* then TRIG\* lines at the SSI interface (see [Table 1-3](#)), or by SSI commands (see [Chapter 10, Simple Serial Interface](#)).

The Blink mode can be used for triggerless operation in presentation scanning applications. To minimize power consumption, low duty cycle scanning is performed until a change in background is detected. Once detected, high duty cycle scanning is performed until the bar code, if one is present, is decoded. The scan engine would then return to low duty cycle scanning until the next change in background is detected (see [Triggering Modes on page 8-13](#)).

The Continuous Scanning mode is the mode in which the scan engine is always scanning and decoding (see [Power Mode on page 8-12](#)).

## Power Management

The SE955 has two power states (Awake and Sleep) and two power modes (Continuous Power and Low Power).

### Power States

WAKEUP and SLEEP commands (see [WAKEUP on page 10-25](#) and [SLEEP on page 10-22](#)), are sent to the scan engine to set the Power state to Awake or Sleep. The Low Power mode has an automatic timer that puts the unit into the Sleep state after a specified period of time.

When the SE955 is in the Sleep power state the PWRDWN signal (see [Table 1-3](#)) is asserted. The host uses this signal to remove power from the SE955. Do not remove power without using this signal since the PWRDWN signal is the only indication if the decoder is not transmitting, receiving, decoding, or writing data to non-volatile memory.

### Power Modes

Power modes are controlled by the Power Mode parameter (see [Power Mode on page 8-12](#)).

- In **Continuous Power** mode, the scan engine remains in the Awake state after each decode attempt. The Continuous Power mode parameter (see [Power Mode on page 8-12](#)) sets the SE955 to remain in the Awake power state unless it receives a SLEEP command. In this mode, the SE955 can switch power states using the SLEEP and WAKEUP commands (see [SLEEP on page 10-22](#) and [WAKEUP on page 10-25](#)); automatic power state switching is not supported.
- In **Low Power** mode, the scan engine enters into a low power consumption Sleep state whenever possible (provided all WAKEUP commands were released), drawing less current than in Continuous Power mode. This makes the Low Power mode more suitable for battery powered applications. The Low Power mode also allows the SE955 to switch power states using the SLEEP and WAKEUP commands (see [SLEEP on page 10-22](#) and [WAKEUP on page 10-25](#)). The SE955 must be awakened from the Sleep power state before performing any functions.

Table 1-1 shows how to put the SE955 into Low Power mode. Table 1-2 shows how to awaken it.

**Table 1-1. Putting the SE955 into Low Power Mode**

Action	Behavior
<b>Set the Power Mode parameter to Low Power</b>	The SE955 enters Low Power mode and automatically switches to the Sleep power state whenever possible.
<b>Send the serial SLEEP command</b>	The SE955 enters Sleep power state only once, as soon as possible.
<b>Note:</b> All wake up signals (see Table 1-2) must be inactive to enter Sleep power state. Once the <b>SE955</b> is awakened, at least 1 second must elapse before it re-enters Low Power mode.	

**Table 1-2. Waking Up the SE955**

Signal	State to Wake Up
<b>AIM/WKUP*</b>	Low
<b>TRIG*</b>	Low
<b>CTS*</b>	Low
<b>RXD</b>	Send 0x00
Signal names with the "*" modifier are asserted when at the positive logic 0 state (active low). Signal names without the "*" modifier are asserted when at the positive logic 1 state (active high).	

When the SE955 is awakened, it remains awake for at least 1 second before re-entering Low Power mode. The host must perform its first action within the 1 second time period if the power mode parameter is set to Low Power.

## Electrical Interface

Table 1-3 lists the pin functions of the SE955 interface and illustrates typical input and output circuitry for the SE955-I100R, SE955-E100R, SE955-I105R and SE955-E105R. The SE955-I100R/E100R accepts a 3.3 VDC +/- 10% power input, designated as  $V_{BATT}$ . The SE955-I105R/E105R accepts a 3.2 VDC to 5.5 VDC power input, designated as  $V_{BATT}$ .

**Table 1-3. Electrical Interface**

Mnemonic	Pin No.	Type	Description
$V_{BATT}$	2	PWR	<b>Power Supply:</b> <b>SE955-I100R/E100R:</b> 3.0 to 3.6 VDC <b>SE955-I105R/E105R:</b> 3.2 to 5.5 VDC.
GND	3	PWR	<b>Ground</b>
AIM/WAKE*	11	I	<b>Wake Up:</b> When the SE955 is in low power mode, pulsing this pin low for 200 nsec awakens the SE955. <b>AIM:</b> This pin provides a hard wired trigger line that creates an AIM pattern (a spot). This spot allows positioning the bar code and laser beam alignment to maximize the scan capability of the SE955. Aim mode is not supported on the SE955-E100R.
FLASH_DWLD*	1	I	<b>Flash Down Load:</b> Do not drive high. Pull low for download.
RXD	4	I	<b>Received Data:</b> Serial input port.
CTS*	6	I	<b>Clear to Send:</b> Serial port handshaking line.
TRIG*	12	I	<b>Trigger:</b> Hardware triggering line. Driving this pin low causes the SE955 to start a scan and decode session.
TXD	5	O	<b>Transmitted Data:</b> Serial output port.
RTS*	7	O	<b>Request to Send:</b> Serial port handshaking line.
PWRDWN	8	O	<b>Power Down Ready:</b> When high, the decoder is in low power mode.
BPR*	9	O	<b>Beeper:</b> Low current beeper output.
DLED*	10	O	<b>Decode LED:</b> Low current decode LED output.
Notes: 1. SE955-I105R input lines diode isolated to prevent back biasing of the engine. 2. Signal names with the "*" modifier are asserted when at the ground level (logic 0, active low). 3. Signal names without the "*" modifier are asserted when at the positive supply voltage level (logic 1, active high).			

**Table 1-4. Input/Output Characteristics**

Input Characteristics			Output Characteristics			
SE955-I100R/E100R			SE955-I100R/E100R			
	Min	Max		Min	Max	Conditions
V <sub>IL</sub>		V <sub>BATT</sub> *0.2	V <sub>OL</sub>		0.4 V	I <sub>OL</sub> = 0.8mA
V <sub>IH</sub>	V <sub>BATT</sub> *0.8		V <sub>OH</sub>	V <sub>BATT</sub> - 0.5 V		I <sub>OH</sub> = -0.2mA
				V <sub>BATT</sub> - 1.0 V		I <sub>OH</sub> = -1.0mA
SE955-I105R/E105R			SE955-I105R/E105R			
V <sub>IL</sub>		0.28 V	V <sub>OL</sub>		0.1 V	I <sub>OL</sub> = 0.1mA
V <sub>IH</sub>	2.26 V				0.18 V	I <sub>OL</sub> = 1.0mA
			V <sub>OH</sub>	V <sub>BATT</sub> - 0.1 V		I <sub>OH</sub> = -0.1mA
				2.98 V		I <sub>OH</sub> = -1.0mA
						V <sub>BATT</sub> - 3.2 V
				4.49 V		I <sub>OH</sub> = -1.0mA
						V <sub>BATT</sub> - 5.5 V



## Beeper Definitions

The SE955 issues different beep sequences and patterns to indicate status. [Table 1-5](#) defines beep sequences that occur during both normal scanning and while programming the scan engine.

**Table 1-5. Beeper Definitions**

Beeper Sequence	Indication
<b>Standard Use</b>	
Low/medium/high beeps	Power up.
Short high beeps	A bar code symbol was decoded (if decode beeper is enabled).
4 long low beeps	A transmission error was detected in a scanned symbol. The data is ignored. This occurs if a unit is not properly configured. Check option setting.
5 low beeps	Conversion or format error.
Hi/hi/hi/lo beeps	RS232 receive error.
<b>Parameter Menu Scanning</b>	
Short high beeps	Correct entry scanned or correct menu sequence performed.
Lo/hi beeps	Input error, incorrect bar code or "Cancel" scanned, wrong entry, incorrect bar code programming sequence; remain in program mode.
Hi/lo beeps	Keyboard parameter selected. Enter value using bar code keypad.
Hi/lo/hi/lo beeps	Successful program exit with change in the parameter setting.
Low/hi/low/hi beeps	Out of host parameter storage space. Scan <a href="#">Set Default Parameter on page 8-7</a> .



## Introduction

This chapter provides information for mounting and installing the SE955 scan engine, including physical and electrical considerations and recommended window properties.

### Grounding



CAUTION

*The SE955 chassis is connected to GROUND. If you are installing the SE955 to a hot or powered host, you must isolate the two. The best integration practice is to avoid ground loops wherever possible. There is a potential for creating a ground loop by grounding the SE955 chassis to the ground of the system in which the SE955 is being integrated.*

An insulator can be inserted between the two chassis, and if metallic (non-magnetic) screws are used, shoulder washers must be used to isolate the screws from the host. Non-metallic screws may also be used if mechanical considerations permit.

### ESD

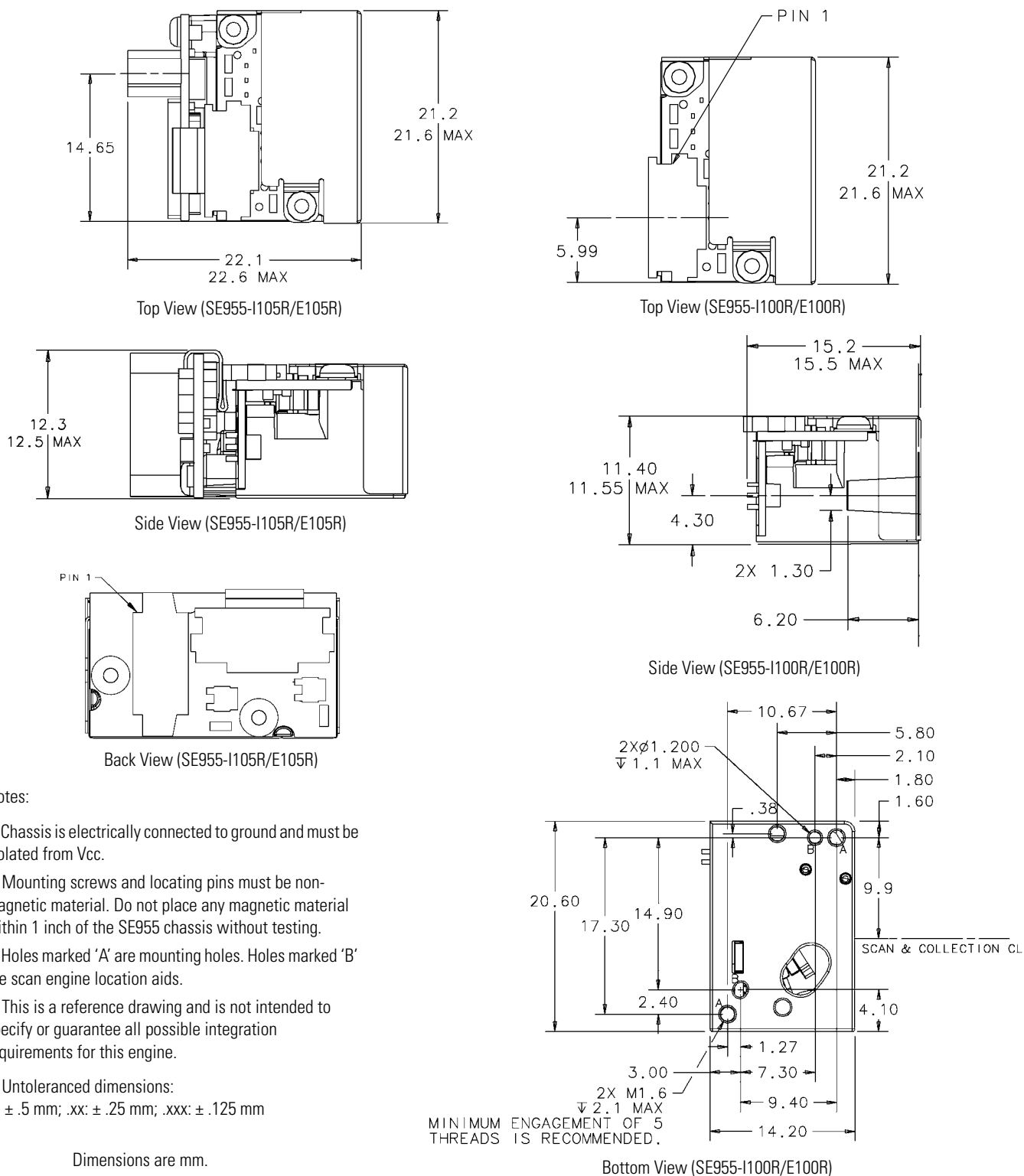
The SE955 is protected from ESD events that may occur in an ESD-controlled environment. Always exercise care when handling the module. Use grounding wrist straps and handle in a properly grounded work area.

### Environment

The SE955 must be sufficiently enclosed to prevent dust particles from gathering on the mirrors, laser lens, and the photodiode. Dust and other external contaminants eventually cause degradation in unit performance. Symbol does not guarantee performance of the engine when used in an exposed application.

## Mounting

There are two mounting holes (M1.6 x 0.35), and two locator holes on the bottom of the chassis (see [Figure 2-1](#)). The SE955 can be mounted in any orientation with no degradation in performance.



### Figure 2-1. SE955 Mounting Diagram

## Installing the SE955

Before installing the SE955 into the host equipment, consider five important points:

- The SE955 chassis is electrically connected to ground. It must be isolated from power and ground.
- Use only non-magnetic screws (i.e. stainless steel 300 Series screws), or locating pins when mounting the SE955. Magnetic screws or pins can cause the scan element/mirror neutral position to change. Recommended screw torque is shown in [Table 2-1](#).

**Table 2-1. Screw Torque**

	Recommended
<b>Standard</b>	$10 \pm 2$ oz-in
<b>Metric</b>	$0.72 \pm 0.14$ kg-cm

- It is strongly recommended that you use a thread locking method, such as a Nylok patch (a thread locking compound pre-applied to the screws).
- Do not place magnetic material (e.g., dynamic speakers, ringers, vibrators, inductors, metal parts) within 1 inch of the SE955 chassis. The SE955 scan element used to generate the scan line has a magnet on one end. Locating magnetic or ferrous material near the scan engine may influence the pointing of the scan line emitted from the engine. Evaluate placement of all magnetic or ferrous material during system layout to determine if 1 inch is sufficient.
- Leave sufficient space to accommodate the maximum size of the engine.



CAUTION

*When using metallic non-magnetic screws, ensure that the screwdriver or screw tip that you use is non-magnetic. Magnetic screwdrivers or screw tips change the scan element/mirror neutral position. Note that magnetic screwdrivers do not permanently alter pointing, as long as they are removed.*



CAUTION

*When integrating scan engines into their final destination, adhesives may be required. High quality optical surfaces are sensitive to out-gassing from adhesives such as cyanoacrylates (super glue). Out-gassing is the release of a gas/vapor/particulate trapped in the adhesive. Most out-gassing occurs while the adhesive is curing. During this time, particulate can collect on critical surfaces and reduce engine performance. Therefore, Symbol strongly recommends using very low out-gassing/blooming adhesives such as acrylics or epoxies when adhesives are required.*

## Optical

The SE955 uses a sophisticated optical system that provides scanning performance that matches or exceeds the performance of much larger scan engines. The performance of the scan engine is not affected by a properly designed enclosure.



This guide provides general instructions for the installation of the scan engine into a customer's device. It is recommended that an opto-mechanical engineer perform an opto-mechanical analysis prior to integration.

The following guidelines aid the Optical Engineer in design and specification of the window and enclosure.

### Housing Design

The orientation of the exit window has a large effect on scan engine performance. See tables [2-4](#), [2-5](#), [2-6](#), [2-7](#) beginning on [page 2-10](#) for exit window distances. In addition to providing obstacle-free paths for outgoing and incoming light, a good housing design ensures that the outgoing laser light reflected off of the window back into the housing is attenuated sufficiently before reaching the detector.

Unwanted laser light reaching the detector is termed "stray light". As a goal, stray light should be kept below 5 nanowatts for full range performance. Stray light is difficult to model and is highly dependent upon the housing design. It is influenced by the placement of the exit window and the surface properties of the components in the immediate vicinity of the scan engine. The surface color and finish of components surrounding the engine must be considered. Black surfaces can absorb as much as 90%-98% of the incident light. Smooth specular reflecting surfaces can be used to steer stray light away from the engine. Diffuse surfaces can be used to attenuate the light by spreading the reflected light over a wide range of angles. Use caution if the scan line reflects off of circuit boards. Traces and solder pads behave like mirrors and can inadvertently cause performance degradation.

The position of the detector creates a Side Field of View area where ambient light into this area can affect scanner performance. See [Figure 2-6 on page 2-11](#) for an illustration of the Side Field of View. Although not required, it is recommended that the housing be designed to block ambient light into the Side Field of View to increase performance at 10,000 FCD.

The tilt of the exit window is properly determined by ray tracing the exit beam reflection off of the window, and ensuring that the reflected light is directed away from the inside of the scan engine. This analysis should include the positional and angular tolerances of the scan engine and exit window. Recessing the window into the housing is also recommended to prevent scratches on the window. In keeping with good practice, a proper design should be supplemented with testing and verification.

The height and width of the exit window is determined such that the outgoing laser beam and return light is not clipped. See [Figure 2-6 on page 2-11](#), [Table 2-4](#) and [Table 2-5 on page 2-10](#) for recommended minimum widths at various window positions. It is highly recommended to analyze additional positioning tolerance of the scan engine based on your specific application and increase window size accordingly.



SE955 performance is not sensitive to exit window thickness. However, window thickness is application dependent. For most applications it is 1.0 mm to 2.0 mm (.039 in to .079 in).

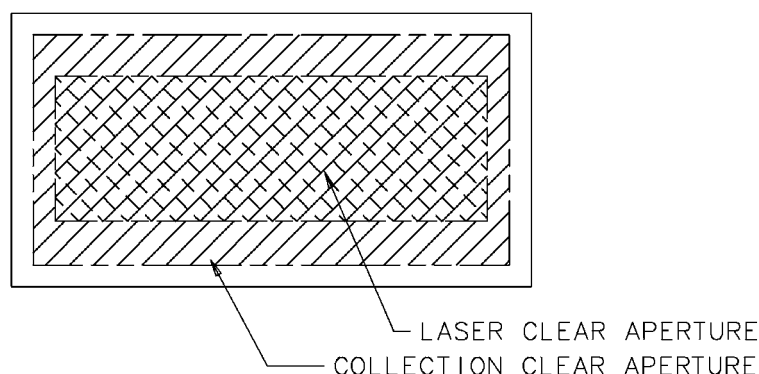
### Wavefront Distortion

Wavefront distortion is a measure of the window's optical quality. Since the optical requirements of the exit window are different for the exit and entrance beam envelopes, a laser clear aperture and the collection clear aperture are defined. The laser clear aperture requires high optical performance, and the collection clear aperture requires fair optical performance. Refer to [Figure 2-2](#) for the location of the two apertures.

The following Wavefront Distortion specifications are recommended:

Wavefront Distortion (transmission) measured at 633 nm

1. Within laser clear aperture: Over any 1.0 mm diameter area.
  - optical power measured in any direction: <0.050 waves
  - irregularities after subtracting optical power and astigmatism: <0.120 waves (P-V) and < 0.015 waves (RMS).
2. Within collection clear aperture: < 10 waves (P-V).



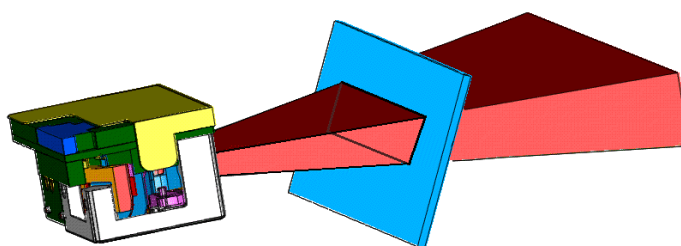
**Figure 2-2. Clear Apertures**

### ***Collection Beam Geometry***

Figure 2-2 also illustrates the beam envelope entering the scan engine. Ensure that the collection path is free of obstructions for full scan angle performance.

### ***Laser Clear Aperture***

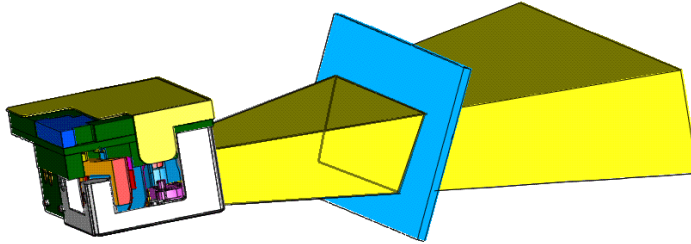
The laser clear aperture is the area on the exit window that intersects the exit beam envelope as shown in Figure 2-3. Note that at any instance in time, the outgoing laser beam is collimated and approximately 1 mm in diameter, while during scanner operation the beam is constrained within the exit beam envelope. For dimensions and information about clear aperture calculations see [Exit Window Characteristics on page 2-10](#) and [Exit Window Positioning on page 2-11](#)



**Figure 2-3. Exit Beam Envelope**

### **Collection Clear Aperture**

As shown in Figure 2-4, the collection clear aperture is the area on the exit window which intersects the collection beam envelope. In both cases, ensure that the paths are free of obstructions. Also incorporate a minimum of a 0.020" to 0.040" spacing between the clear apertures and the window borders.



**Figure 2-4. Entrance Beam Envelope**

### **Exit Window Materials**

Many window materials that look perfectly clear to the eye can contain stresses and distortions which affect the laser beam and reduce scan engine performance. For this reason, only optical glass or cell cast plastics are recommended. Following are descriptions of three popular exit window materials:

- Poly-methyl Methacrylic (PMMA)
- Allyl Diglycol Carbonate (ADC)
- Chemically tempered float glass.

#### **Cell Cast Acrylic (ASTM: PMMA)**

Cell Cast Acrylic, or Poly-methyl Methacrylic is fabricated by casting acrylic between two precision sheets of glass. This material has very good optical quality, but is relatively soft and susceptible to attack by chemicals, mechanical stress and UV light. It is strongly recommended to have acrylic hard-coated with Polysiloxane to provide abrasion resistance and protection from environmental factors. Acrylic can be laser-cut into odd shapes and ultrasonically welded.

#### **Cell Cast ADC, Allyl Diglycol Carbonate (ASTM: ADC)**

Also known as CR-39™, ADC, a thermal setting plastic widely used for plastic eyeglasses, has excellent chemical and environmental resistance. It also has an inherently moderate surface hardness and therefore does not require hard-coating. This material cannot be ultrasonically welded.

#### **Chemically Tempered Float Glass**

Glass is a hard material which provides excellent scratch and abrasion resistance. However, unannealed glass is brittle. Increased flexibility strength with minimal optical distortion requires chemical tempering. Glass cannot be ultrasonically welded and is difficult to cut into odd shapes.



## Abrasion Resistance

To gauge a window's durability, quantify its abrasion resistance using ASTM standard D1044, Standard Test Method for Resistance of Transparent Plastics to Surface Abrasion. Also known as the Taber Test, this measurement quantifies abrasion resistance as a percent increase in haze after a specified number of cycles and load. Lower values of the increase in haze correspond to better abrasion and scratch resistance. Refer to [Table 2-2](#).

**Table 2-2. Taber Test Results on Common Exit Window Materials**

Sample	Haze 100 cycles	Haze 500 cycles	Abrasion Resistance
Chemically Tempered Float Glass	1.20%	1.50%	Best
PMMA with Polysiloxane Hardcoat	3%	10%	
ADC	5%	30%	
PMMA	30%		Worst
* All measurements use a 100 gram load and CS-10F Abraser			

## Anti-Reflection (AR) Coatings

Anti-reflection coatings may be used for stray light control or to achieve maximum working range, however, they are expensive and therefore not recommended. Also, AR coatings have very poor abrasion and scratch resistance, making only single side AR coatings practical (the AR coated side of the window would face the interior of the scanner).

## Color

Plastic is available in a wide range of colors. Exit windows can be colored if desired as long as the optical transmission is in the spectral region between 640 nm and 670 nm (a minimum of 85%).

## Surface Quality

Surface quality refers to residual defects on the surfaces of the window. The recommended window specification for this follows the US Military Specification Standard MIL-O-13830A for scratch and dig performance.

Surface Quality: 60-20 per MIL-O-13830A

## Commercially Available Coatings

[Table 2-3 on page 2-7](#) lists some exit window manufacturers and anti-reflection coaters.

## Polysiloxane Coating

Polysiloxane type coatings are applied to plastic surfaces to improve the surface resistance to both scratch and abrasion. They are generally applied by dipping and then allowed to air dry in an oven with filtered hot air.

**Table 2-3. Exit Window Manufacturers and Coaters**

Company	Discipline	Specifics
Evaporated Coatings, Inc. 2365 Maryland Road Willow Grove, PA 19090 (215) 659-3080	Anti-reflection coater	Acrylic window supplier Anti-reflection coater
Fosta-Tek Optics, Inc. 320 Hamilton Street Leominster, MA 01453 (978) 534-6511	Cell-caster, hard coater, laser cutter	CR39 exit window manufacturer

**Table 2-3. Exit Window Manufacturers and Coaters (Continued)**

<b>Company</b>	<b>Discipline</b>	<b>Specifics</b>
Glasflex Corporation 4 Sterling Road Sterling, NJ 07980 (908) 647-4100	Cell-caster	Acrylic exit window manufacturer
Optical Polymers Int. (OPI) 110 West Main Street Milford, CT 06460 (203)-882-9093	CR-39 cell-caster, coater, laser cutter	CR39 exit window manufacturer
Polycast 70 Carlisle Place Stamford, CT 06902 800-243-9002	acrylic cell-caster, hard coater, laser cutter	Acrylic exit window manufacturer
TSP 2009 Glen Parkway Batavia, OH 45103 800-277-9778	acrylic cell-caster, coater, laser cutter	Acrylic exit window manufacturer

## Location and Positioning



Integrate the scan engine in an environment no more extreme than the product's specification, where the engine does not exceed its temperature range. For instance, do not mount the engine on to or next to a large heat source. When placing the engine with another device, ensure there is proper convection or venting for heat. Follow these suggestions to ensure product longevity, warranty, and overall satisfaction with the scan engine.

### ***Symbol Position with Respect to a Fixed-Mount Scan Engine***

It is sometimes necessary to mount the SE955 in such a way that it is able to read symbols that are automatically presented to it, or that are always presented in a pre-determined location. In these situations positioning of the SE955 with respect to the symbol location is critical. Failure to properly position the scan engine and symbol may lead to unsatisfactory reading performance.

The SE955 can be programmed to two different scan angles. It is recommended that the position of the scan engine is set using the widest scan angle (47°). Setting the position for the narrow scan angle (35°) and then changing the scan angle might cause clipping of the laser beam against the housing.

Following is a series of steps you should take to ensure satisfactory operation of the SE955 in your installation:

1. Determine the optimum distance between the scan engine and the symbol. Due to the large variety of symbol sizes, densities, print quality, etc., there is no simple formula to calculate this optimum symbol distance. Try this:
  - a. Measure the maximum and minimum distance at which your symbols can be read.
  - b. Locate the scan engine so the symbol is near the middle of this range when being scanned.

Check the near and far range on several symbols. If they are not reasonably consistent there may be a printing quality problem that can degrade the performance of your system. Symbol Technologies can provide advice on how to improve your installation.
2. Center the symbol (left to right) in the scan line whenever possible.
3. Position the symbol so that the scan line is as near as possible to perpendicular to the bars and spaces in the symbol.
4. Avoid specular reflection (glare) off the symbol by tilting the top or bottom of the symbol away from the engine. The exact angle is not critical, but it must be large enough so that if a mirror were inserted in the symbol location, the reflected scan line would miss the front surface of the engine. See [Exit Window Characteristics on page 2-10](#) for maximum angles.
5. If a window is to be placed between the engine and the symbol, the determination of optimum symbol location should be made with a representative window in the desired window position. Read the sections of this chapter concerning window quality, coatings and positioning.
6. Give the scan engine time to dwell on the symbol for a minimum of 40 msec. Poor quality symbols take longer to decode. When first enabled, the scan engine may take two or three scans before it reaches maximum performance. Enable the scan engine before the symbol is presented, if possible.

Exit Window Characteristics

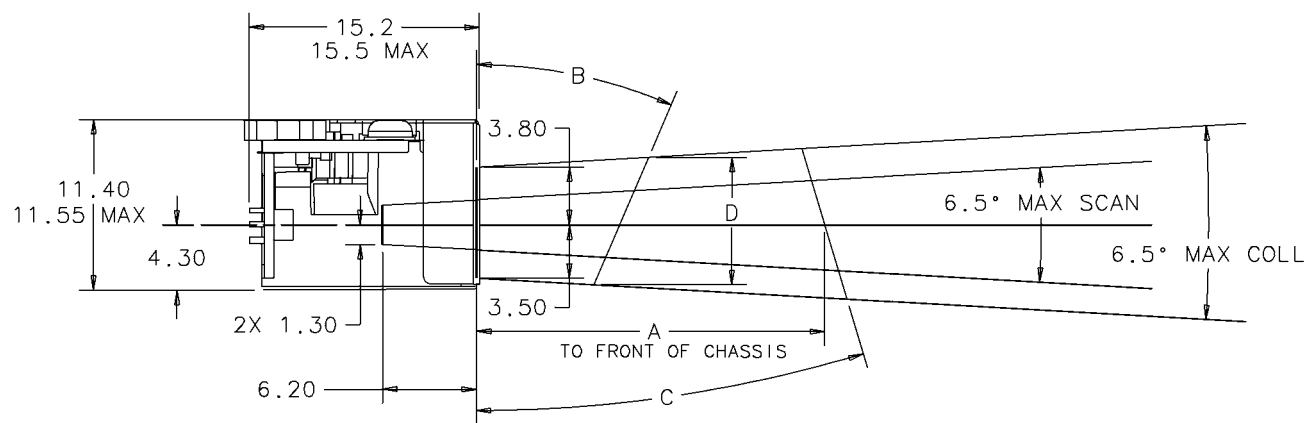


Figure 2-5. Exit Window Tilt Angle

Table 2-4. Exit Window Distance from Scan Engine: 0.15 in - 0.36 in (3.8 mm - 9 mm)

A	Distance from Scan Engine on center line (in./mm)*	0.15/ 3.8	0.156/ 4	0.18/ 4.5	0.20/ 5	0.22/ 5.5	0.24/ 6	0.25/ 6.35	0.26/ 6.5	0.28/ 7	0.31/ 8	0.36/ 9
B	Minimum Window Positive Tilt (degrees)	36.0	35.0	32.5	31.0	29.0	27.5	26.5	26.0	25.0	22.5	20.5
C	Minimum Window Negative Tilt (degrees)	34.0	33.5	31.0	29.5	27.5	26.0	25.0	24.5	23.5	21.5	19.5

Table 2-5. Exit Window Distance from Scan Engine: 0.39 in - 2.00in (10 mm - 50.8 mm)

A	Distance from Scan Engine on center line (in./mm)*	0.39/ 10	0.48/ 12	0.50/ 12.7	0.55/ 14	0.75/ 19	1.00/ 25.4	1.25/ 31.8	1.50/ 38	1.75/ 44.5	2.00/ 50.8
B	Minimum Window Positive Tilt (degrees)	19.0	17.0	16.5	15.0	12.0	10.0	9.0	8.0	7.5	7.0
C	Minimum Window Negative Tilt (degrees)	18.5	16.0	15.5	14.5	12.5	9.5	8.5	7.5	7.0	6.5

## Notes:

1. Chassis is electrically at ground.
2. Maximum horizontal scan/collection envelope (denotes max. scan/max. coll in top views) = nominal scan angle + tolerance.
  - a. Two programmable nominal scan angles: 35°, 47°
  - b. Total tolerance = 10°, includes:
    - i. Scan angle tolerance:  $\pm 3^\circ$  Typ.
    - ii. Pointing error:  $\pm 3^\circ$  Typ.
    - iii. Pointing shift after 2000G shock:  $\pm 1.5^\circ$  Typ.
3. Maximum vertical scan/collection envelope (denotes max. scan/max. coll in side views) = nominal scan line + tolerance.
  - a. Nominal vertical scan line: 0°
  - b. Total tolerance = 6.5°, includes:
    - i. Pointing error:  $\pm 3^\circ$  Typ.
    - ii. Pointing shift after 2000G shock:  $\pm 0.5^\circ$  Typ.
4. Maximum envelope does not include integration tolerances.
5. For increased working range at 10,000 FCD, position opaque material to block ambient light from entering the zone labeled "Direct Field of View of Photo Detector."
6. The SE955 scan engine does not require margin on either side of the bar code to decode. The 47° scan line provides identical scanning performance to older scan engines with a scan line of 53°.
7. Untoleranced dimensions:  
 .x:  $\pm .5$  mm; .xx:  $\pm .25$  mm; .xxx:  $\pm .125$  mm
8. This is a reference drawing and is not intended to specify or guarantee all possible integration requirements for this engine.

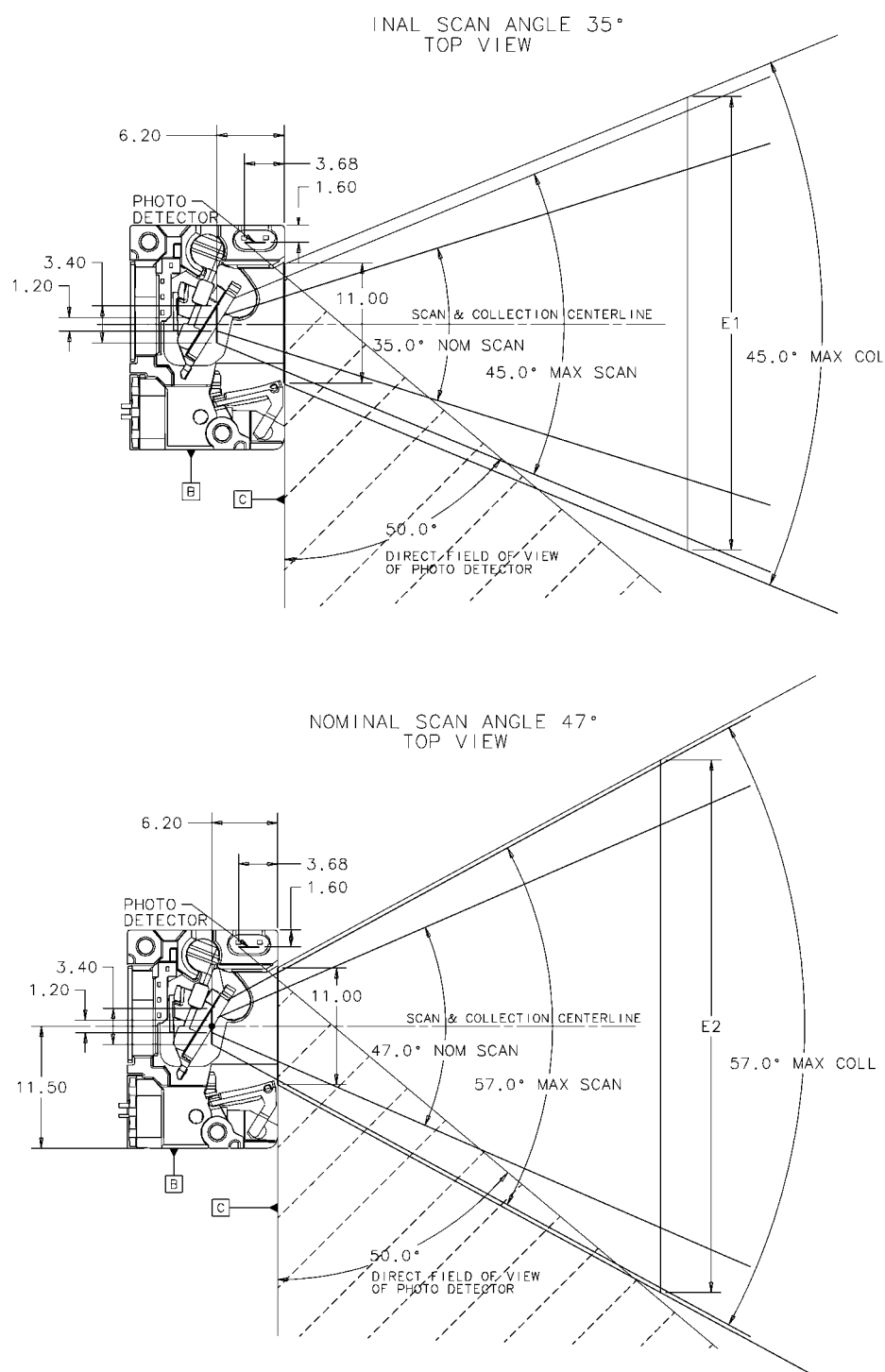


Figure 2-6. Exit Window Positioning

**Table 2-6. Exit Window Distance from Scan Engine: 0.15 in - 0.36 in (3.8 mm - 9 mm)**

[illegible]

**Table 2-7. Exit Window Distance from Scan Engine: 0.39 in - 2.00in (10 mm - 50.8 mm)**

[illegible]

## Accessories

### Flex Cables

A flex strip cable can be used to connect the SE955 scan engine to OEM equipment. [Figure 2-7](#) illustrates the 12-pin tapered flex strip cable (p/n 15-81378-01), [Figure 2-8](#) illustrates the 12-pin 53 mm even width flex strip cable (p/n 50-16000-139R), and [Figure 2-9](#) illustrates the 12-pin 245 mm even width flex strip cable (p/n 50-16000-134R). [Table 2-8](#) lists the available accessories for the scan engine, available from Symbol Technologies.

**Table 2-8. Symbol Accessories: Flex Strips and Adapter Plate**

Item	Symbol Part Number
Tapered 12-Pin Flex Strip	15-81378-01
Even Width 12 pin Straight Flex Strip - 10 in. (254 mm)	50-16000-134R
Even Width 12-Pin Straight Flex Strip - 2 in. (53 mm)	50-16000-139R
12-Pin Straight Flex - connectors on opposite sides	50-16000-308R
Universal (Scan Engine) Developer Kit	DKSE-1000-000R
Mounting Adapter Plate (for SE1200 conversion only)	KT-1200MB-01

### Connectors

[Table 2-10](#) lists the 12-pin ZIF connectors, with gold pin terminations, available in small quantities from Symbol Technologies.

**Table 2-9. Gold Connectors**

Engine	Manufacturer	Manufacturer Part Number	Description
SE955 (3.3V)	Molex	54548-1271	12-pin, horizontal ZIF connector, gold terminations
SE955 (5V)	Molex	52559-1252	12-pin, vertical ZIF connector, gold terminations

### Hardware Accessories

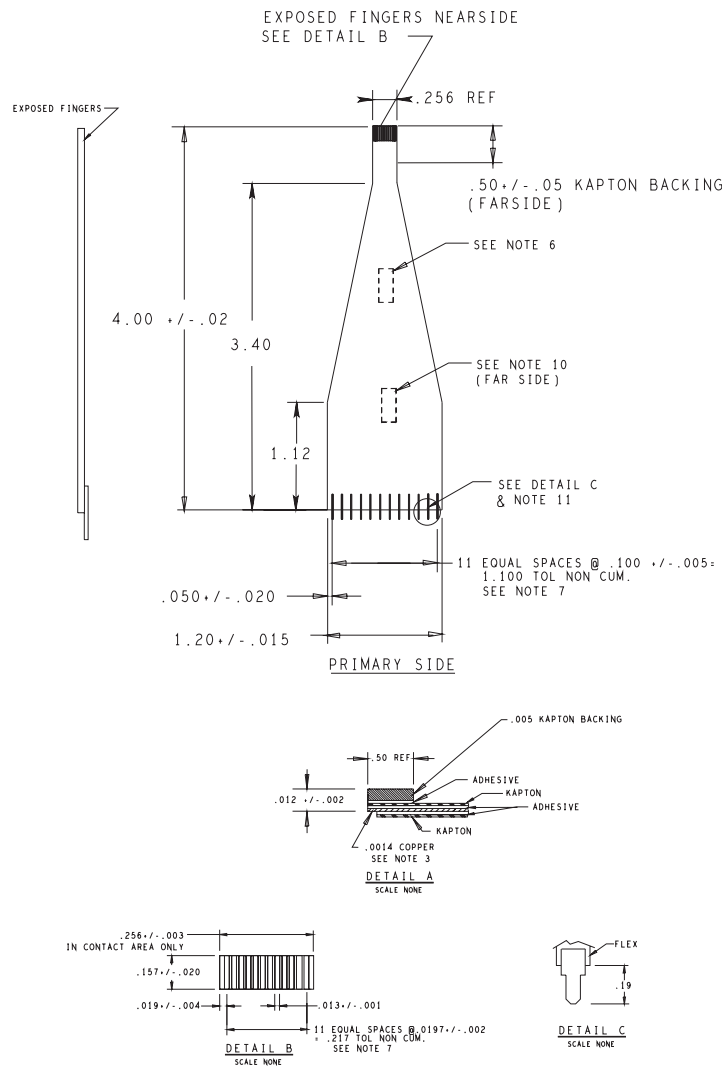
[Table 2-10](#) lists sources for hardware accessories for the scan engine.

**Table 2-10. Hardware Accessories**

Company	Discipline	Specifics
Tower Fasteners Inc. 1690 North Ocean Ave. Holtsville, New York 11742-1823 (631) 289-8800	Fasteners	Metallic, non-magnetic M1.6 x 0.35 machine screws. Length is integration dependent. However, a minimum of 5 threads is recommended.
AXON' Cable Inc. 1314 Plum Grove Road Schaumburg, IL 60173 (847) 230-7800	Flex Cables	Ensure flex mates with Molex 54548-1271.

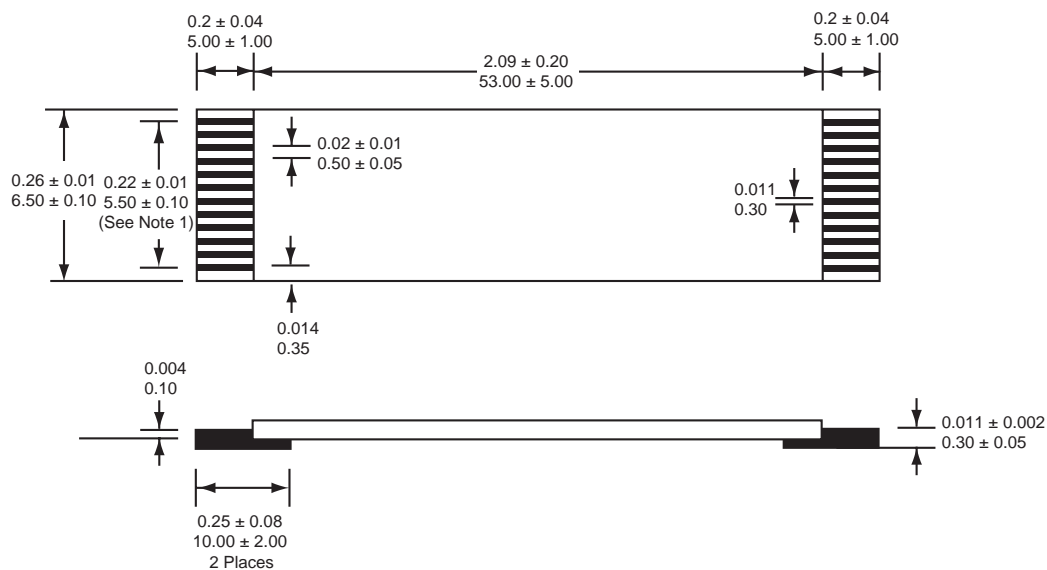
## Tapered 12-Pin Flex Strip

The 12-pin to 12-pin flex strip (p/n 15-81378-01), may be used only for evaluation purposes and not for production units (see Figure 2-7)



**Figure 2-7. Flex Strip, p/n 15-81378-01 (Tapered)**

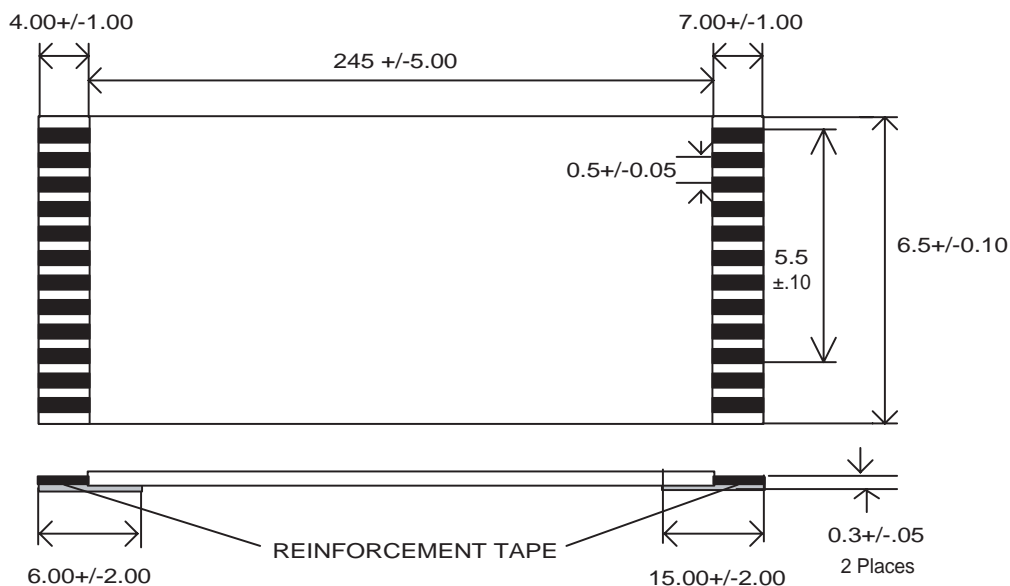




Note:

1. Dimension of conductor is center to center.
2. Dimensions are in:  
inches  
mm

**Figure 2-8. Flex Strip, p/n 50-16000-139R (Even Width, 53 mm)**



**Figure 2-9. Flex Strip, p/n 50-16000-134R (Even Width, 245 mm)**

## **Scan Engine Developer Kit**

The Scan Engine Developer Kit (p/n DKSE-1000-000R) enables development of products and systems around the SE955 using the Windows 98, 2000, or XP platform. The kit provides the software and hardware tools required to design and test the embedded scan engine application before integration into the host device.

The kit allows you to use Symbol's Simple Serial Interface (SSI) protocol to design bar code scanning applications, and contains an SSI ActiveX component to simplify the scan engine application.

The Scan Engine Developer Kit contains:

- CD, which includes:
  - Simple Serial Interface header files
  - Windows Serial Communication Library and source code
  - Simple Serial Interface Library and source code
  - Dynamic Link Library (DLL) with source code
  - ActiveX component
  - Windows demo programs and source code
  - Simple Serial Interface Developer Guide
  - Library documentation
- Developer board for connecting the scan engine to the PC development workstation. Functions of the development board include:
  - Mounting location for scan engine
  - Beeper and LED drivers
  - 9-pin RS232 for connection to PC workstation
  - Aim and trigger buttons
  - Beeper
  - Power, Decode, Low Power Mode LEDs
  - Test points
- Flex strips
- Interface cables for connection between the development board and the PC workstation
- 5V universal power supply.

## Regulatory Requirements

Documentation and labeling requirements for Class 1 and Class 2 laser products are described in [Chapter 6, Regulatory Requirements](#).



# 3

## ***Replacing Existing Engines***

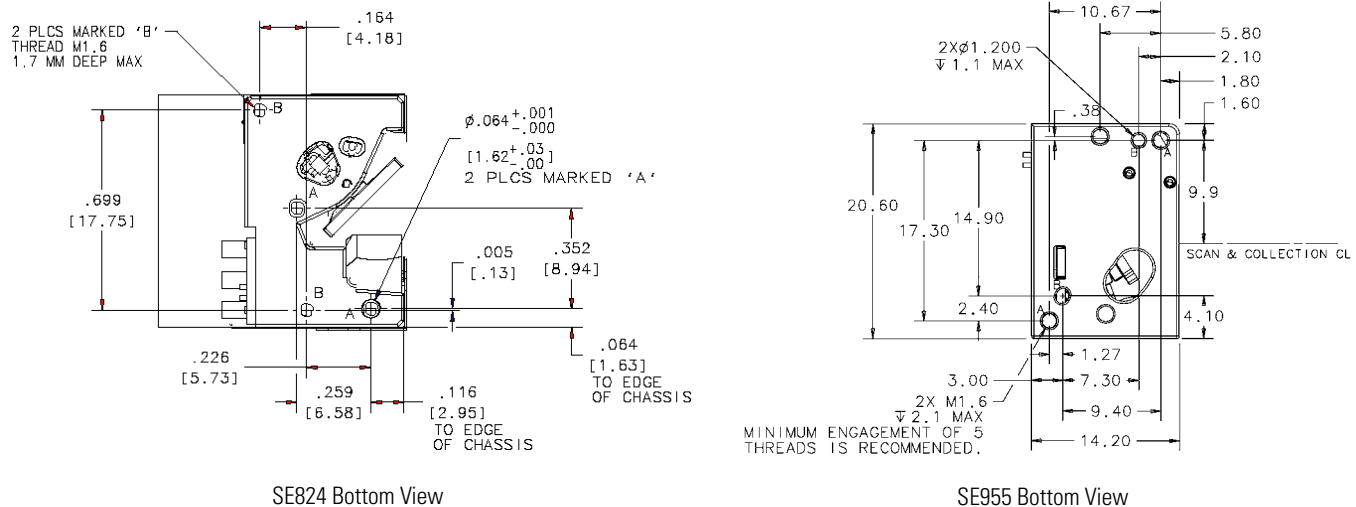
### **Introduction**

This chapter provides information for replacing an SE824, SE923 or SE1223WA scan engine with the SE955. Physical and electrical considerations are presented, together with recommended window properties.

## Replacing an SE824 with the SE955 Scan Engine

### Mounting

Figure 3-1 illustrates the mounting differences between the SE824 and SE955 scan engines. The SE955 can be used as a replacement for the SE824 scan engine, however, the mounting holes for the SE955 do not match those of the SE824. You must modify the mounting holes and locating pins on the host device.



**Figure 3-1. SE824 vs. SE955 Mounting Diagram**

### Electrical

The SE824 chassis is electrically connected to Vcc while the SE955 chassis is electrically connected to ground. The SE955 chassis must be isolated from the host Vcc and ground.

The SE824 scan engine operates at a Vcc of 3.3 VDC ( $\pm 10\%$ ) and the SE955-I100R/E100R scan engines operate at a Vcc of 3.0 VDC to 3.6 VDC. The SE955-I105R/E105R scan engines operate at a Vcc of 3.2 VDC to 5.5 VDC.

The SE955 incorporates SSI that allows configuration of the scan engine. The following SSI features are supported:

- changing scan angle between 35° and 47°
- selecting Aim or Scanstand modes
- Remote Scanner Management (RSM) reporting support
- reflash loading to upgrade firmware.

Refer to [Chapter 10, Simple Serial Interface](#) for detailed information on SSI (RSM) communication.

## **Optical**

When replacing an SE824 scan engine with the SE955 scan engine the following must be taken into consideration:

- Design of housing and scan engine must be reviewed by an optical-mechanical engineer.
- See [Table 2-6](#) and [Table 2-6 on page 2-12](#) to verify whether the exit window angle and size satisfy the recommended minimum requirement.
- Baffles designed for the SE824 may not be applicable for the SE955 due to the positioning of the photo-diode.
- The SE955 scan engine can be programmed to two different scan angles. It is recommended that the position of the scan engine is set using the widest scan angle (47°). Setting the position for the narrow scan angle (35°) and then changing the scan angle might cause clipping (internal reflection) of the laser beam against the housing.
- The SE955 scan engine does not require margin on either side of the bar code to decode. The 47° scan line provides identical scanning performance to older scan engines with a scan line of 53°.
- See [Optical on page 2-4](#) for recommendations on window properties (material, color, spectral transmission, wavefront distortion, surface quality, coating, etc.). Window properties that satisfy the recommendations for the SE824 scan engine automatically satisfy the recommendations for SE955 scan engine.

## **Mechanical**

When replacing an SE824 scan engine with the SE955 scan engine the following must be taken into consideration:

- Regulatory labels must reflect new VLD power.
- Consider that existing cable flexes may not be compatible with the SE955 scan engine.
- Consider mounting holes.

## **Regulatory**

End user documentation and product labeling may need to be changed or updated See [Chapter 6, Regulatory Requirements](#) for more information.

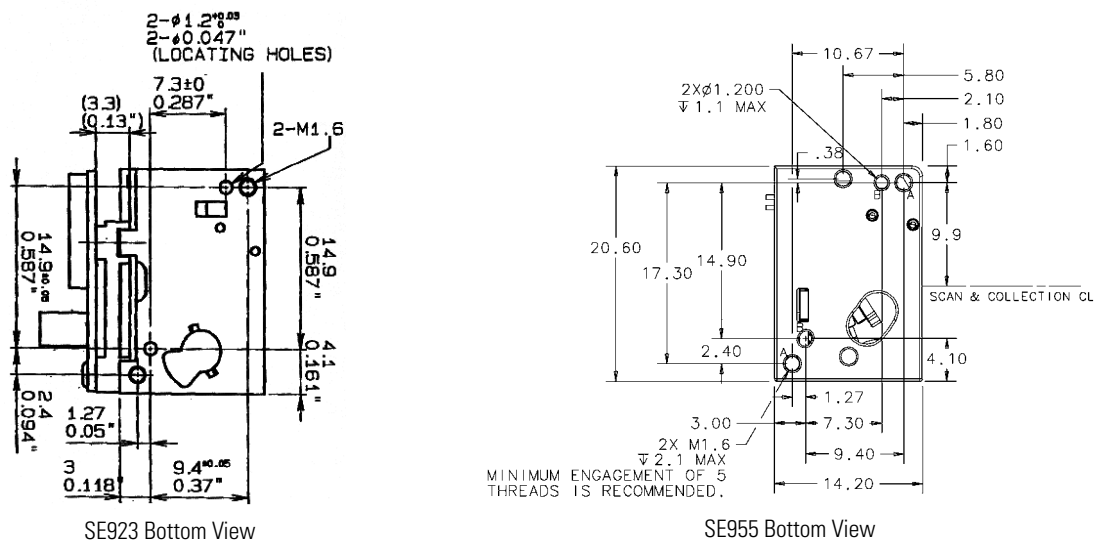
## Replacing an SE923 with the SE955 Scan Engine

The SE955 can be used as a replacement for the SE923 scan engine. The mounting holes for the SE955 match those of the SE923.

The SE923 scan engine chassis is electrically connected to Vcc while the SE955 scan engine chassis is electrically connected to ground and must be isolated from the host Vcc and ground.

### Mounting

Figure 3-1 illustrates the mounting differences between the SE923 and SE955. The SE955 can be used as a replacement for the SE923 scan engine because the mounting holes for the SE955 exactly match those of the SE923. You do not have to modify the mounting holes and locating pins on the host device.



**Figure 3-2. SE923 vs. SE955 Mounting Diagram**

### Electrical

The SE923 chassis is electrically connected to Vcc while the SE955 chassis is electrically connected to ground. The SE955 chassis must be isolated from the host Vcc and ground.

The SE923 scan engine operates at a Vcc of 3.3 to 5.0 VDC ( $\pm 10\%$ ). The SE955-I100R/E100R scan engines operate at a Vcc of 3.0 VDC to 3.6 VDC. The SE955-I105R/E105R scan engines operate at a Vcc of 3.2 VDC to 5.5 VDC.

The SE955 incorporates SSI that allows configuration of the scan engine. The following SSI features are supported:

- changing scan angle between 35° and 47°
- selecting Aim or Scanstand modes
- RSM reporting support
- reflash loading to upgrade firmware.

Refer to [Chapter 10, Simple Serial Interface](#) for detailed information for SSI (RSM) communication.



## **Optical**

When replacing an SE923 scan engine with the SE955 scan engine the following must be taken into consideration:

- Design of housing and scan engine must be reviewed by an optical-mechanical engineer.
- See [Table 2-6](#) and [Table 2-6 on page 2-12](#) to verify whether the exit window angle and size satisfy the recommended minimum requirement.
- Baffles designed for the SE923 may not be applicable for the SE955 due to the positioning of the photo-diode.
- The SE955 can be programmed to two different scan angles. It is recommended that the position of the scan engine is set using the widest scan angle (47°). Setting the position for the narrow scan angle (35°) and then changing the scan angle might cause clipping (internal reflection) of the laser beam against the housing.
- The SE955 scan engine does not require margin on either side of the bar code to decode. The 47° scan line provides identical scanning performance to older scan engines with a scan line of 53°.
- See [Optical on page 2-4](#) for recommendations on window properties (material, color, spectral transmission, wavefront distortion, surface quality, coating, etc.). Window properties that satisfy the recommendations for the SE923 scan engine automatically satisfy the recommendations for SE955 scan engine.

## **Mechanical**

When replacing an SE923 scan engine with the SE955 scan engine the following must be taken into consideration:

- Regulatory labels must reflect new VLD power.
- Consider that existing cable flexes may not be compatible with the SE955 scan engine.
- Consider mounting holes.

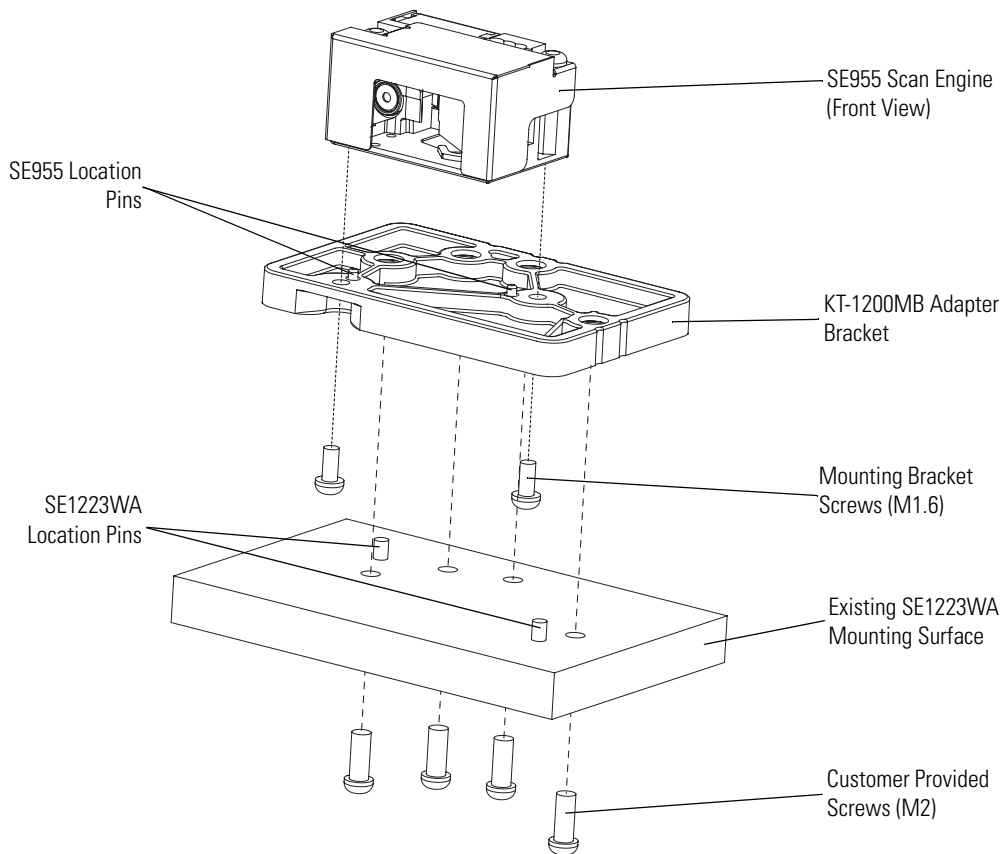
## **Regulatory**

End user documentation and product labeling may need to be changed or updated See [Chapter 6, Regulatory Requirements](#) for more information.

## Replacing an SE1223WA with the SE955 Scan Engine

### Mounting

The SE955 can be used as a replacement for the SE1223WA scan engine. However, the mounting holes for the SE955 do not match those of the SE1223WA. In order to mount the SE955 in place of an SE1223WA, use adapter bracket, KT-1200MB-01, to mount the SE955.



**Figure 3-3. SE1223WA Adapter Bracket**



An extended flex cable is required to compensate for the adapter bracket when connecting the SE955 scan engine to your host interface.

To mount the SE955 scan engine and adapter bracket to an existing SE1223WA housing:

1. Align the locations pins on the adapter bracket with the holes on the SE955.
2. Secure the adapter bracket to the SE955 using the two screw provided.
3. Align the scan engine and adapter bracket with the location pins on the housing.
4. Secure the scan engine and adapter bracket with the housing using customer provided screws.

## Electrical

The SE1223WA chassis is electrically connected to Vcc while the SE955 chassis is electrically connect to ground. The SE955 must be isolated from the host Vcc and ground.

The SE1223WA scan engine operates at a Vcc of 5.0 VDC ( $\pm 10\%$ ). The SE955-I100R/E100R scan engines operate at a Vcc of 3.0 VDC to 3.6 VDC. The SE955-I105R/E105R scan engines operate at a Vcc of 3.2 VDC to 5.5 VDC.

The SE955 incorporates SSI that allows configuration of the scan engine. The following SSI features are supported:

- changing scan angle between 35° and 47°
- selecting Aim or Scanstand modes
- RSM reporting support
- reflash loading to upgrade firmware.

Refer to [Chapter 10, Simple Serial Interface](#) for detailed information for SSI (RSM) communication.

## Optical

When replacing an SE1223WA scan engine with the SE955 scan engine the following must be taken into consideration:

- Design of housing and scan engine must be reviewed by an optical-mechanical engineer.
- See [Table 2-6](#) and [Table 2-6 on page 2-12](#) to verify whether the exit window angle and size satisfy the recommended minimum requirement.
- Baffles designed for the SE1223WA may not be applicable for the SE955 due to the positioning of the photo-diode.
- The SE955 can be programmed to two different scan angles. It is recommended that the position of the scan engine is set using the widest scan angle (47°). Setting the position for the narrow scan angle (35°) and then changing the scan angle might cause clipping (internal reflection) of the laser beam against the housing.
- The SE955 scan engine does not require margin on either side of the bar code to decode. The 47° scan line provides identical scanning performance to older scan engines with a scan line of 53°.
- See [Optical on page 2-4](#) for recommendations on window properties (material, color, spectral transmission, wavefront distortion, surface quality, coating, etc.). Window properties that satisfy the recommendations for the SE1200 scan engine automatically satisfy the recommendations for SE955 scan engine.

## Mechanical

When replacing an SE1223WA scan engine with the SE955 scan engine the following must be taken into consideration:

- Regulatory labels must reflect new VLD power.
- Consider that existing cable flexes may not be compatible with the SE955 scan engine.
- Consider mounting holes.

## Regulatory

End user documentation and product labeling may need to be changed or updated See [Chapter 6, Regulatory Requirements](#) for more information.



## ***SE955-I100R/E100R Specifications***

### **Introduction**

This chapter provides the technical specifications of the SE955-I100R and SE955-E100R scan engines. Decode zone and exit window characteristics are also presented.

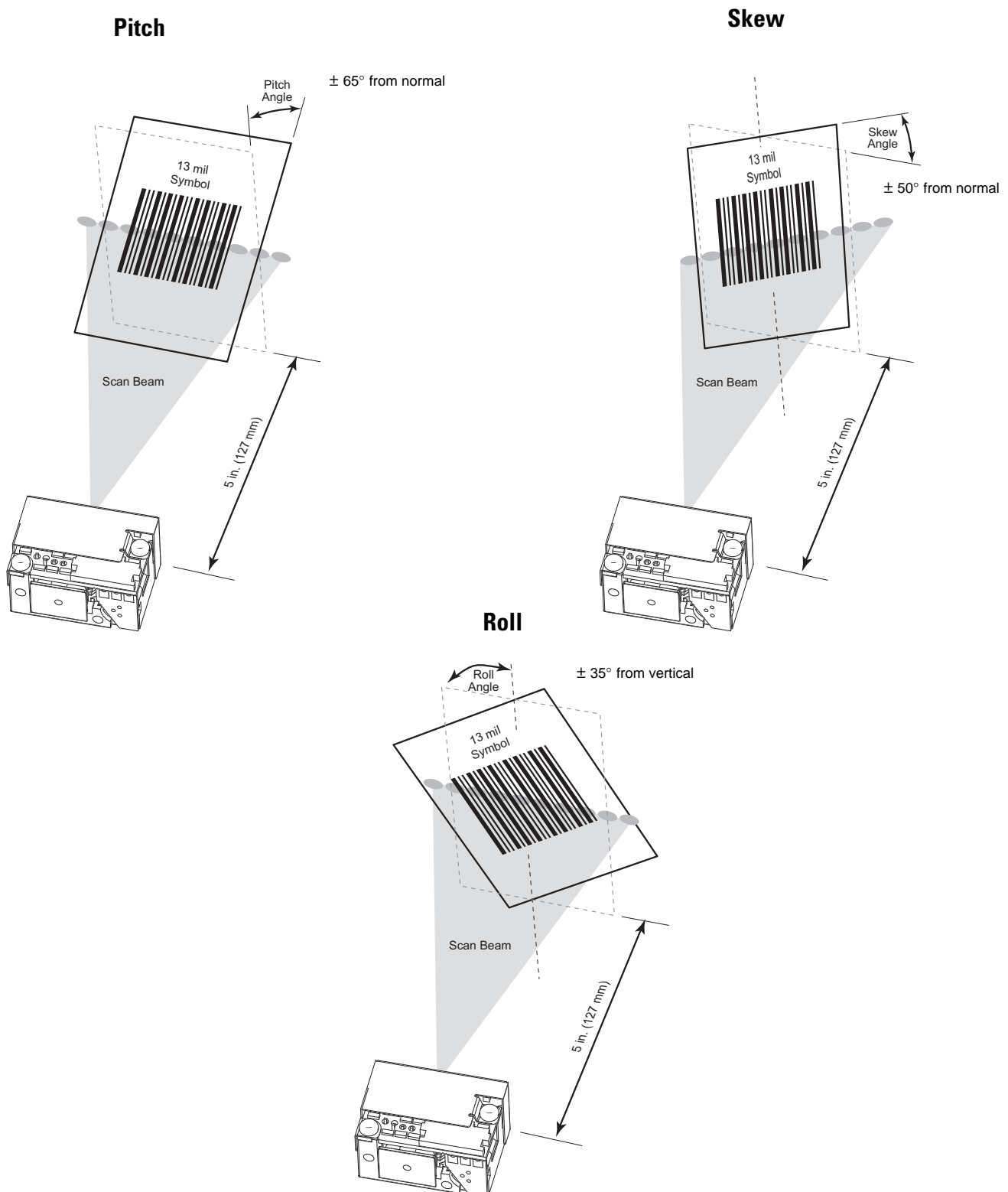
## Technical Specifications

**Table 4-1. Technical Specifications @ 23°C**

Item	Description
<b>Power Requirements</b>	
<b>Input Voltage</b>	3.0 VDC to 3.6 VDC
<b>Scanning Current</b>	86 mA typical / 114 mA max.
<b>Continuous Mode Current (Laser not on)</b>	45 mA typical / 56 mA max.
<b>Standby Current</b>	12µA typical / 60 µA max
<b>V<sub>cc</sub> Noise Level</b>	100 mV peak to peak max.
<b>Surge Current</b>	200 mA typical (depending on power supply rise time)
<b>Scan Repetition Rate</b>	92 min., 104 typical, 116 max. scans/sec (bidirectional)
<b>Laser Power (at 650 nm)</b>	<b>SE955-I100R:</b> 1.7 mW (± 0.2 mW) scan mode; 0.5 mW (± 0.05 mW) aim mode <b>SE955-E100R:</b> 0.7 mW (± 0.1 mW) scan mode; aim mode not supported
<b>Optical Resolution</b>	0.004 in. minimum element width
<b>Print Contrast</b>	minimum 25% absolute dark/light reflectance measured at 650 nm.
<b>Scan Angle</b> <b>Wide (Default)</b> <b>Narrow</b>	47° ± 3° 35° ± 3° Note: The SE955 scan engine does not require margin on either side of the bar code to decode. The 47° scan line provides identical scanning performance to older scan engines with a scan line of 53°.
<b>Decode Depth of Field</b>	See decode zone diagrams beginning on <a href="#">page 4-5</a> .
<b>Pitch Angle</b>	Condition: 100% UPC at 5 in. ± 65° from normal (see <a href="#">Figure 4-1 on page 4-4</a> )
<b>Skew Tolerance</b>	Condition: 100% UPC at 5 in. ± 50° from normal (see <a href="#">Figure 4-1 on page 4-4</a> )
<b>Roll</b>	Condition: 100% UPC at 5 in. ± 35° from vertical (see <a href="#">Figure 4-1 on page 4-4</a> )
<b>Ambient Light Immunity</b> <b>Sunlight</b> <b>Artificial Light</b>	10,000 ft. candles (107,640 lux) 450 ft. candles (4,844 lux) Note: 10,000 ft. candles ambient light immunity is guaranteed when the engine is mounted using the illustrations and notes shown in <a href="#">Figure 2-6 on page 2-11</a> . If a different mounting configuration is used, then ambient light immunity may be reduced.
<b>Shock Endurance</b>	2,000G applied via any mounting surface from -4° F to 140° F (-20° C to 60° C) for a period of 0.85 msec.

**Table 4-1. Technical Specifications @ 23°C (Continued)**

Item	Description
<b>Vibration</b>	Unpowered engine withstands a random vibration along each of the X, Y and Z axes for a period of one hour per axis, define as follows: 20 to 80 Hz                      Ramp up to 0.04 G <sup>2</sup> /Hz at the rate of 3 dB/octave. 80 to 350 Hz                     0.04 G <sup>2</sup> /Hz 350 to 2000 Hz                 Ramp down at the rate of 3 dB/octave.
<b>Laser Class</b>	<b>SE955-I100R:</b> The scan engine, by itself, is a classified component. It is intended for use in CDRH Class II/IEC Class 2 devices with proper housing, labeling, and instructions to comply with federal and/or international standards.  <b>SE955-E100R:</b> The scan engine, by itself, is a classified component. It is intended for use in IEC Class 1 devices with proper housing, labeling, and instructions to comply with federal and/or international standards.
<b>RoHS</b>	Meets RoHS requirements
<b>ESD Protection (IEC 61000-4-2)</b>	±2kV Contact pin direct discharge, ±8kV indirect discharge
<b>RF Immunity (IEC 61000-4-3)</b>	10V/m
<b>Emissions</b>	FCC Part 15 Class B, ICES-003 Class B, CISPR Class B, Japan VCCI Class B
<b>Laser Safety</b>	<b>SE955-I100R:</b> IEC60825-1 Class 2 <b>SE955-E100R:</b> IEC60825-1 Class 1
<b>Operating Temperature (chassis)</b>	-4° F to 140° F (-20° C to 60° C)
<b>Storage Temperature</b>	-40°F to 158° F (-40° C to 70° C)
<b>Humidity</b>	5% to 95% (non-condensing)
<b>Height</b>	0.465 in (11.8mm) maximum
<b>Width</b>	0.85 in (21.6mm) maximum
<b>Depth of Chassis</b>	0.60 in (15.2mm) maximum
<b>Depth of Chassis (total)</b>	0.61 in (15.5mm) maximum
<b>Weight</b>	0.265 ounces ± 0.009 ounces (7.5 grams ± 0.25 grams)

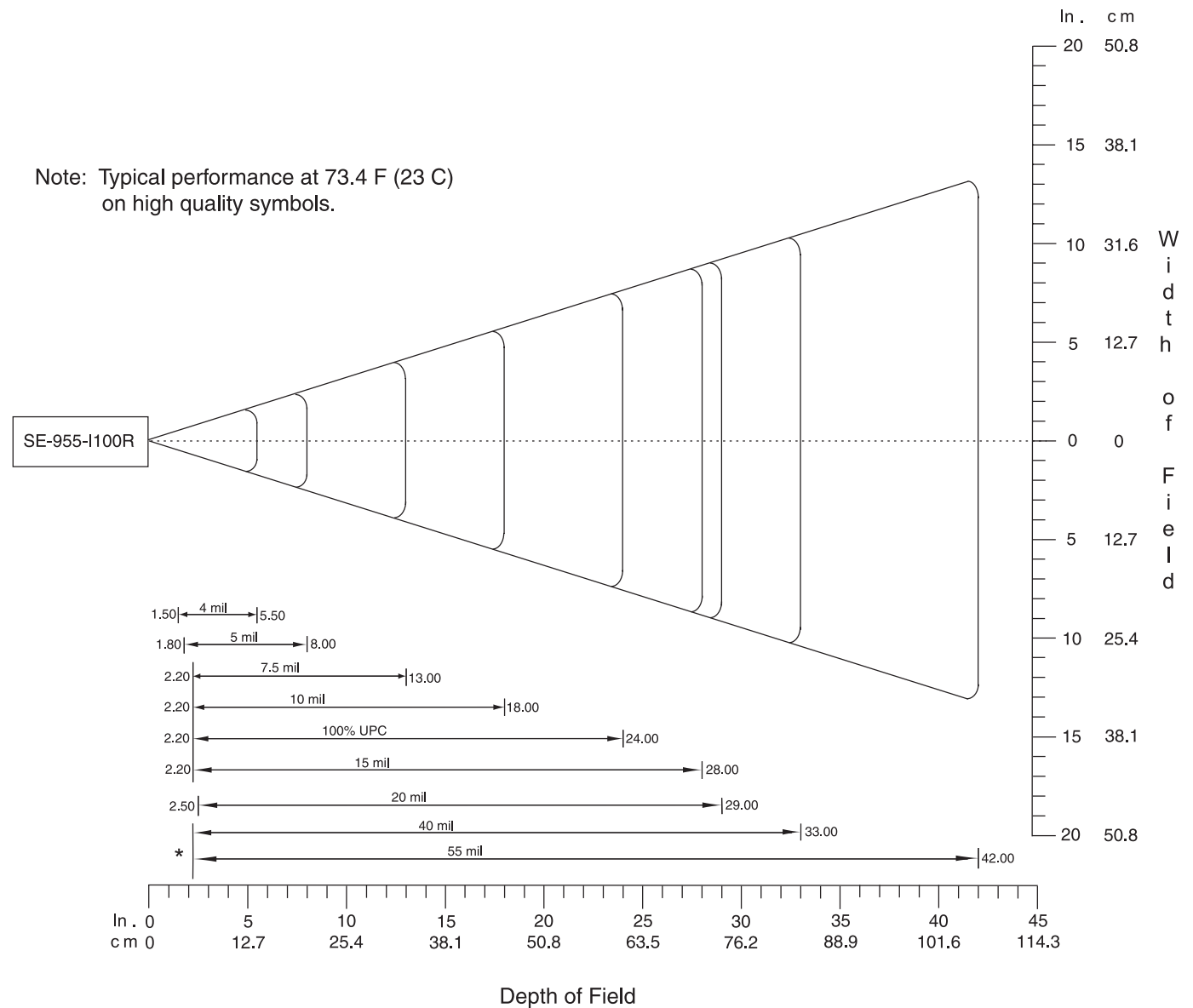


**Figure 4-1. Pitch, Skew and Roll**



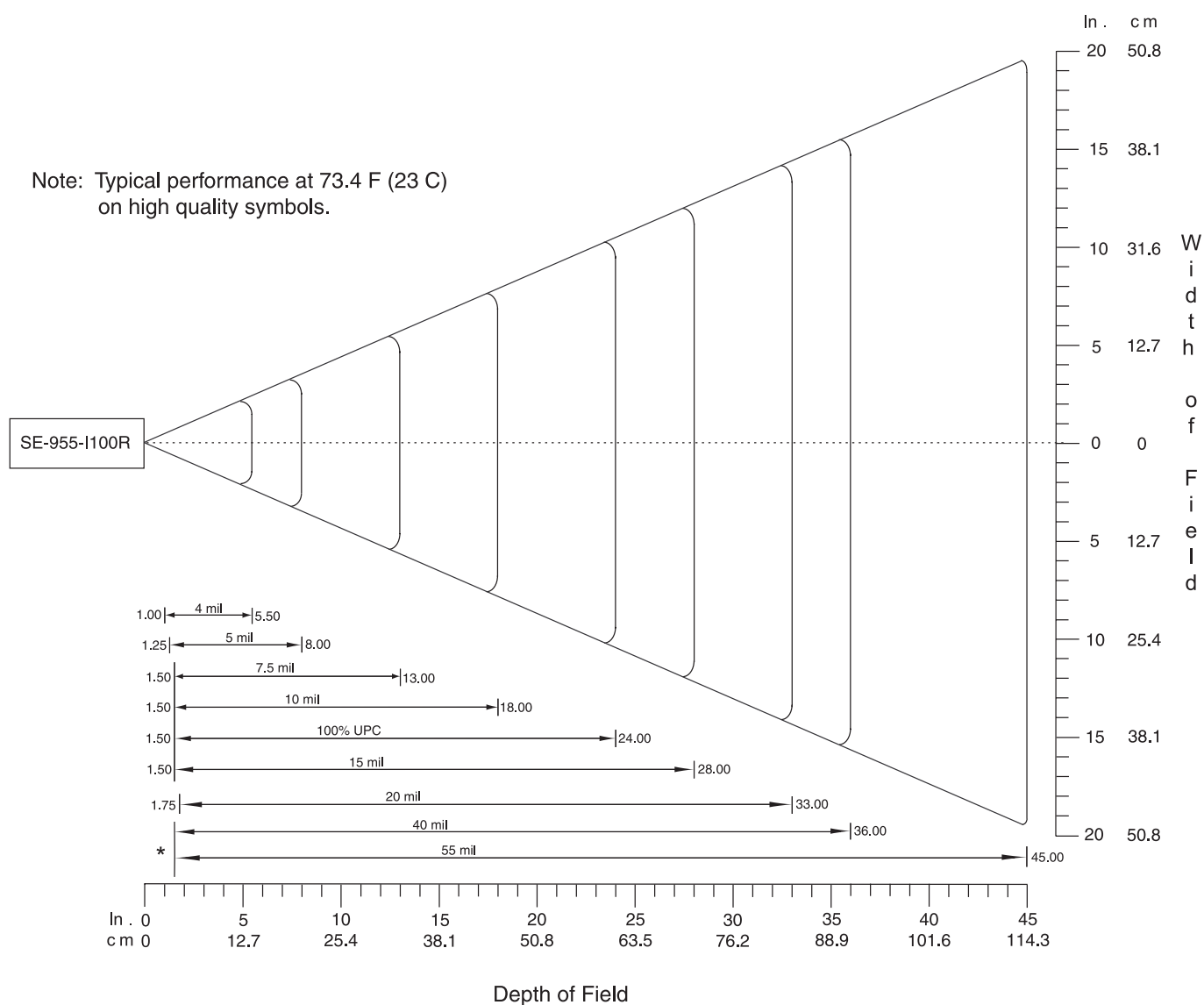
## Decode Zones

The decode zones for the SE955-I100R scan engines are shown in Figure 4-2 and Figure 4-3. The decode zones for the SE955-E100R scan engines are shown in Figure 4-4 through Figure 4-5. The figures shown are typical values. Table 4-2 and Table 4-3 lists the typical and guaranteed distances for selected bar code densities. The minimum element width (or "symbol density") is the width in mils of the narrowest element (bar or space) in the symbol.



\*Minimum distance determined by symbol length and scan angle

**Figure 4-2. SE955-I100R Standard Version 35° Decode Zone**



**Figure 4-3. SE955-I100R Standard Version 47° Decode Zone**

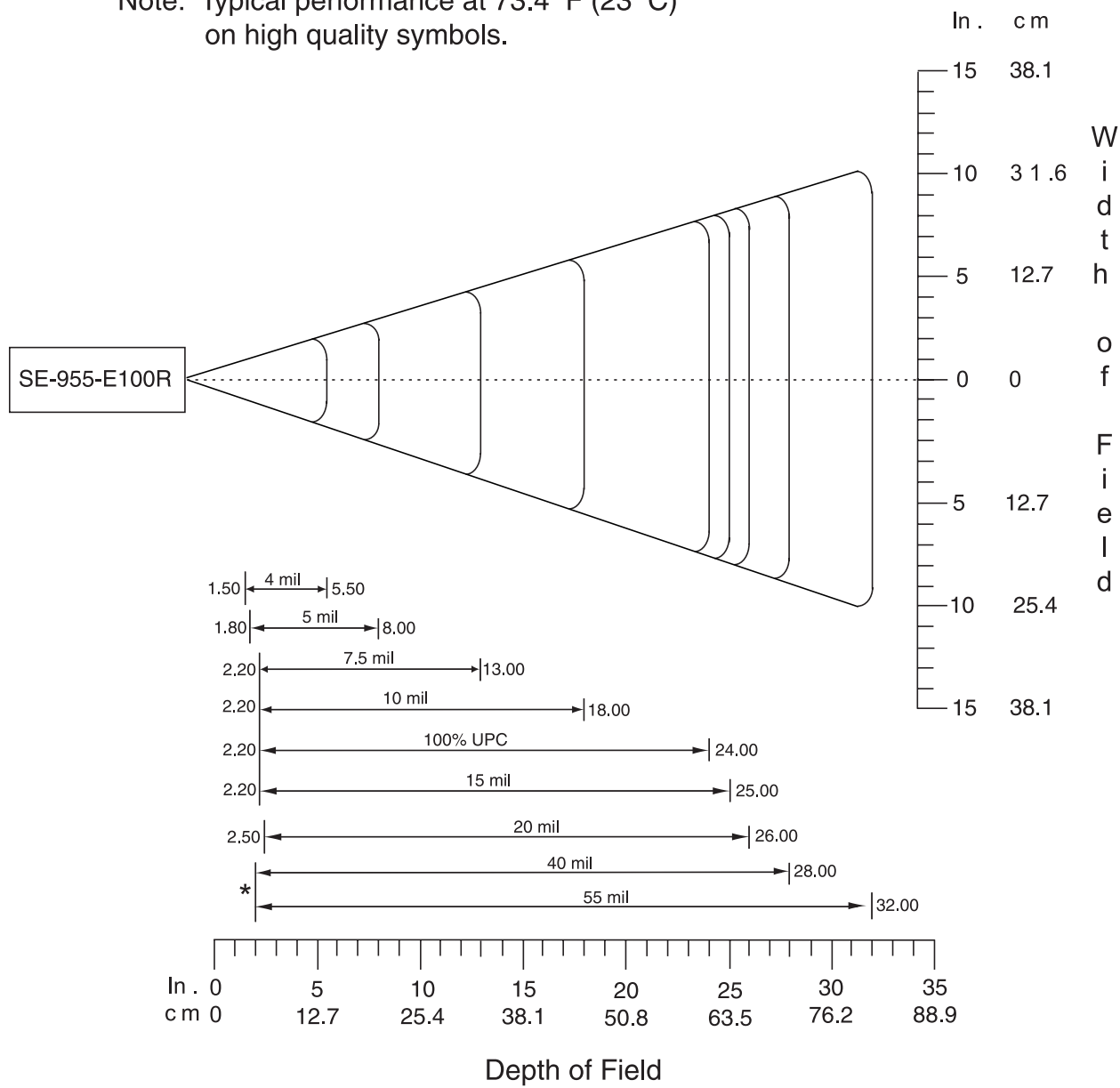
### Table 4-2. SE955-I100R Decode Distances

Symbol Density/ Bar Code Type/ W-N Ratio	Bar Code Content/ Contrast <sup>Note 1</sup>	35 ° Typical Working Ranges		35 ° Guaranteed Working Ranges		47 ° Typical Working Ranges		47 ° Guaranteed Working Ranges	
		Near	Far	Near	Far	Near	Far	Near	Far
<b>4.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.50 in</b> 3.81 cm	<b>5.50 in</b> 13.97 cm	<b>X</b>	<b>3.20 in</b> 8.13 cm	<b>1.00 in</b> 2.50 cm	<b>5.50 in</b> 13.97 cm	<b>2.20 in</b> 5.60 cm	<b>3.20 in</b> 8.13 cm
<b>5.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.80 in</b> 4.57 cm	<b>8.00 in</b> 20.32 cm	<b>X</b>	<b>5.50 in</b> 13.97 cm	<b>1.25 in</b> 3.18 cm	<b>8.00 in</b> 20.32 cm	<b>2.20 in</b> 5.60 cm	<b>5.50 in</b> 13.97 cm
<b>7.5 mil</b> Code 39; 2.5:1	<b>ABCDEF</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>13.00 in</b> 33.02 cm	<b>X</b>	<b>9.50 in</b> 24.13 cm	<b>1.50 in</b> 3.81 cm	<b>13.00 in</b> 33.02 cm	<b>2.00 in</b> 5.08 cm	<b>9.50 in</b> 24.13 cm
<b>10 mil</b> Code 39; 2.5:1	<b>ABCDE</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>18.00 in</b> 45.72 cm	<b>X</b>	<b>14.00 in</b> 35.56 cm	<b>1.50 in</b> 3.81 cm	<b>18.00 in</b> 45.72 cm	<b>1.75 in</b> 4.45 cm	<b>14.00 in</b> 35.56 cm
<b>13 mil</b> 100% UPC	<b>12345678905</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>24.00 in</b> 60.96 cm	<b>2.50 in</b> 6.35 cm	<b>18.00 in</b> 45.72 cm	<b>1.50 in</b> 3.81 cm	<b>24.00 in</b> 60.96 cm	<b>1.75 in</b> 4.45 cm	<b>18.00 in</b> 45.72 cm
<b>15 mil</b> Code 39; 2.5:1	<b>ABCD</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>28.00 in</b> 71.12 cm	<b>X</b>	<b>21.00 in</b> 53.34 cm	<b>1.50 in</b> 3.81 cm	<b>28.00 in</b> 71.12 cm	<b>1.75 in</b> 4.45 cm	<b>21.00 in</b> 53.34 cm
<b>20 mil</b> Code 39; 2.2:1	<b>123</b> 80% MRD	<b>2.50 in</b> 6.35 cm	<b>29.00 in</b> 73.66 cm	<b>X</b>	<b>24.00 in</b> 60.96 cm	<b>1.75 in</b> 4.45 cm	<b>33.00 in</b> 83.82 cm	<b>X</b>	<b>27.00 in</b> 68.58 cm
<b>40 mil</b> Code 39; 2.2:1	<b>AB</b> 80% MRD	<b>X</b>	<b>33.00 in</b> 83.82 cm	<b>X</b>	<b>25.00 in</b> 63.50cm	<b>X</b>	<b>36.00 in</b> 91.44 cm	<b>X</b>	<b>28.00 in</b> 71.12 cm
<b>55 mil</b> Code 39; 2.2:1	<b>CD</b> 80% MRD	<b>X</b>	<b>42.00 in</b> 106.68 cm	<b>X</b>	<b>30.00 in</b> 76.20 cm	<b>X</b>	<b>45.00 in</b> 114.30 cm	<b>X</b>	<b>34.00 in</b> 86.36 cm

Notes:

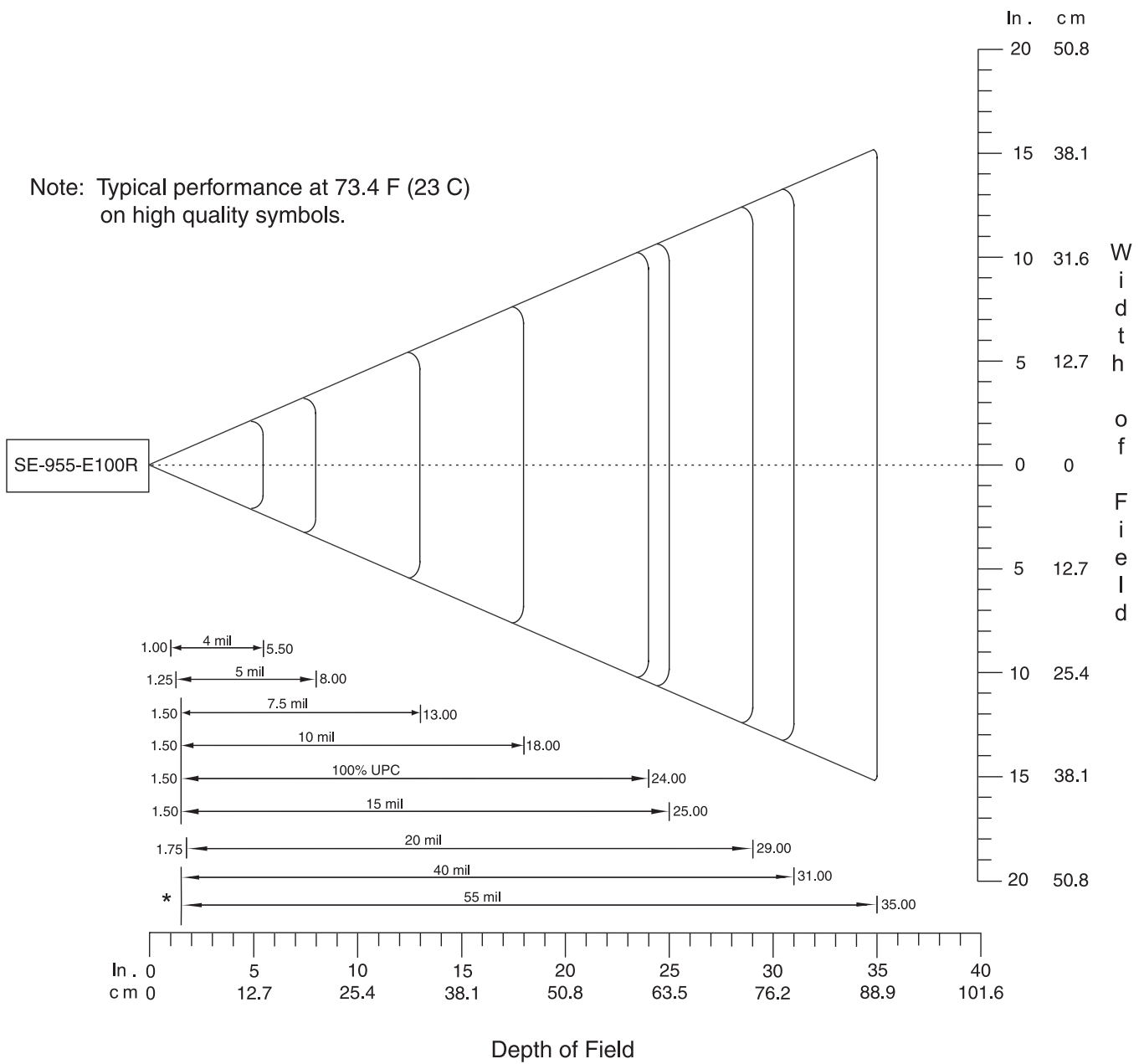
1. CONTRAST measured as Mean Reflective Difference (MRD) at 650 nm.
2. Near ranges on lower densities (not specified) are largely dependent upon the width of the bar code and the scan angle.
3. Working range specifications at ambient temperature (23°C), Photographic quality symbols. pitch=10°, roll=0°, skew=0°, ambient light < 150 ft-candles.
4. X - Dependent on width of bar code.
5. Distances measured from front edge of chassis.

Note: Typical performance at 73.4° F (23° C)  
on high quality symbols.



**Figure 4-4. SE955-E100R Standard Version 35° Decode Zone**

Note: Typical performance at 73.4 F (23 C)  
on high quality symbols.



\*Minimum distance determined by symbol length and scan angle

**Figure 4-5. SE955-E100R Standard Version 47° Decode Zone**

### Table 4-3. SE955-E100R Decode Distances

Symbol Density/ Bar Code Type/ W-N Ratio	Bar Code Content/ Contrast <sup>Note 1</sup>	35 ° Typical Working Ranges		35 ° Guaranteed Working Ranges		47 ° Typical Working Ranges		47 ° Guaranteed Working Ranges	
		Near	Far	Near	Far	Near	Far	Near	Far
<b>4.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.50 in</b> 3.81 cm	<b>5.50 in</b> 13.97 cm	<b>X</b>	<b>3.20 in</b> 8.13 cm	<b>1.00 in</b> 2.50 cm	<b>5.50 in</b> 13.97 cm	<b>2.20 in</b> 5.59 cm	<b>3.20 in</b> 8.13 cm
<b>5.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.80 in</b> 4.57 cm	<b>8.00 in</b> 20.32 cm	<b>X</b>	<b>5.50 in</b> 13.97 cm	<b>1.25 in</b> 3.18 cm	<b>8.00 in</b> 20.32 cm	<b>2.20 in</b> 5.59 cm	<b>5.50 in</b> 13.97 cm
<b>7.5 mil</b> Code 39; 2.5:1	<b>ABCDEF</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>13.00 in</b> 33.02 cm	<b>X</b>	<b>9.00 in</b> 22.86 cm	<b>1.50 in</b> 3.81 cm	<b>13.00 in</b> 33.02 cm	<b>2.00 in</b> 5.08 cm	<b>9.00 in</b> 22.86 cm
<b>10 mil</b> Code 39; 2.5:1	<b>ABCDE</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>18.00 in</b> 45.72 cm	<b>X</b>	<b>13.00 in</b> 33.02 cm	<b>1.50 in</b> 3.81cm	<b>18.00 in</b> 45.72 cm	<b>1.75 in</b> 4.45 cm	<b>13.00 in</b> 33.02 cm
<b>13 mil</b> 100% UPC	<b>12345678905</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>24.00 in</b> 60.96 cm	<b>2.50 in</b> 6.35 cm	<b>17.00 in</b> 43.18 cm	<b>1.50 in</b> 3.81 cm	<b>24.00 in</b> 60.96 cm	<b>1.75 in</b> 4.45 cm	<b>17.00 in</b> 43.18cm
<b>15 mil</b> Code 39; 2.5:1	<b>ABCD</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>25.00 in</b> 63.50 cm	<b>X</b>	<b>18.00 in</b> 45.72 cm	<b>1.50 in</b> 3.81 cm	<b>25.00 in</b> 63.50 cm	<b>1.75 in</b> 4.45 cm	<b>18.00 in</b> 45.72 cm
<b>20 mil</b> Code 39; 2.2:1	<b>123</b> 80% MRD	<b>2.50 in</b> 6.35 cm	<b>26.00 in</b> 66.04 cm	<b>X</b>	<b>18.00 in</b> 45.72 cm	<b>1.75 in</b> 4.45cm	<b>29.00 in</b> 73.66 cm	<b>X</b>	<b>19.00 in</b> 48.26 cm
<b>40 mil</b> Code 39; 2.2:1	<b>AB</b> 80% MRD	<b>X</b>	<b>28.00 in</b> 71.12 cm	<b>X</b>	<b>19.00 in</b> 48.26 cm	<b>X</b>	<b>31.00 in</b> 78.74 cm	<b>X</b>	<b>21.00 in</b> 53.34 cm
<b>55 mil</b> Code 39; 2.2:1	<b>CD</b> 80% MRD	<b>X</b>	<b>32.00 in</b> 81.28 cm	<b>X</b>	<b>23.00 in</b> 58.42 cm	<b>X</b>	<b>35.00 in</b> 88.90 cm	<b>X</b>	<b>25.00 in</b> 63.50 cm

Notes:

1. CONTRAST measured as Mean Reflective Difference (MRD) at 650 nm.
2. Near ranges on lower densities (not specified) are largely dependent upon the width of the bar code and the scan angle.
3. Working range specifications at ambient temperature (23°C), Photographic quality symbols. pitch=10°, roll=0°, skew=0°, ambient light < 150 ft-candles.
4. X - Dependent on width of bar code.
5. Distances measured from front edge of chassis.

# 5

## ***SE955-I105R/E105R Specifications***

### **Introduction**

This chapter provides the technical specifications and Decode Zones for the SE955-I105R and SE955-E105R scan engines.

## Technical Specifications

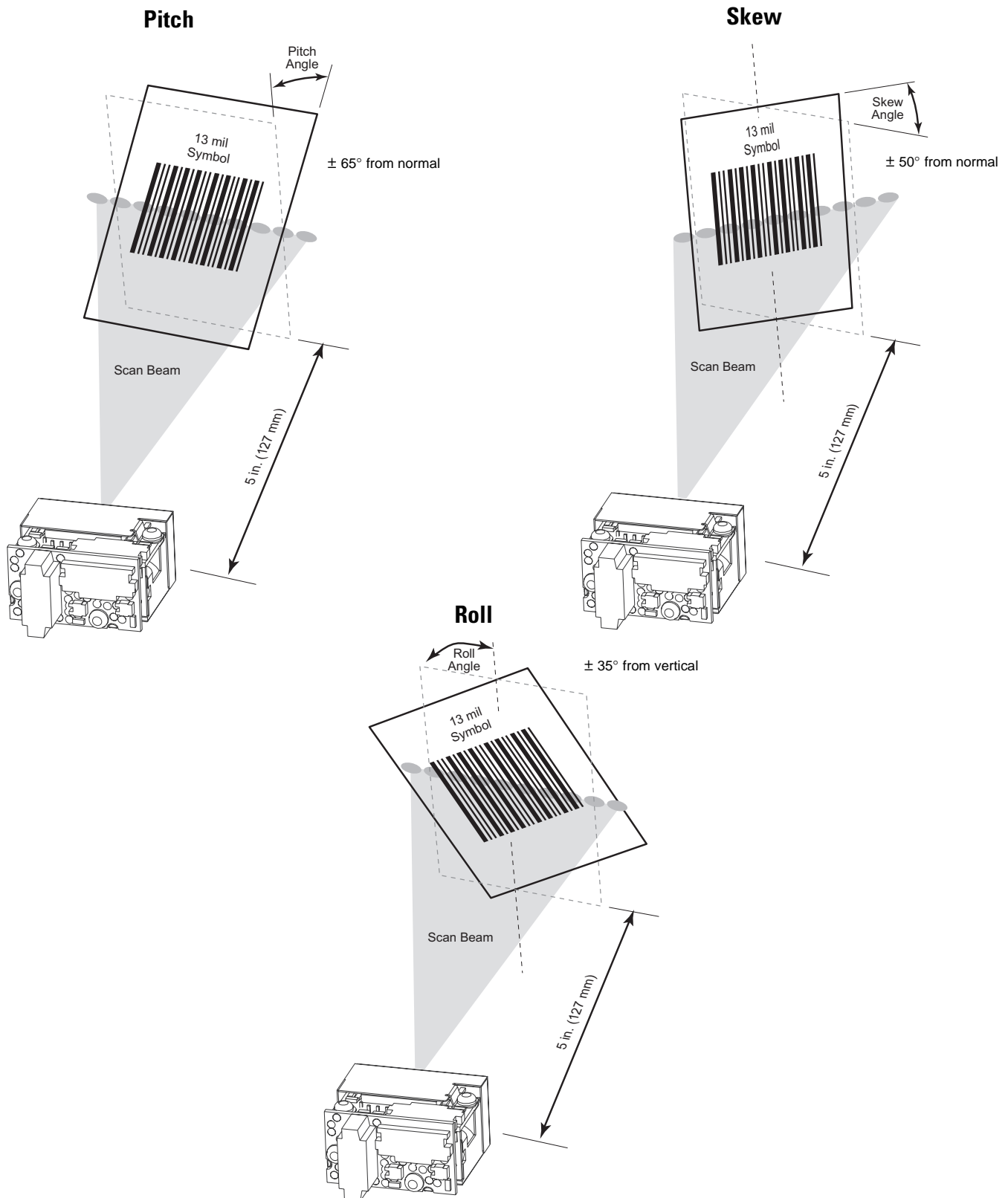
**Table 5-1. Technical Specifications @ 23°C**

Item	Description
<b>Power Requirements</b>	
<b>Input Voltage</b>	3.2 VDC - 5.5 VDC
<b>Scanning Current</b>	90 mA typical / 105 mA max.
<b>Continuous On Current (laser not on)</b>	45 mA typical / 50 mA max.
<b>Standby Current</b>	35 $\mu$ A typical / 210 $\mu$ A max
<b>V<sub>cc</sub> Noise Level</b>	100 mV peak to peak max.
<b>Surge Current</b>	200 mA typical (depending on power supply rise time)
<b>Scan Repetition Rate</b>	92 min., 104 typical, 116 max. scans/sec (bidirectional)
<b>Laser Power (at 650 nm)</b>	<b>SE955-I105R:</b> 1.7 mW ( $\pm$ 0.2 mW) scan mode; 0.5 mW ( $\pm$ 0.05 mW) aim mode <b>SE955-E105R:</b> 0.7 mW ( $\pm$ 0.1 mW) scan mode; aim mode not supported
<b>Optical Resolution</b>	0.004 in. minimum element width
<b>Print Contrast</b>	Minimum 25% absolute dark/light reflectance measured at 650 nm.
<b>Scan Angle</b>	
<b>Wide (Default)</b>	47° $\pm$ 3°
<b>Narrow</b>	35° $\pm$ 3°
	Note: The SE955 scan engine does not require margin on either side of the bar code to decode. The 47° scan line provides identical scanning performance to older scan engines with a scan line of 53°.
<b>Decode Depth of Field</b>	See decode zone diagrams beginning on <a href="#">page 5-5</a> .
<b>Pitch Angle</b>	Condition: 100% UPC at 5 in. $\pm$ 65° from normal (see <a href="#">Figure 5-1 on page 5-4</a> )
<b>Skew Tolerance</b>	Condition: 100% UPC at 5 in. $\pm$ 50° from normal (see <a href="#">Figure 5-1 on page 5-4</a> )
<b>Roll</b>	Condition: 100% UPC at 5 in. $\pm$ 35° from vertical (see <a href="#">Figure 5-1 on page 5-4</a> )
<b>Ambient Light Immunity</b>	
<b>Sunlight</b>	10,000 ft. candles (107,640 lux)
<b>Artificial Light</b>	450 ft. candles (4,844 lux)
	Note: 10,000 ft. candles ambient light immunity is guaranteed when the engine is mounted using the illustrations and notes shown in <a href="#">Figure 2-6 on page 2-11</a> . If a different mounting configuration is used, then ambient light immunity may be reduced.
<b>Shock Endurance</b>	2,000G applied via any mounting surface from -4° F to 140° F (-20° C to 60° C) for a period of 0.85 msec.
<b>Vibration</b>	Unpowered engine withstands a random vibration along each of the X, Y and Z axes for a period of one hour per axis, define as follows: 20 to 80 Hz                      Ramp up to 0.04 G <sup>2</sup> /Hz at the rate of 3 dB/octave. 80 to 350 Hz                    0.04 G <sup>2</sup> /Hz 350 to 2000 Hz                Ramp down at the rate of 3 dB/octave.



**Table 5-1. Technical Specifications @ 23°C (Continued)**

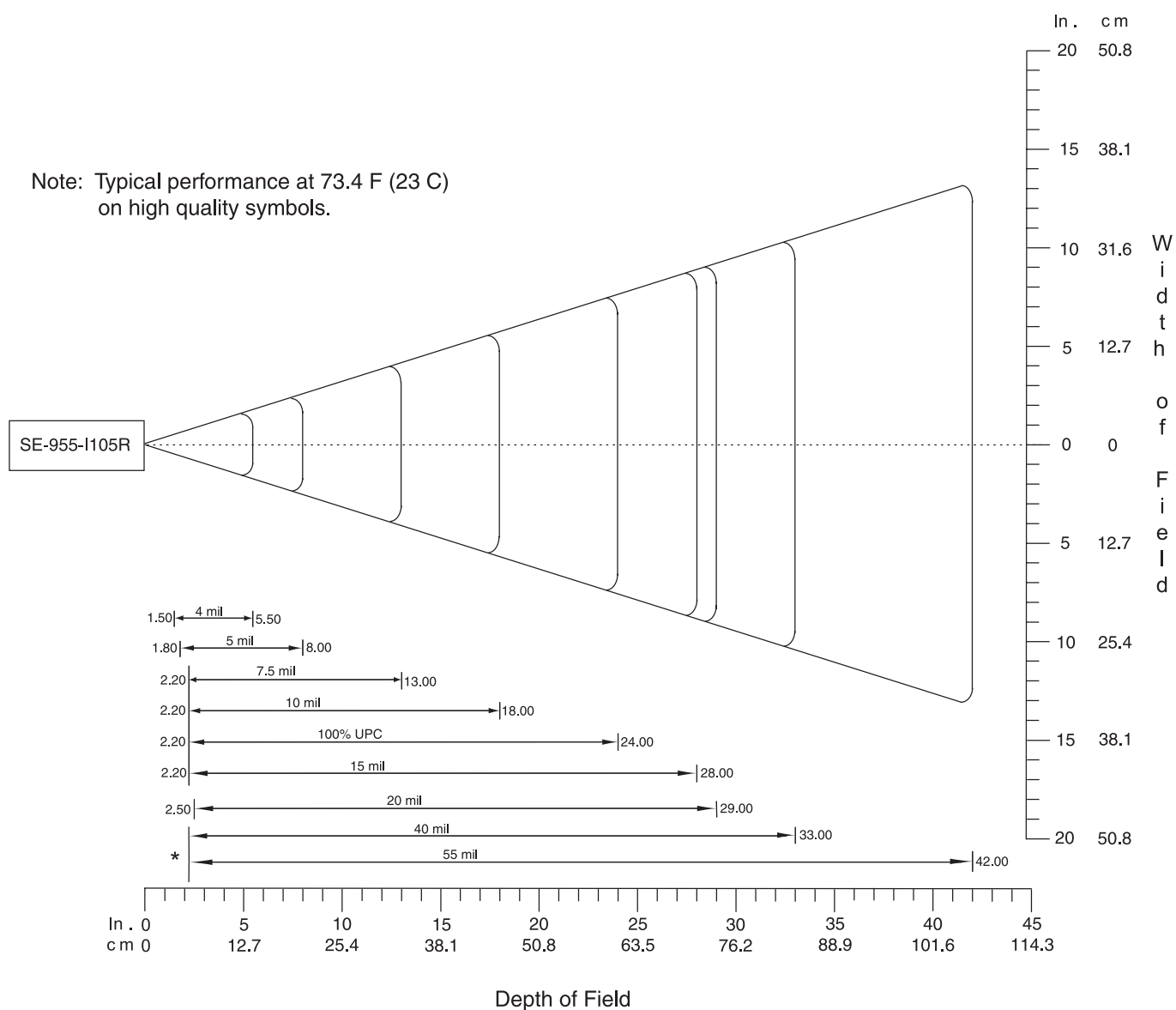
Item	Description
<b>Laser Class</b>	<p><b>SE955-I105R:</b> The scan engine, by itself, is a classified component. It is intended for use in CDRH Class II/IEC Class 2 devices with proper housing, labeling, and instructions to comply with federal and/or international standards.</p> <p><b>SE955-E105R:</b> The scan engine, by itself, is a classified component. It is intended for use in IEC Class 1 devices with proper housing, labeling, and instructions to comply with federal and/or international standards.</p>
<b>RoHS</b>	Meets RoHS requirements
<b>ESD Protection (IEC 61000-4-2)</b>	±2kV Contact pin direct discharge, ±8kV indirect discharge
<b>RF Immunity (IEC 61000-4-3)</b>	10V/m
<b>Emissions</b>	FCC Part 15 Class B, ICES-003 Class B, CISPR Class B, Japan VCCI Class B
<b>Laser Safety</b>	<p><b>SE955-I105R:</b> IEC60825-1 Class 2</p> <p><b>SE955-E105R:</b> IEC60825-1 Class 1</p>
<b>Operating Temperature (chassis)</b>	-4° F to 140° F (-20° C to 60° C)
<b>Storage Temperature</b>	-40°F to 158° F (-40° C to 70° C)
<b>Humidity</b>	5% to 95% (non-condensing)
<b>Height</b>	0.48 in (12.1mm) maximum
<b>Width</b>	0.85 in (21.6mm) maximum
<b>Depth of Chassis</b>	0.89 in (22.6mm) maximum
<b>Weight</b>	0.282 ounces ± -0.009 ounces (8.0grams+/-0.25grams)



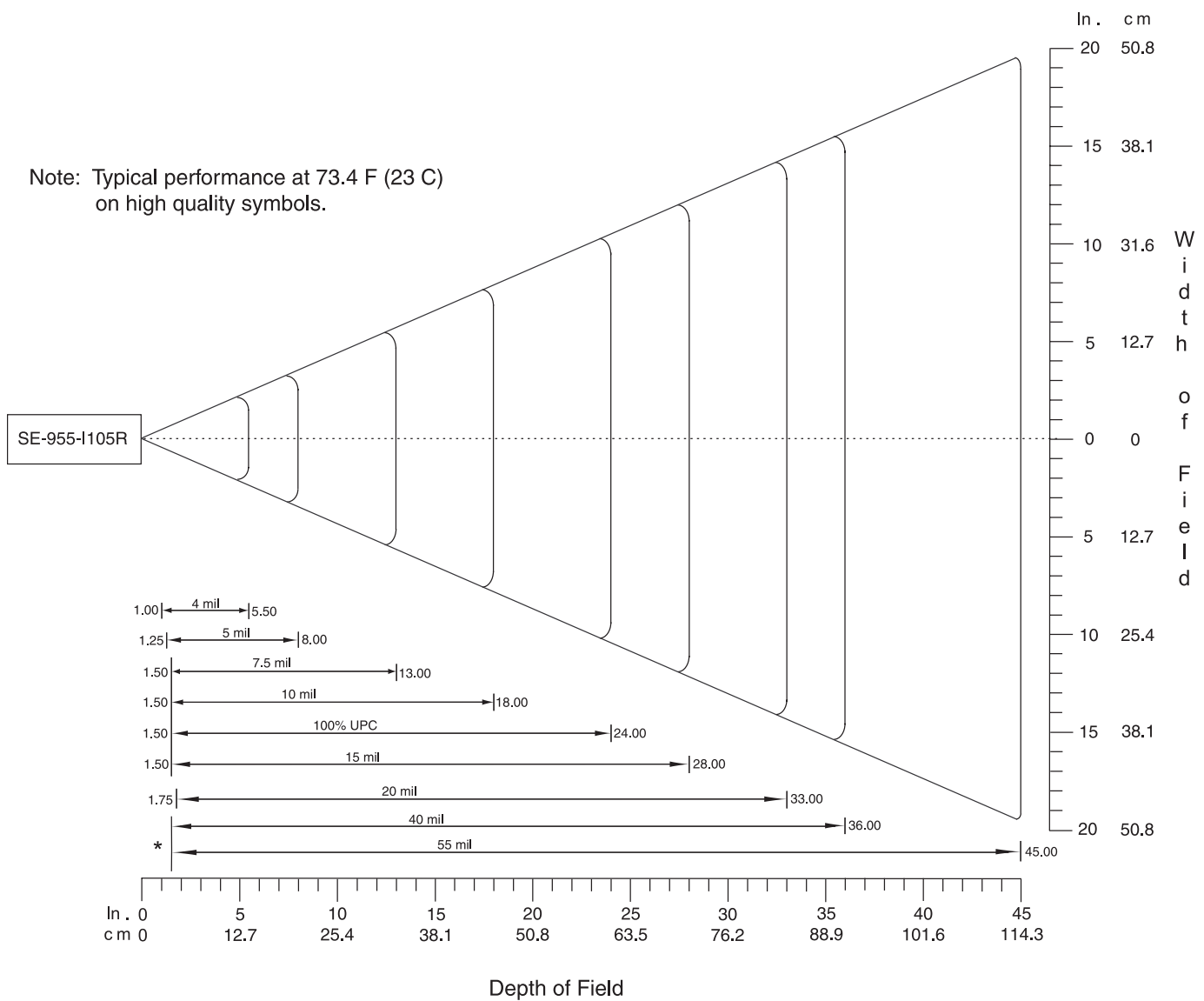
**Figure 5-1. Pitch, Skew and Roll**

## Decode Zones

The decode zones for the SE955-I105R scan engine are shown in Figure 5-2 through Figure 5-3. The decode zones for the SE955-E105R scan engine are shown in Figure 5-4 through Figure 5-5. The figures shown are typical values. Table 5-2 and Table 5-3 lists the typical and guaranteed distances for selected bar code densities. The minimum element width (or "symbol density") is the width in mils of the narrowest element (bar or space) in the symbol.



**Figure 5-2. SE955-I105R Standard Version 35° Decode Zone**



\*Minimum distance determined by symbol length and scan angle

**Figure 5-3. SE955-I105R Standard Version 47° Decode Zone**

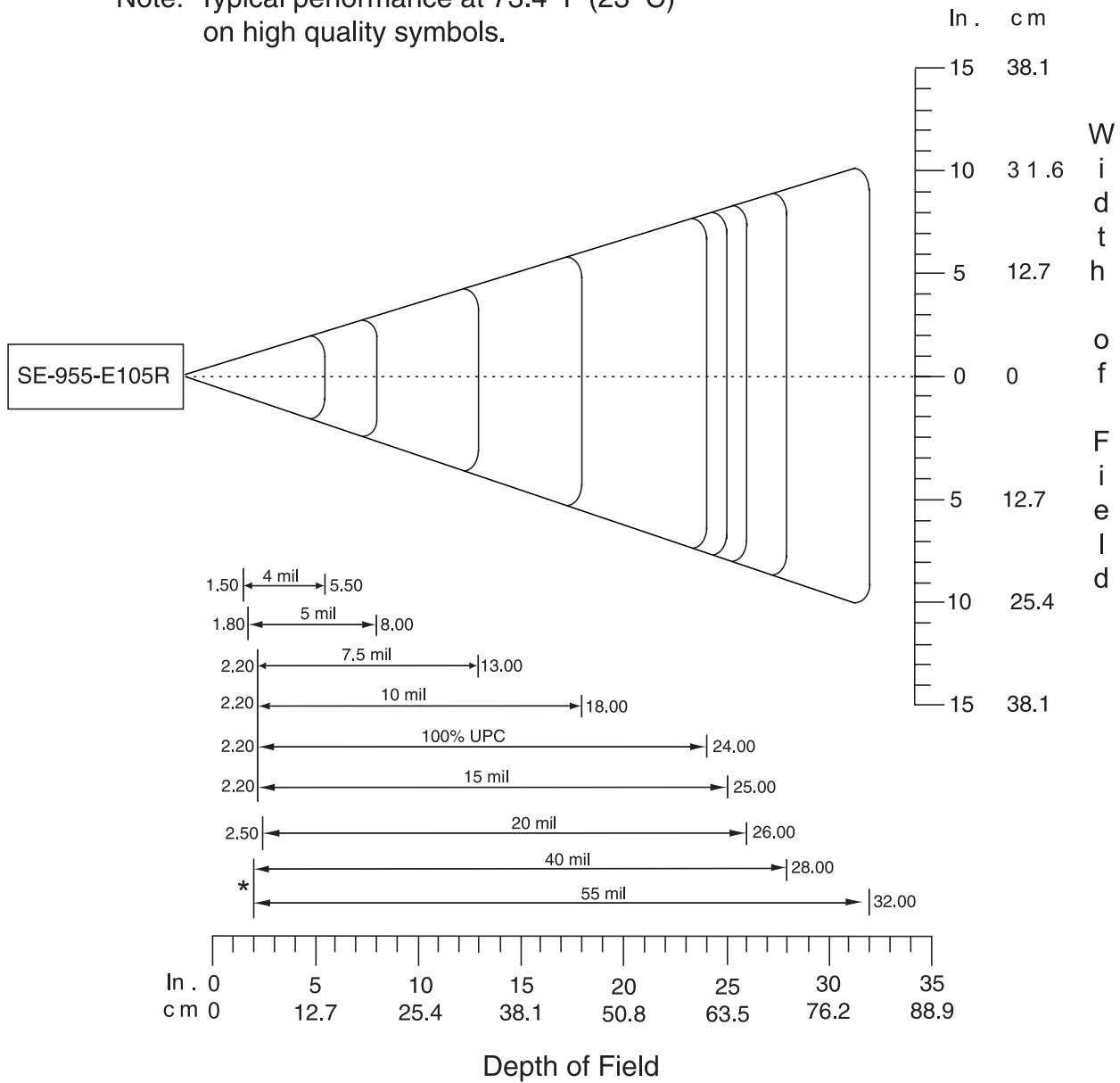
### Table 5-2. SE955-I105R Decode Distances

Symbol Density/ Bar Code Type/ W-N Ratio	Bar Code Content/ Contrast <sup>Note 1</sup>	35 ° Typical Working Ranges		35 ° Guaranteed Working Ranges		47 ° Typical Working Ranges		47 ° Guaranteed Working Ranges	
		Near	Far	Near	Far	Near	Far	Near	Far
<b>4.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.50 in</b> 3.81 cm	<b>5.50 in</b> 13.97 cm	<b>X</b>	<b>3.20 in</b> 8.13 cm	<b>1.00 in</b> 2.50 cm	<b>5.50 in</b> 13.97 cm	<b>2.20 in</b> 5.60 cm	<b>3.20 in</b> 8.13 cm
<b>5.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.80 in</b> 4.57 cm	<b>8.00 in</b> 20.32 cm	<b>X</b>	<b>5.50 in</b> 13.97 cm	<b>1.25 in</b> 3.18 cm	<b>8.00 in</b> 20.32 cm	<b>2.20 in</b> 5.60 cm	<b>5.50 in</b> 13.97 cm
<b>7.5 mil</b> Code 39; 2.5:1	<b>ABCDEF</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>13.00 in</b> 33.02 cm	<b>X</b>	<b>9.50 in</b> 24.13 cm	<b>1.50 in</b> 3.81 cm	<b>13.00 in</b> 33.02 cm	<b>2.00 in</b> 5.08 cm	<b>9.50 in</b> 24.13 cm
<b>10 mil</b> Code 39; 2.5:1	<b>ABCDE</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>18.00 in</b> 45.72 cm	<b>X</b>	<b>14.00 in</b> 35.56 cm	<b>1.50 in</b> 3.81 cm	<b>18.00 in</b> 45.72 cm	<b>1.75 in</b> 4.45 cm	<b>14.00 in</b> 35.56 cm
<b>13 mil</b> 100% UPC	<b>12345678905</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>24.00 in</b> 60.96 cm	<b>2.50 in</b> 6.35 cm	<b>18.00 in</b> 45.72 cm	<b>1.50 in</b> 3.81 cm	<b>24.00 in</b> 60.96 cm	<b>1.75 in</b> 4.45 cm	<b>18.00 in</b> 45.72 cm
<b>15 mil</b> Code 39; 2.5:1	<b>ABCD</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>28.00 in</b> 71.12 cm	<b>X</b>	<b>21.00 in</b> 53.34 cm	<b>1.50 in</b> 3.81 cm	<b>28.00 in</b> 71.12 cm	<b>1.75 in</b> 4.45 cm	<b>21.00 in</b> 53.34 cm
<b>20 mil</b> Code 39; 2.2:1	<b>123</b> 80% MRD	<b>2.50 in</b> 6.35 cm	<b>29.00 in</b> 73.66 cm	<b>X</b>	<b>24.00 in</b> 60.96 cm	<b>1.75 in</b> 4.45 cm	<b>33.00 in</b> 83.82 cm	<b>X</b>	<b>27.00 in</b> 68.58 cm
<b>40 mil</b> Code 39; 2.2:1	<b>AB</b> 80% MRD	<b>X</b>	<b>33.00 in</b> 83.82 cm	<b>X</b>	<b>25.00 in</b> 63.50cm	<b>X</b>	<b>36.00 in</b> 91.44 cm	<b>X</b>	<b>28.00 in</b> 71.12 cm
<b>55 mil</b> Code 39; 2.2:1	<b>CD</b> 80% MRD	<b>X</b>	<b>42.00 in</b> 106.68 cm	<b>X</b>	<b>30.00 in</b> 76.20 cm	<b>X</b>	<b>45.00 in</b> 114.30 cm	<b>X</b>	<b>34.00 in</b> 86.36 cm

Notes:

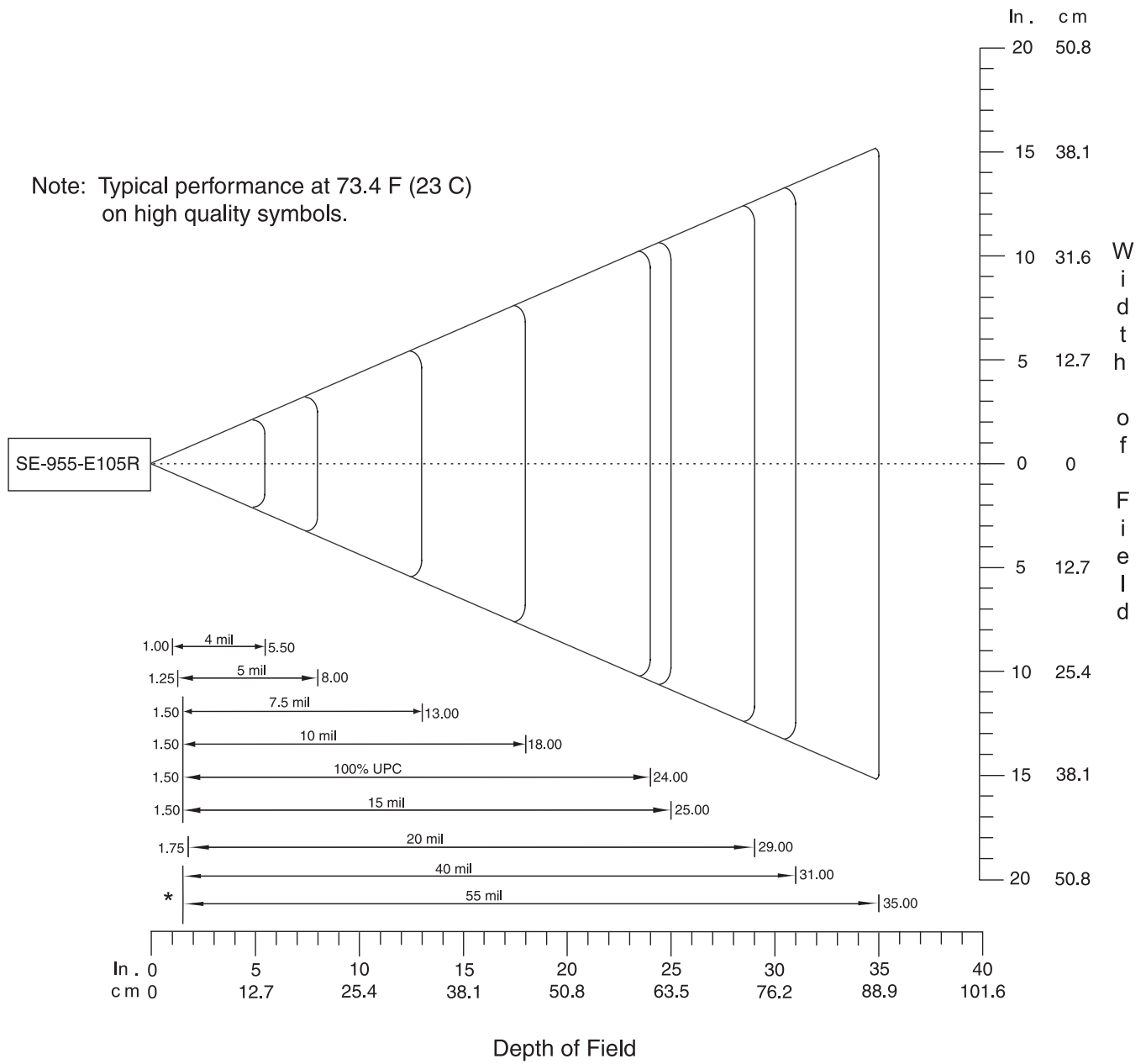
1. CONTRAST measured as Mean Reflective Difference (MRD) at 650 nm.
2. Near ranges on lower densities (not specified) are largely dependent upon the width of the bar code and the scan angle.
3. Working range specifications at ambient temperature (23°C), Photographic quality symbols. pitch=10°, roll=0°, skew=0°, ambient light < 150 ft-candles.
4. X - Dependent on width of bar code.
5. Distances measured from front edge of chassis.

Note: Typical performance at 73.4° F (23° C)  
on high quality symbols.



\*Minimum distance determined by symbol length and scan angle

**Figure 5-4. SE955-E105R Standard Version 35° Decode Zone**



\*Minimum distance determined by symbol length and scan angle

**Figure 5-5. SE955-E105R Standard Version 47° Decode Zone**

### Table 5-3. SE955-E105R Decode Distances

Symbol Density/ Bar Code Type/ W-N Ratio	Bar Code Content/ Contrast <sup>Note 1</sup>	35 ° Typical Working Ranges		35 ° Guaranteed Working Ranges		47 ° Typical Working Ranges		47 ° Guaranteed Working Ranges	
		Near	Far	Near	Far	Near	Far	Near	Far
<b>4.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.50 in</b> 3.81 cm	<b>5.50 in</b> 13.97 cm	<b>X</b>	<b>3.20 in</b> 8.13 cm	<b>1.00 in</b> 2.50 cm	<b>5.50 in</b> 13.97 cm	<b>2.20 in</b> 5.59 cm	<b>3.20 in</b> 8.13 cm
<b>5.0 mil</b> Code 39; 2.5:1	<b>ABCDEFGH</b> 80% MRD	<b>1.80 in</b> 4.57 cm	<b>8.00 in</b> 20.32 cm	<b>X</b>	<b>5.50 in</b> 13.97 cm	<b>1.25 in</b> 3.18 cm	<b>8.00 in</b> 20.32 cm	<b>2.20 in</b> 5.59 cm	<b>5.50 in</b> 13.97 cm
<b>7.5 mil</b> Code 39; 2.5:1	<b>ABCDEF</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>13.00 in</b> 33.02 cm	<b>X</b>	<b>9.00 in</b> 22.86 cm	<b>1.50 in</b> 3.81 cm	<b>13.00 in</b> 33.02 cm	<b>2.00 in</b> 5.08 cm	<b>9.00 in</b> 22.86 cm
<b>10 mil</b> Code 39; 2.5:1	<b>ABCDE</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>18.00 in</b> 45.72 cm	<b>X</b>	<b>13.00 in</b> 33.02 cm	<b>1.50 in</b> 3.81cm	<b>18.00 in</b> 45.72 cm	<b>1.75 in</b> 4.45 cm	<b>13.00 in</b> 33.02 cm
<b>13 mil</b> 100% UPC	<b>12345678905</b> 90% MRD	<b>2.20 in</b> 5.59 cm	<b>24.00 in</b> 60.96 cm	<b>2.50 in</b> 6.35 cm	<b>17.00 in</b> 43.18 cm	<b>1.50 in</b> 3.81 cm	<b>24.00 in</b> 60.96 cm	<b>1.75 in</b> 4.45 cm	<b>17.00 in</b> 43.18cm
<b>15 mil</b> Code 39; 2.5:1	<b>ABCD</b> 80% MRD	<b>2.20 in</b> 5.59 cm	<b>25.00 in</b> 63.50 cm	<b>X</b>	<b>18.00 in</b> 45.72 cm	<b>1.50 in</b> 3.81 cm	<b>25.00 in</b> 63.50 cm	<b>1.75 in</b> 4.45 cm	<b>18.00 in</b> 45.72 cm
<b>20 mil</b> Code 39; 2.2:1	<b>123</b> 80% MRD	<b>2.50 in</b> 6.35 cm	<b>26.00 in</b> 66.04 cm	<b>X</b>	<b>18.00 in</b> 45.72 cm	<b>1.75 in</b> 4.45cm	<b>29.00 in</b> 73.66 cm	<b>X</b>	<b>19.00 in</b> 48.26 cm
<b>40 mil</b> Code 39; 2.2:1	<b>AB</b> 80% MRD	<b>X</b>	<b>28.00 in</b> 71.12 cm	<b>X</b>	<b>19.00 in</b> 48.26 cm	<b>X</b>	<b>31.00 in</b> 78.74 cm	<b>X</b>	<b>21.00 in</b> 53.34 cm
<b>55 mil</b> Code 39; 2.2:1	<b>CD</b> 80% MRD	<b>X</b>	<b>32.00 in</b> 81.28 cm	<b>X</b>	<b>23.00 in</b> 58.42 cm	<b>X</b>	<b>35.00 in</b> 88.90 cm	<b>X</b>	<b>25.00 in</b> 63.50 cm

Notes:

1. CONTRAST measured as Mean Reflective Difference (MRD) at 650 nm.
2. Near ranges on lower densities (not specified) are largely dependent upon the width of the bar code and the scan angle.
3. Working range specifications at ambient temperature (23°C), Photographic quality symbols. pitch=10°, roll=0°, skew=0°, ambient light < 150 ft-candles.
4. X - Dependent on width of bar code.
5. Distances measured from front edge of chassis.



# 6

## ***Regulatory Requirements***

### **Regulatory Requirements**

The sections that follow describe the integration, documentation, and labeling requirements for Class 1 and Class 2 laser products.

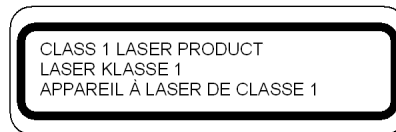
## Required Documentation for Class 1 Laser Products

The documentation accompanying the end product should contain the following:

- “Complies with 21CFR1040.10 and 1040.11 except for deviations pursuant to Laser Notice No. 50, dated July 26, 2001.”
- “EN60825-1:1994+ A1:2002 +A2:2001”
- “IEC60825-1:1993+A1:1997+A2:2001”
- “Class 1 Laser devices are not considered to be hazardous when used for their intended purpose. The following statement is required to comply with US and international regulations:

**Caution:** Use of controls, adjustments or performance of procedures other than those specified herein may result in hazardous laser light exposure.”

A label such as the one below should appear in the product documentation, depending on the end product. Refer to the current applicable laser safety standards for the end product or specific requirements.



**Figure 6-1. Example of Class 1 Laser Warning Label**

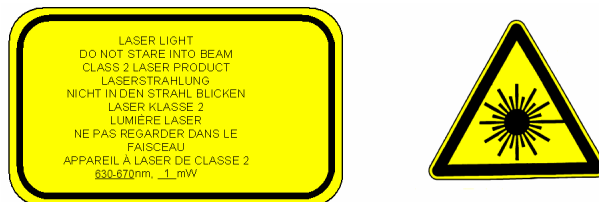
## Required Documentation for Class 2 Laser Products

The documentation accompanying the end product should contain the following:

- “Complies with 21CFR1040.10 and 1040.11 except for deviations pursuant to Laser Notice No. 50, dated July 26, 2001.”
- “EN60825-1:1994+ A1:2002 +A2:2001”
- “IEC60825-1:1993+A1:1997+A2:2001”
- “Caution: Use of controls, adjustments or performance of procedures other than those specified herein may result in hazardous laser light exposure.

Class 2 laser scanners use a low power, visible light diode. As with any very bright light source, such as the sun, the user should avoid staring directly into the light beam. Momentary exposure to a Class 2 laser is not known to be harmful.”

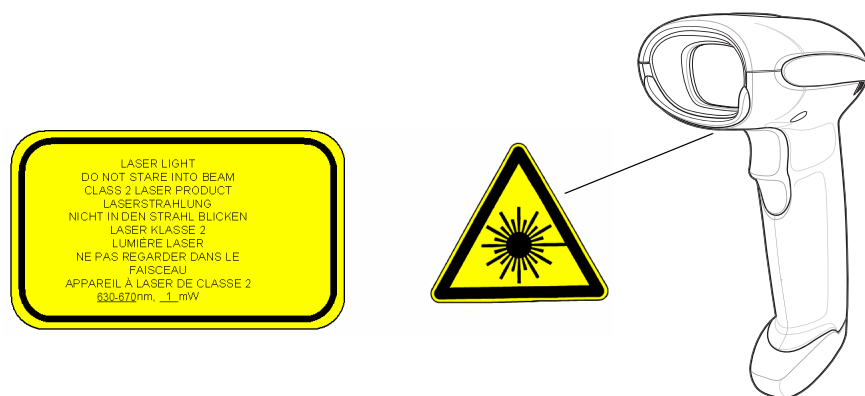
A copy of the product’s laser safety label, such as the one below, should appear in the product documentation, depending on the end product. Refer to the current applicable laser safety standards for the end product or specific requirements.



**Figure 6-2. Example of Class 2 Laser Warning Label**

## Required Documentation for all End Products

The documentation should contain a diagram showing the location of the laser aperture and warning statement as shown in the example in [Figure 6-3](#).



**Figure 6-3. Example of Diagram Showing Class 2 Laser Labeling**

## Required Labeling for Class 1 End Products

The following guidance is provided for end product labelling for products containing Class 1 scan engines:

### 1 - Certification Statement from FDA/IEC Label Set, 2005

The following text must appear on the end product:

- "Complies with 21CFR1040.10 and 1040.11 except for deviations pursuant to Laser Notice No. 50, dated July 26, 2001. "
- "EN60825-1:1994+ A1:2002 +A2:2001"
- "IEC60825-1:1993+A1:1997+A2:2001"

**Required location:** For most end products, this text should be located on the exterior of the product, or inside the battery compartment, software module compartment or other user accessible area. Access to these areas must not require special tools. Refer to the current applicable laser safety standards for the end product for specific requirements for the end product.

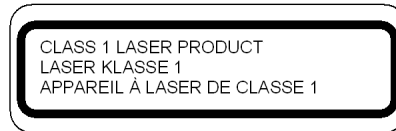
**Font:** Use a sans serif type font such as Arial or equivalent. The height must be at least 0.032 in. minimum.

**Color:** No color requirement. Contrast must be high enough to render this text legible.

### 2 - Identification

The name and address of the manufacturer must appear on the product.

### **3 - Laser Warning Label**



**Figure 6-4. Example of Class 1 Laser Warning Label**

Required location: For most end products, the label shown in [Figure 6-4](#) should be located on the exterior of the product. Refer to the current applicable laser safety standards for the end product for specific requirements for the end product.

Font: Use a sans serif type font such as Arial or equivalent. The height must be at least 0.032 in. minimum.

Color: No color requirement. Contrast must be high enough to render this text legible.

### **4 - Protective Housing Statement**

Not required for Class 1.

## Required Labeling for Class 2 End Products

The following guidance is provided for end product labelling for products containing Class 2 scan engines:

### 1 - Certification Statement from FDA/IEC Label Set, 2005

The following text must appear on the product:

- "Complies with 21CFR1040.10 and 1040.11 except for deviations pursuant to Laser Notice No. 50, dated July 26, 2001. "
- "EN60825-1:1994+ A1:2002 +A2:2001"
- "IEC60825-1:1993+A1:1997+A2:2001"

**Required location:** For most end products, this text should be located on the exterior of the product, or inside the battery compartment, software module compartment or other user accessible area. Access to these areas must not require special tools. Refer to the current applicable laser safety standards for the end product for specific requirements for the end product.

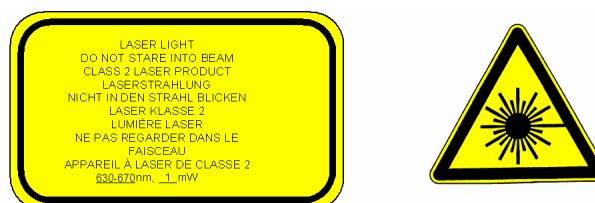
**Font:** Use a sans serif type font such as Arial or equivalent. The height must be at least 0.032 in. minimum.

**Color:** No color requirement. Contrast must be high enough to render this text legible.

### 2 - Identification

The name and address of the manufacturer must appear on the product.

### 3 - Laser Warning Label



**Figure 6-5. Example of Class 2 Laser Warning Label**

**Required location:** For most end products, the label shown in [Figure 6-5](#) should be located on the exterior of the product. Refer to the current applicable laser safety standards for the end product for specific requirements for the end product.

**Font:** Use a sans serif type font such as Arial or equivalent. The height must be at least 0.032 in. minimum.

**Color:** Must have a yellow background with black text

## **4 - Protective Housing Statements**

CAUTION - CLASS 2 LASER LIGHT WHEN OPEN, DO NOT STARE INTO THE BEAM

ATTENTION - LUMIÈRE LASER EN CAS D'OUVERTURE. NE PAS REGARDER DANS LE FAISCEAU.

VORSICHT - LASERLICHT KLASSE 2, WENN ABDECKUNG GEÖFFNET. NICHT IN DEN STRAHL BLICKEN.

Required location: For most end products, this text should be located on the exterior of the product, or inside the battery compartment, software module compartment or other user accessible area. Access to these areas must not require special tools. Refer to the current applicable laser safety standards for the end product for specific requirements for the end product.

Font: Use a sans serif type font such as Arial or equivalent. The height must be at least 0.032 in. minimum.

Color: No color requirement. Contrast must be high enough to render this text legible.

## **Recycling**

The Customer shall be responsible for complying with all recycling laws and regulations, including European Directive: Waste Electrical and Electronic Equipment (WEEE). Symbol shall have no responsibility for collecting the products sold to Customer.

## **RoHS Compliance**

This product is RoHS compliant.

# Application Notes

## Introduction

This chapter includes AC electrical characteristics and timing information.

## AC Electrical Characteristics

AC electrical characteristics appear in [Table 7-1](#). All output lines are measured with 10K pull-up.

**Table 7-1. Timing Characteristics**

Symbol	Figure	Parameter	Min	Max	Unit
General Characteristics					
$t_f$	<a href="#">Figure 7-1</a>	High-to-Low fall time, all outputs, $C_L = 50$ pf		1.0	$\mu$ sec
$t_r$	<a href="#">Figure 7-1</a>	Low-to-High rise time, all outputs, $C_L = 50$ pf		1.0	$\mu$ sec
Serial I/O Timing, Host Transmit					
$t_{rlcl}$	<a href="#">Figure 7-2</a>	Request to Send low to Clear to Send low	0	25	msec
$t_{clxl}$	<a href="#">Figure 7-2</a>	Clear to Send low to first start bit		Note 2	
$t_{xlxl}$	<a href="#">Figure 7-2</a>	Byte to byte delay, (see Note 1)		990	msec
$t_{thrh}$	<a href="#">Figure 7-2</a>	End of the packet to RTS* high		Note 4	msec
Notes: 1. If byte to byte delay exceeds the maximum specified time, a transmission error is declared. The sender is expected to retransmit the packet in its entirety. 2. The host may hold the Host RTS* low indefinitely, but it locks out the SE955 from transmitting. 3. The decoder may transmit any time the Host RTS* is high. 4. The host should release its Host RTS* as soon as possible after transmitting so the decoder can process the message. 5. The SE955's micro-controller is in full operation whenever the PWRDWN line is driven low. 6. See the <a href="#">Power Management on page 1-4</a> if trigger is not pulled after the maximum specified amount of time. 7. In addition, refer to <a href="#">Parameter # 0x88 on page 8-10</a> and <a href="#">Parameter # 0x8A on page 8-13</a> .					

**Table 7-1. Timing Characteristics (Continued)**

Symbol	Figure	Parameter	Min	Max	Unit
Serial I/O Timing, Decoder Transmit, (see Note 3)					
$t_{vvl}$	<a href="#">Figure 7-4</a>	Byte to byte delay, (see Note 1)		99	msec
Hardware Trigger Timing					
$t_{glwl}$	<a href="#">Figure 7-5</a>	Trigger hold time, level and pulse trigger mode, (see Note 6)	6		msec
$t_{ghtw}$	<a href="#">Figure 7-5</a>	Trigger release time, level and pulse trigger mode (see Note 6)	25		msec
$t_{dbt}$	<a href="#">Figure 7-3</a>	Trigger debounce time		1	msec
Beeper Timing					
$f_{blht}$	<a href="#">Figure 7-6</a>	Beeper frequency	1220	3770	Hz
$t_{btw}$	<a href="#">Figure 7-6</a>	Beeper duration (decode)	90 (typ)		msec
Power Up Timing					
$t_{ehpm}$	<a href="#">Figure 7-7</a>	$V_{BATT}$ rise time		10	msec
Wake Up Timing					
$t_{aldl}$	<a href="#">Figure 7-8</a>	From wake up to full operation, (see Note 5)		8	msec
$t_{dlgl}$	<a href="#">Figure 7-8</a>	Trigger low after full operation, (see Notes 6 and 7)	0	1	sec
$t_{ehdl}$	<a href="#">Figure 7-9</a>	Power Enable High to Power Down Logic Low	0	8	msec
Notes: 1. If byte to byte delay exceeds the maximum specified time, a transmission error is declared. The sender is expected to retransmit the packet in its entirety. 2. The host may hold the Host RTS* low indefinitely, but it locks out the SE955 from transmitting. 3. The decoder may transmit any time the Host RTS* is high. 4. The host should release its Host RTS* as soon as possible after transmitting so the decoder can process the message. 5. The SE955's micro-controller is in full operation whenever the PWRDWN line is driven low. 6. See the <a href="#">Power Management on page 1-4</a> if trigger is not pulled after the maximum specified amount of time. 7. In addition, refer to <a href="#">Parameter # 0x88 on page 8-10</a> and <a href="#">Parameter # 0x8A on page 8-13</a> .					



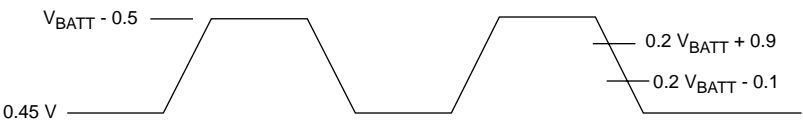
# Timing Waveforms

## Explanation Of The AC Symbols

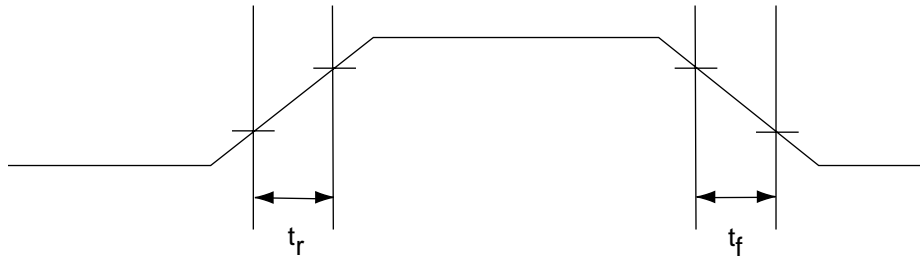
Each timing symbol has five characters. The first character is either “t” for time or “f” for frequency. The other characters indicate the name of the signal or the logical status of that signal. Designations are:

<b>a</b>	WKUP*
<b>b</b>	BPR*
<b>c</b>	Host CTS*
<b>d</b>	PWRDWN
<b>f</b>	float, fall time
<b>g</b>	trigger
<b>h</b>	logic level high
<b>l</b>	logic level low
<b>pm</b>	minimum voltage level
<b>r</b>	Host RTS*
<b>tw</b>	time duration
<b>v</b>	Host RXD
<b>w</b>	width
<b>x</b>	Host TXD
* Active Low <b>Examples:</b> $t_{bltw}$ = Beeper drive low time $t_{rlcl}$ = Time for RTS low to CTS low	

## AC Test Points



AC inputs during testing are driven at  $V_{BATT} - 0.5$  for logic “1” and  $0.45$  for logic “0.” Timing measurements are made at  $0.2 V_{BATT} + 0.9$  and  $0.2 V_{BATT} - 0.1$ .



**Figure 7-1. General Characteristics**

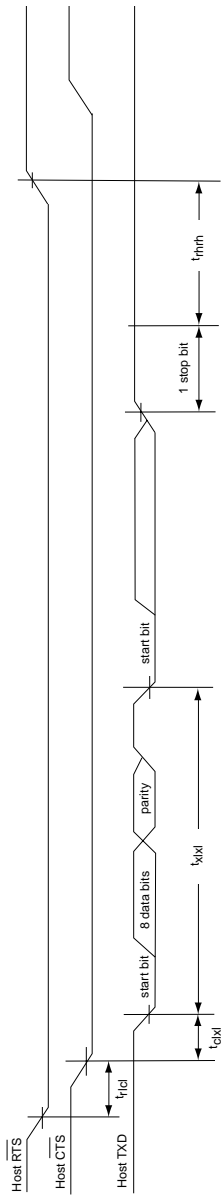


Figure 7-2. Serial I/O Timing, Host Transmit

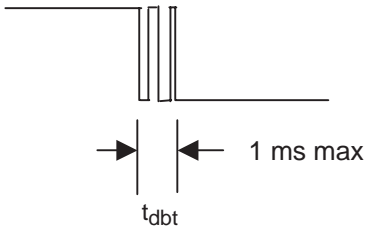


Figure 7-3. Trigger Debounce Timing

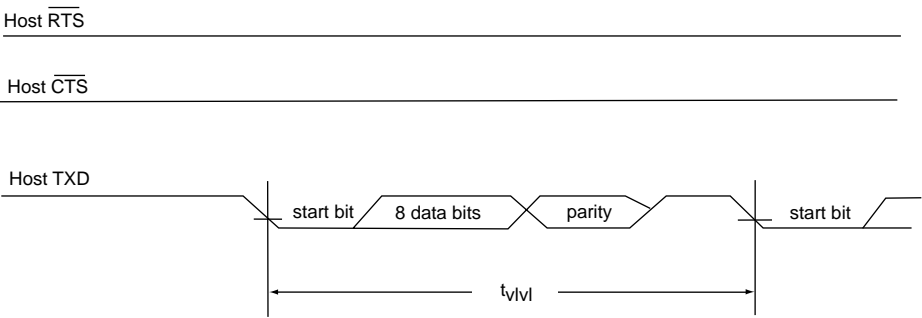


Figure 7-4. Serial I/O Timing, Decoder Transmit

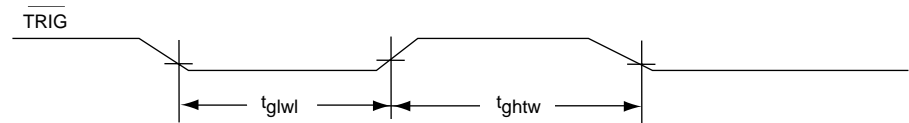


Figure 7-5. Hardware Trigger Timing

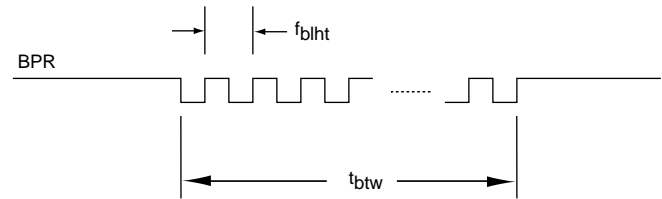


Figure 7-6. Beeper Timing

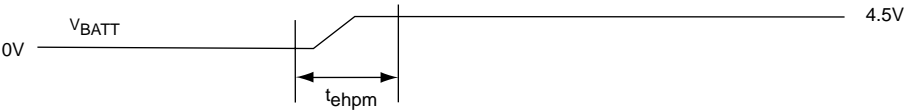


Figure 7-7.  $V_{BATT}$  Rise Time

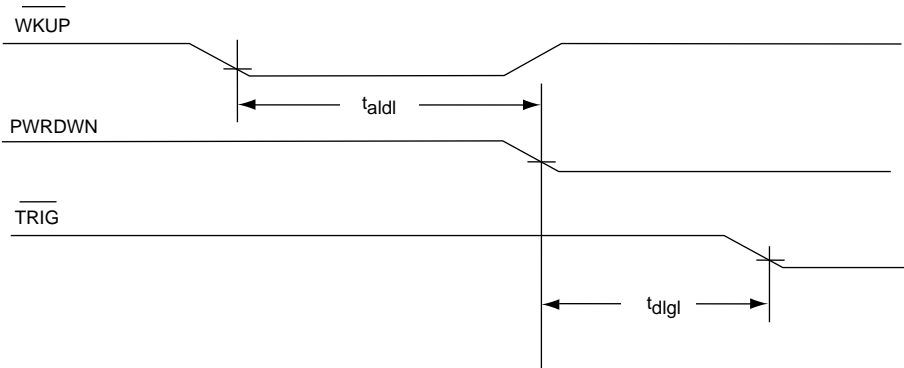


Figure 7-8. Wake Up Timing

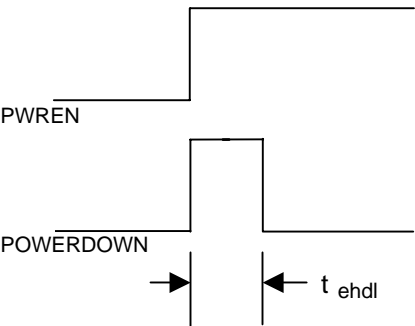


Figure 7-9. Power Enable to Power Down



## Parameter Menus

### Introduction

This chapter describes the programmable parameters, provides bar codes for programming, and hexadecimal equivalents for host parameter programming through SSI.

### Operational Parameters

The SE955 is shipped with the factory default settings shown in [Table 8-1 on page 8-2](#). These factory default values are stored in non-volatile memory and are preserved even when the scan engine is powered down. Changes to the factory default values can be stored as custom defaults. These values are also stored in non-volatile memory and are preserved even when the scan engine is powered down.

To change the parameter values:

- Scan the appropriate bar codes included in this chapter. The new values replace the existing memory values. To set the new values as custom defaults, scan the **Write to Custom Defaults** bar code. The factory default or custom default parameter values can be recalled by scanning the **Set Factory Defaults** bar code or the **Restore Defaults** bar code on [page 8-7](#).

or

- Send the parameter through the scan engine's serial port using the SSI command PARAM\_SEND. Hexadecimal parameter numbers are shown in this chapter below the parameter title, and options appear in parenthesis beneath the accompanying bar codes. Instructions for changing parameters using this method are found in [Chapter 10, Simple Serial Interface](#).

[Table 8-1](#) lists the factory defaults for all parameters. To change any option, scan the appropriate bar code(s).

**Table 8-1. Factory Default Table**

Parameter	Parameter Number (Hex)	Factory Default	Page Number
Set Factory Default		All Defaults	<a href="#">8-7</a>
Beeper Volume	0x8C	Medium	<a href="#">8-8</a>
Beeper Tone	0x91	Medium Frequency	<a href="#">8-9</a>
Beeper Frequency Adjustment	0xF0 0x91	2500 Hz	<a href="#">8-10</a>
Laser On Time	0x88	3.0 sec	<a href="#">8-10</a>
Aim Duration	0xED	0.0 sec	<a href="#">8-11</a>
Scan Angle	0xBF	Wide (47°)	<a href="#">8-11</a>
Power Mode	0x80	Low Power	<a href="#">8-12</a>
Trigger Mode	0x8A	Level	<a href="#">8-13</a>
Time-out Between Same Symbol	0x89	1.0 sec	<a href="#">8-14</a>
Beep After Good Decode	0x38	Enable	<a href="#">8-14</a>
Transmit "No Read" Message	0x5E	Disable	<a href="#">8-15</a>
Parameter Scanning	0xEC	Enable	<a href="#">8-15</a>
Linear Code Type Security Levels	0x4E	1	<a href="#">8-16</a>
Bi-directional Redundancy	0x43	Disable	<a href="#">8-17</a>
<b>UPC/EAN</b>			
UPC-A	0x01	Enable	<a href="#">8-18</a>
UPC-E	0x02	Enable	<a href="#">8-18</a>
UPC-E1	0x0C	Disable	<a href="#">8-19</a>
EAN-8	0x04	Enable	<a href="#">8-19</a>
EAN-13	0x03	Enable	<a href="#">8-20</a>
Bookland EAN	0x53	Disable	<a href="#">8-20</a>
Decode UPC/EAN Supplementals	0x10	Ignore	<a href="#">8-21</a>
Decode UPC/EAN Supplemental Redundancy	0x50	7	<a href="#">8-22</a>
Transmit UPC-A Check Digit	0x28	Enable	<a href="#">8-23</a>
Transmit UPC-E Check Digit	0x29	Enable	<a href="#">8-23</a>
*See <a href="#">Table 10-9 on page 10-17</a> for formatting of any parameter whose number is 0x100 or greater.			



**Table 8-1. Factory Default Table (Continued)**

Parameter	Parameter Number (Hex)	Factory Default	Page Number
Transmit UPC-E1 Check Digit	0x2A	Enable	<a href="#">8-24</a>
UPC-A Preamble	0x22	System Character	<a href="#">8-25</a>
UPC-E Preamble	0x23	System Character	<a href="#">8-26</a>
UPC-E1 Preamble	0x24	System Character	<a href="#">8-27</a>
Convert UPC-E to A	0x25	Disable	<a href="#">8-28</a>
Convert UPC-E1 to A	0x26	Disable	<a href="#">8-28</a>
EAN-8 Zero Extend	0x27	Disable	<a href="#">8-29</a>
Convert EAN-8 to EAN-13 Type	0xE0	Type is EAN-13	<a href="#">8-29</a>
UPC/EAN Security Level	0x4D	0	<a href="#">8-30</a>
UCC Coupon Extended Code	0x55	Disable	<a href="#">8-31</a>
<b>Code 128</b>			
Code-128	0x08	Enable	<a href="#">8-32</a>
UCC/EAN-128	0x0E	Enable	<a href="#">8-32</a>
ISBT 128	0x54	Enable	<a href="#">8-33</a>
<b>Code 39</b>			
Code 39	0x00	Enable	<a href="#">8-34</a>
Trioptic Code 39	0x0D	Disable	<a href="#">8-34</a>
Convert Code 39 to Code 32	0x56	Disable	<a href="#">8-35</a>
Code 32 Prefix	0xE7	Disable	<a href="#">8-35</a>
Set Length(s) for Code 39	0x12 0x13	2-55	<a href="#">8-36</a>
Code 39 Check Digit Verification	0x30	Disable	<a href="#">8-37</a>
Transmit Code 39 Check Digit	0x2B	Disable	<a href="#">8-37</a>
Code 39 Full ASCII Conversion	0x11	Disable	<a href="#">8-38</a>
<b>Code 93</b>			
Code 93	0x09	Disable	<a href="#">8-39</a>
Set Length(s) for Code 93	0x1A 0x1B	4-55	<a href="#">8-40</a>
*See <a href="#">Table 10-9 on page 10-17</a> for formatting of any parameter whose number is 0x100 or greater.			

**Table 8-1. Factory Default Table (Continued)**

Parameter	Parameter Number (Hex)	Factory Default	Page Number
<b>Code 11</b>			
Code 11	0x0A	Disable	<a href="#">8-41</a>
Set Lengths for Code 11	0x1C 0x1D	4 to 55	<a href="#">8-41</a>
Code 11 Check Digit Verification	0x34	Disable	<a href="#">8-43</a>
Transmit Code 11 Check Digit(s)	0x2F	Disable	<a href="#">8-43</a>
<b>Interleaved 2 of 5</b>			
Interleaved 2 of 5	0x06	Enable	<a href="#">8-44</a>
Set Length(s) for I 2 of 5	0x16 0x17	14	<a href="#">8-45</a>
I 2 of 5 Check Digit Verification	0x31	Disable	<a href="#">8-47</a>
Transmit I 2 of 5 Check Digit	0x2C	Disable	<a href="#">8-48</a>
Convert I 2 of 5 to EAN 13	0x52	Disable	<a href="#">8-48</a>
<b>Discrete 2 of 5</b>			
Discrete 2 of 5	0x05	Disable	<a href="#">8-49</a>
Set Length(s) for D 2 of 5	0x14 0x15	12	<a href="#">8-50</a>
<b>Chinese 2 of 5</b>			
Chinese 2 of 5	0xF0 0x98	Disable	<a href="#">8-51</a>
<b>Codabar</b>			
Codabar	0x07	Disable	<a href="#">8-52</a>
Set Lengths for Codabar	0x18 0x19	5-55	<a href="#">8-53</a>
CLSI Editing	0x36	Disable	<a href="#">8-54</a>
NOTIS Editing	0x37	Disable	<a href="#">8-54</a>
<b>MSI</b>			
MSI	0x0B	Disable	<a href="#">8-55</a>
Set Length(s) for MSI	0x1E 0x1F	6-55	<a href="#">8-56</a>
*See <a href="#">Table 10-9 on page 10-17</a> for formatting of any parameter whose number is 0x100 or greater.			

**Table 8-1. Factory Default Table (Continued)**

Parameter	Parameter Number (Hex)	Factory Default	Page Number
MSI Check Digits	0x32	One	<a href="#">8-57</a>
Transmit MSI Check Digit	0x2E	Disable	<a href="#">8-57</a>
MSI Check Digit Algorithm	0x33	Mod 10/Mod 10	<a href="#">8-58</a>
<b>RSS</b>			
RSS-14	0xF0 0x52	Disable	<a href="#">8-59</a>
RSS-Limited	0xF0 0x53	Disable	<a href="#">8-59</a>
RSS-Expanded	0xF0 0x54	Disable	<a href="#">8-60</a>
Convert RSS to UPC/EAN	0xF0 0x8D	Disable	<a href="#">8-60</a>
<b>Data Options</b>			
Transmit Code ID Character	0x2D	None	<a href="#">8-61</a>
Prefix/Suffix Values Prefix Suffix 1 Suffix 2	0x69 0x68 0x6A	NULL LF CR	<a href="#">8-62</a>
Scan Data Transmission Format	0xEB	Data as is	<a href="#">8-63</a>
<b>Serial Interface</b>			
Baud Rate	0x9C	9600	<a href="#">8-65</a>
Parity	0x9E	None	<a href="#">8-67</a>
Software Handshaking	0x9F	Enable	<a href="#">8-68</a>
Decode Data Packet Format	0xEE	Unpacketed	<a href="#">8-69</a>
Host Serial Response Time-out	0x9B	2 sec	<a href="#">8-69</a>
Stop Bit Select	0x9D	1	<a href="#">8-70</a>
Intercharacter Delay	0x6E	0	<a href="#">8-70</a>
Host Character Time-out	0xEF	200 msec	<a href="#">8-70</a>
<b>Event Reporting*</b>			
Decode Event	0xF0 0x00	Disable	<a href="#">8-71</a>
*See <a href="#">Table 10-9 on page 10-17</a> for formatting of any parameter whose number is 0x100 or greater.			

**Table 8-1. Factory Default Table (Continued)**

Parameter	Parameter Number (Hex)	Factory Default	Page Number
Boot Up Event	0xF0 0x02	Disable	<a href="#">8-72</a>
Parameter Event	0xF0 0x03	Disable	<a href="#">8-72</a>
*See <a href="#">Table 10-9 on page 10-17</a> for formatting of any parameter whose number is 0x100 or greater.			

## Set Default Parameter

The SE955 can be reset to two types of defaults: factory defaults or custom defaults. Scan the appropriate bar code below to reset the SE955 to its default settings and/or set the scan engine's current settings as the custom default.

- **Restore Defaults** - Scan this bar code to reset all default parameters as follows.
  - If custom defaults were set by scanning **Write to Custom Defaults**, scan **Restore Defaults** to retrieve and restore the scan engine's custom default settings.
  - If no custom defaults were set, scan **Restore Defaults** to restore the factory default values listed in [Table 8-1 on page 8-2](#).
- **Set Factory Defaults** - Scan this bar code to restore the factory default values listed in [Table 8-1 on page 8-2](#). If custom defaults were set, they are eliminated.
- **Write to Custom Defaults** - Scan this bar code to store the current scan engine settings as custom defaults. Once custom default settings are stored, they can be recovered at any time by scanning **Restore Defaults**.



**\* Restore Defaults**



**Set Factory Defaults**



**Write to Custom Defaults**

## Beeper Volume

### ***Parameter # 0x8C***

To select a decode beep volume, scan the appropriate bar code.



**Low**  
**(0x02)**



**\*Medium**  
**(0x01)**

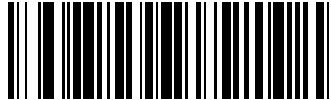


**High**  
**(0x00)**

## Beeper Tone

### ***Parameter # 0x91***

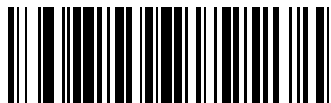
To select a decode beep frequency (tone), scan the appropriate bar code.



**Low Frequency  
(0x02)**



**\*Medium Frequency  
(0x01)**



**High Frequency  
(0x00)**

## Beeper Frequency Adjustment

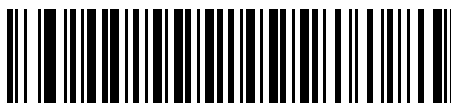
### **Parameter # 0xF0 0x91**

This parameter adjusts the frequency of the high beeper tone from the nominal 2500 Hz to another frequency matching the resonances of the installation. It is programmable in 10 Hz increments from 1220 Hz to 3770 Hz.

To increase the frequency, scan the bar code below, then scan three numeric bar codes beginning on [page 8-73](#) that correspond to the desired frequency adjustment divided by 10. For example, to set the frequency to 3000 Hz (an increase of 500 Hz), scan numeric bar codes 0, 5, 0, corresponding to 50, or (500/10).

To decrease the frequency, scan the bar code below, then scan three numeric bar codes beginning on [page 8-73](#) that correspond to the value (256 - desired adjustment/10). For example, to set the frequency to 2000 Hz (a decrease of 500 Hz), scan numeric bar codes 2, 0, 6, corresponding to 206, or (256 - 500/10).

To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



**Beeper Frequency Adjustment**  
(Default: 2500 Hz)

## Laser On Time

### **Parameter # 0x88**

This parameter sets the maximum time decode processing continues during a scan attempt. It is programmable in 0.1 second increments from 0.50 to 25.5 seconds.

To set a Laser On Time, scan the bar code below. Next scan two numeric bar codes beginning on [page 8-73](#) that correspond to the desired on time. Single digit numbers must have a leading zero. For example, to set an on time of 0.5 seconds, scan the bar code below, then scan the "0", "5" and "0" bar codes; to set an on time of 10.5 seconds, scan the bar code below, then scan the "1", "0" and "5" bar codes. To change the selection or cancel an incorrect entry, scan [Cancel](#) on [page 8-74](#).



**Laser On Time**  
(Default: 3.0 sec.)



# Aim Duration

## Parameter # 0xED

When a scan engine with an aim mode (see [Table 10-10 on page 10-19](#)) is triggered either by a trigger pull, or a [START\\_DECODE](#) command, this parameter sets the duration the aiming pattern is seen before a scan attempt begins. It does not apply to the aim signal or the [AIM\\_ON](#) command. It is programmable in 0.1 second increments from 0.0 to 9.9 seconds. No aim pattern is visible when the value is 0.0. For more information on the use of this parameter, see the [AIM\\_ON](#) command on [page 10-5](#).

To set an aim duration, scan the bar code below. Next scan two numeric bar codes beginning on [page 8-73](#) that correspond to the desired aim duration. Single digit numbers must have a leading zero. For example, to set an aim duration of 0.5 seconds, scan the bar code below, then scan the “0” and “5” bar codes. To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



**Aim Duration**  
(Default: 0.0 sec.)

# Scan Angle

## Parameter # 0xBF

This parameter sets the scan angle to narrow or wide.



**Narrow Angle (35°)**  
(0x05)



**\*Wide Angle (47°)**  
(0x06)



Note

The allowed values for this setting are different for some legacy models of scan engines. These old values can still be used and are interpreted by the scan engine as follows.

0x00 - 0x05	0x06 - 0x2C	0x2D - 0x4A	0x4B - 0xFF
Narrow (0x05)	Wide (0x06)	Narrow (0x05)	Wide (0x06)

## Power Mode

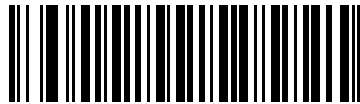
### ***Parameter # 0x80***

This parameter determines the power mode of the engine.

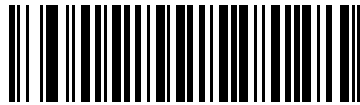
In Low Power mode, the scan engine enters into a low power consumption Sleep power state whenever possible (provided all WAKEUP commands were released). See [Power Management on page 1-4](#).

In Continuous Power mode, the scan engine remains in the Awake state after each decode attempt (see [Power Management on page 1-4](#)).

The Sleep and Awake commands (see [SLEEP on page 10-22](#) and [WAKEUP on page 10-25](#)) can be used to change the power state in either the Low Power mode or the Continuous Power mode.



**Continuous Power**  
**(0x00)**



**Low Power**  
**(0x01)**

## Triggering Modes

### Parameter # 0x8A

Choose one of the options below to trigger the scan engine. Bar codes and option numbers are on the following page.

- **Scan (Level)** - A trigger pull activates the laser and decode processing. The laser remains on and decode processing continues until a trigger release, a valid decode, or the Laser On Time-out is reached.
- **Scan (Pulse)** - A trigger pull activates the laser and decode processing. The laser remains on and decode processing continues until a valid decode or the Laser On Time-out is reached.
- **Continuous** - The laser is always on and decoding.
- **Blink** - This trigger mode is used for triggerless operation. Scanning range is reduced in this mode. This mode cannot be used with engines that support an aim mode (see [Table 10-10 on page 10-19](#)).
- **Host** - A host command issues the triggering signal. The scan engine interprets an actual trigger pull as a Level triggering option.



**\*Level**  
**(0X00)**



**Pulse**  
**(0X02)**



**Continuous**  
**(0X04)**



**Blinking**  
**(0X07)**



**Host**  
**(0X08)**

## Time-out Between Same Symbol

### ***Parameter # 0x89***

When in Continuous triggering mode, this parameter sets the minimum time that must elapse before the scan engine decodes a second bar code identical to one just decoded. This reduces the risk of accidentally scanning the same symbol twice. It is programmable in 0.1 second increments from 0.0 to 9.9 seconds.

To set a time-out between same symbol, scan the bar code below. Next scan two numeric bar codes beginning on [page 8-73](#) that correspond to the desired time-out. Single digit values must have a leading zero. For example, to set a time-out of 0.5 seconds, scan the bar code below, then scan the "0" and "5" bar codes. To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



**Time-out Between Same Symbol**  
(Default: 1.0 sec.)

## Beep After Good Decode

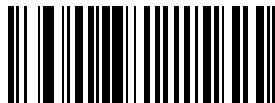
### ***Parameter # 0x38***

Scan this symbol to set the scan engine to beep after a good decode.



**\*Beep After Good Decode**  
(0x01)

Scan this symbol to set the scan engine not to beep after a good decode. The beeper still operates during parameter menu scanning and indicates error conditions.



**Do Not Beep After Good Decode**  
(0x00)

## Transmit “No Read” Message

### ***Parameter # 0x5E***

Enable this option to transmit “NR” if a symbol does not decode during the timeout period or before the trigger is released. Any enabled prefix or suffixes are appended around this message.



**Enable No Read  
(0x01)**

When disabled, and a symbol cannot be decoded, no message is sent to the host.



**\*Disable No Read  
(0x00)**

## Parameter Scanning

### ***Parameter # 0xEC***

To disable decoding of parameter bar codes, scan the bar code below. The **Set Defaults** parameter bar code can still be decoded. To enable decoding of parameter bar codes, either scan **Enable Parameter Scanning** below, **Set Factory Defaults** on [page 8-7](#) or set this parameter to 0x01 via a serial command.



**\*Enable Parameter Scanning  
(0x01)**



**Disable Parameter Scanning  
(0x00)**

## Linear Code Type Security Level

### ***Parameter # 0x4E***

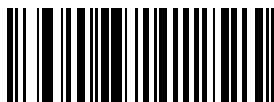
The SE955 offers four levels of decode security for linear code types (e.g. Code 39, Interleaved 2 of 5). Select higher security levels for decreasing levels of bar code quality. As security levels increase, the scan engine's aggressiveness decreases.

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	4 or less
D 2 of 5	8 or less
I 2 of 5	8 or less



**\*Linear Security Level 1  
(0x01)**

### ***Linear Security Level 2***

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



**Linear Security Level 2  
(0x02)**

### ***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:

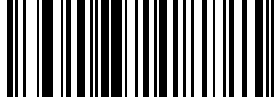
Code Type	Length
MSI	4 or less
D 2 of 5	8 or less
I 2 of 5	8 or less



**Linear Security Level 3  
(0x03)**

### ***Linear Security Level 4***

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



**Linear Security Level 4  
(0x04)**

## **Bi-directional Redundancy**

### ***Parameter # 0x43***

This parameter is only valid when a [Linear Code Type Security Level](#) is enabled (see [page 8-16](#)). When this parameter is enabled, a bar code must be successfully scanned in both directions (forward and reverse) before being decoded.



**Enable Bi-directional Redundancy  
(0x01)**



**\*Disable Bi-directional Redundancy  
(0x00)**

## UPC/EAN

### *Enable/Disable UPC-A*

#### *Parameter # 0x01*

To enable or disable UPC-A, scan the appropriate bar code below.



**\*Enable UPC-A  
(0x01)**



**Disable UPC-A  
(0x00)**

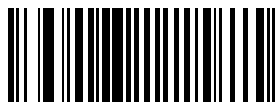
### *Enable/Disable UPC-E*

#### *Parameter # 0x02*

To enable or disable UPC-E, scan the appropriate bar code below.



**\*Enable UPC-E  
(0x01)**



**Disable UPC-E  
(0x00)**



## ***Enable/Disable UPC-E1***

### ***Parameter # 0x0C***

To enable or disable UPC-E1, scan the appropriate bar code below.



UPC-E1 is not a UCC (Uniform Code Council) approved symbology.



**Enable UPC-E1  
(0x01)**

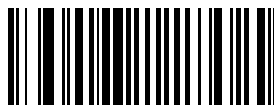


**\*Disable UPC-E1  
(0x00)**

## ***Enable/Disable EAN-8***

### ***Parameter # 0x04***

To enable or disable EAN-8, scan the appropriate bar code below.



**\*Enable EAN-8  
(0x01)**

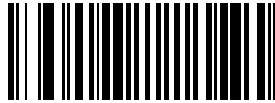


**Disable EAN-8  
(0x00)**

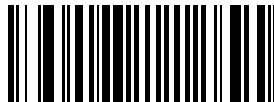
## ***Enable/Disable EAN-13***

### ***Parameter # 0x03***

To enable or disable EAN-13, scan the appropriate bar code below.



**\*Enable EAN-13  
(0x01)**



**Disable EAN-13  
(0x00)**

## ***Enable/Disable Bookland EAN***

### ***Parameter # 0x53***

To enable or disable EAN Bookland, scan the appropriate bar code below.



**Enable Bookland EAN  
(0x01)**



**\*Disable Bookland EAN  
(0x00)**

## Decode UPC/EAN Supplementals

### Parameter # 0x10

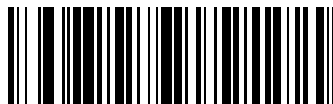
Supplementals are appended characters (2 or 5) according to specific code format conventions (e.g., UPC A+2, UPC E+2). Several options are available:

- If **Decode UPC/EAN with Supplemental** characters is selected, the scan engine does not decode UPC/EAN symbols without supplemental characters.
- If **Ignore UPC/EAN with Supplemental** characters is selected, and the SE955 is presented with a UPC/EAN symbol with a supplemental, the scan engine decodes the UPC/EAN and ignores the supplemental characters.
- If **Autodiscriminate UPC/EAN Supplementals** is selected, scan [Decode UPC/EAN Supplemental Redundancy on page 8-22](#), then select a value from the numeric bar codes beginning on [page 8-73](#). A value of 5 or more is recommended.
- Select **Enable 378/379 Supplemental Mode** to enable the SE955 to identify supplementals for EAN-13 bar codes starting with a '378' or '379' prefix only. All other UPC/EAN bar codes are decoded immediately and the supplemental characters ignored.
- Select **Enable 978 Supplemental Mode** to enable the SE955 to identify supplementals for EAN-13 bar codes starting with a '978' prefix only. All other UPC/EAN bar codes are decoded immediately and the supplemental characters ignored.
- Select **Enable Smart Supplemental Mode** to enable the SE955 to identify supplementals for EAN-13 bar codes starting with a '378', '379', or '978' prefix only. All other UPC/EAN bar codes are decoded immediately and the supplemental characters ignored.

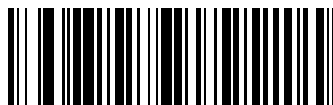


To minimize the risk of invalid data transmission, we recommend selecting whether to read or ignore supplemental characters.

Select the desired option by scanning one of the following bar codes.

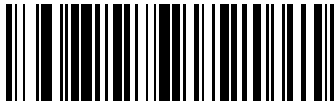


**Decode UPC/EAN With Supplementals**  
(0x01)

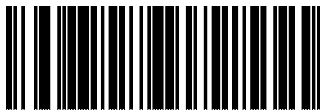


**\*Ignore UPC/EAN With Supplementals**  
(0x00)

## Decode UPC/EAN Supplementals (continued)



**Autodiscriminate UPC/EAN Supplementals  
(0x02)**



**Enable 378/379 Supplemental Mode  
(0x04)**



**Enable 978 Supplemental Mode  
(0x05)**



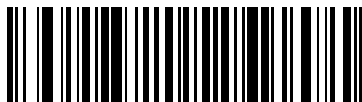
**Enable Smart Supplemental Mode  
(0x03)**

## Decode UPC/EAN Supplemental Redundancy

### Parameter # 0x50

With *Autodiscriminate UPC/EAN Supplementals* selected, this option adjusts the number of times a symbol without supplementals are decoded before transmission. The range is from 2 to 30 times. Five or above is recommended when decoding a mix of UPC/EAN symbols with and without supplementals, and the autodiscriminate option is selected.

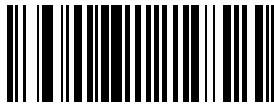
Scan the bar code below to select a decode redundancy value. Next scan two numeric bar codes beginning on [page 8-73](#). Single digit numbers must have a leading zero. To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



**Decode UPC/EAN  
Supplemental Redundancy  
(Default: 7)**

***Transmit UPC-A Check Digit******Parameter # 0x28***

Scan the appropriate bar code below to transmit the symbol with or without the UPC-A check digit.



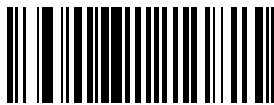
**\*Transmit UPC-A Check Digit  
(0x01)**



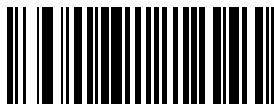
**Do Not Transmit UPC-A Check Digit  
(0x00)**

***Transmit UPC-E Check Digit******Parameter # 0x29***

Scan the appropriate bar code below to transmit the symbol with or without the UPC-E check digit.



**\*Transmit UPC-E Check Digit  
(0x01)**



**Do Not Transmit UPC-E Check Digit  
(0x00)**

## ***Transmit UPC-E1 Check Digit***

### ***Parameter # 0x2A***

Scan the appropriate bar code below to transmit the symbol with or without the UPC-E1 check digit.



**\*Transmit UPC-E1 Check Digit  
(0x01)**



**Do Not Transmit UPC-E1 Check Digit  
(0x00)**

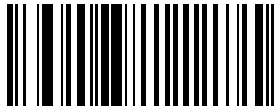
## **UPC-A Preamble**

### **Parameter # 0x22**

Preamble characters (Country Code and System Character) can be transmitted as part of a UPC-A symbol. Select one of the following options for transmitting UPC-A preamble to the host device: transmit system character only, transmit system character and country code ("0" for USA), or transmit no preamble.



**No Preamble**  
(<DATA>)  
(0x00)



**\*System Character**  
(<SYSTEM CHARACTER> <DATA>)  
(0x01)



**System Character & Country Code**  
(< COUNTRY CODE> <SYSTEM CHARACTER> <DATA>)  
(0x02)

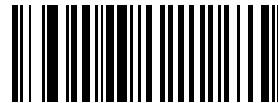
## **UPC-E Preamble**

### **Parameter # 0x23**

Preamble characters (Country Code and System Character) can be transmitted as part of a UPC-E symbol. Select one of the following options for transmitting UPC-E preamble to the host device: transmit system character only, transmit system character and country code ("0" for USA), or transmit no preamble.



**No Preamble**  
(<DATA>)  
(0x00)



**\*System Character**  
(<SYSTEM CHARACTER> <DATA>)  
(0x01)



**System Character & Country Code**  
(< COUNTRY CODE> <SYSTEM CHARACTER> <DATA>)  
(0x02)



## **UPC-E1 Preamble**

### **Parameter # 0x24**

Preamble characters (Country Code and System Character) can be transmitted as part of a UPC-E1 symbol. Select one of the following options for transmitting UPC-E1 preamble to the host device: transmit system character only, transmit system character and country code ("0" for USA), or transmit no preamble.



**No Preamble**  
(<DATA>)  
(0x00)



**\*System Character**  
(<SYSTEM CHARACTER> <DATA>)  
(0x01)



**System Character & Country Code**  
(< COUNTRY CODE> <SYSTEM CHARACTER> <DATA>)  
(0x02)

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

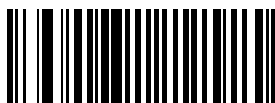
### ***Parameter # 0x25***

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

Scan **DO NOT CONVERT UPC-E TO UPC-A** to transmit UPC-E (zero suppressed) decoded data.



**Convert UPC-E to UPC-A (Enable)**  
**(0x01)**



**\*Do Not Convert UPC-E to UPC-A (Disable)**  
**(0x00)**

## ***Convert UPC-E1 to UPC-A***

### ***Parameter # 0x26***

Enable this parameter to convert UPC-E1 (zero suppressed) decoded data to UPC-A format before transmission. After conversion, data follows UPC-A format and is affected by UPC-A programming selections (e.g., Preamble, Check Digit).

Scan **DO NOT CONVERT UPC-E TO UPC-A** to transmit UPC-E1 (zero suppressed) decoded data.



**Convert UPC-E1 to UPC-A (Enable)**  
**(0x01)**



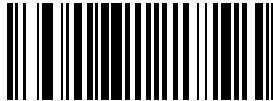
**\*Do Not Convert UPC-E1 to UPC-A (Disable)**  
**(0x00)**

## ***EAN Zero Extend***

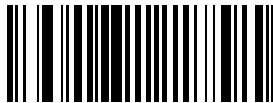
### ***Parameter # 0x27***

When enabled, this parameter adds five leading zeros to decoded EAN-8 symbols to make them compatible in format to EAN-13 symbols.

Disable this parameter to transmit EAN-8 symbols as is.



**Enable EAN Zero Extend  
(0x01)**



**\*Disable EAN Zero Extend  
(0x00)**

## ***Convert EAN-8 to EAN-13 Type***

### ***Parameter # 0xE0***

When EAN Zero Extend is enabled, you can label the extended symbol as either an EAN-13 bar code, or an EAN-8 bar code. This affects [Transmit Code ID Character](#) and [DECODE\\_DATA](#) message.

When EAN Zero Extend is disabled, this parameter has no effect on bar code data.



**\*Type Is EAN-13  
(0x00)**



**Type Is EAN-8  
(0x01)**

## **UPC/EAN Security Level**

### **Parameter # 0x4D**

The SE955 offers four levels of decode security for UPC/EAN bar codes. Increasing levels of security are provided for decreasing levels of bar code quality. Select higher levels of security for decreasing levels of bar code quality. Increasing security decreases the scan engine's aggressiveness, so choose only that level of security necessary for the application.

#### **UPC/EAN Security Level 0**

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



**\*UPC/EAN Security Level 0**  
(0x00)

#### **UPC/EAN Security Level 1**

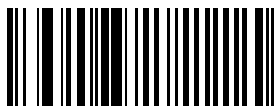
As bar code quality levels diminish, certain characters become prone to mis-decodes before others (i.e., 1, 2, 7, 8). If mis-decodes of poorly printed bar codes occur, and the mis-decodes are limited to these characters, select this security level.



**UPC/EAN Security Level 1**  
(0x01)

#### **UPC/EAN Security Level 2**

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



**UPC/EAN Security Level 2**  
(0x02)

#### **UPC/EAN Security Level 3**

If misdecodes still occur after selecting Security Level 2, 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 scan engine. If this level of security is necessary, try to improve the quality of the bar codes.



**UPC/EAN Security Level 3**  
(0x03)

## ***UCC Coupon Extended Code***

### ***Parameter # 0x55***

The UCC Coupon Extended Code is an additional bar code adjacent to a UCC Coupon Code. To enable or disable UCC Coupon Extended Code, scan the appropriate bar code below.



**Enable UCC Coupon Extended Code  
(0x01)**



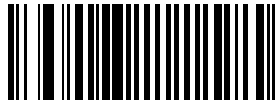
**\*Disable UCC Coupon Extended Code  
(0x00)**

## Code 128

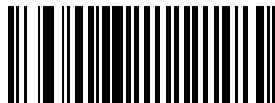
### ***Enable/Disable Code 128***

#### ***Parameter # 0x08***

To enable or disable Code 128, scan the appropriate bar code below.



**\*Enable Code 128  
(0x01)**

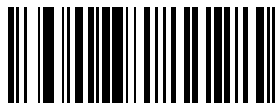


**Disable Code 128  
(0x00)**

### ***Enable/Disable UCC/EAN-128***

#### ***Parameter # 0x0E***

To enable or disable UCC/EAN-128, scan the appropriate bar code below. (See [Appendix B, Miscellaneous Code Information](#) for details on [UCC/EAN-128](#).)



**\*Enable UCC/EAN-128  
(0x01)**



**Disable UCC/EAN-128  
(0x00)**

## ***Enable/Disable ISBT 128***

### ***Parameter # 0x54***

To enable or disable ISBT 128, scan the appropriate bar code below.



**\*Enable ISBT 128  
(0x01)**



**Disable ISBT 128  
(0x00)**

### ***Lengths for Code 128***

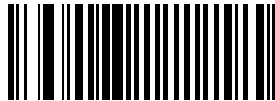
No length setting is required for Code 128.

## Code 39

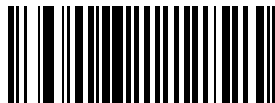
### *Enable/Disable Code 39*

#### *Parameter # 0x00*

To enable or disable Code 39, scan the appropriate bar code below.



**\*Enable Code 39  
(0x01)**



**Disable Code 39  
(0x00)**

### *Enable/Disable Trioptic Code 39*

#### *Parameter # 0x0D*

Trioptic Code 39 is a variant of Code 39 used in marking computer tape cartridges. Trioptic Code 39 symbols always contain six characters.

To enable or disable Trioptic Code 39, scan the appropriate bar code below.



**Enable Trioptic Code 39  
(0x01)**



**\*Disable Trioptic Code 39  
(0x00)**



Note

Trioptic Code 39 and Code 39 Full ASCII cannot be enabled simultaneously. If an error beep sounds when enabling Trioptic Code 39, disable Code 39 Full ASCII and try again.



## ***Convert Code 39 to Code 32 (Italian Pharma Code)***

### ***Parameter # 0x56***

Code 32 is a variant of Code 39 used by the Italian pharmaceutical industry. Scan the appropriate bar code below to enable or disable converting Code 39 to Code 32.



Code 39 must be enabled in order for this parameter to function.



**Enable Convert Code 39 to Code 32  
(0x01)**



**\*Disable Convert Code 39 to Code 32  
(0x00)**

## ***Code 32 Prefix***

### ***Parameter # 0xE7***

Enable this parameter to add the prefix character "A" to all Code 32 bar codes. Convert Code 39 to Code 32 (Italian Pharma Code) must be enabled for this parameter to function.



**Enable Code 32 Prefix  
(0x01)**



**\*Disable Code 32 Prefix  
(0x00)**

## Set Lengths for Code 39

### Parameter # L1 = 0x12, L2 = 0x13

The length of a code refers to the number of characters (i.e., human readable characters), including check digit(s) the code contains. Lengths for Code 39 may be set for any length, one or two discrete lengths, or lengths within a specific range. If Code 39 Full ASCII is enabled, **Length Within a Range** or **Any Length** are the preferred options. To set lengths via serial commands, see [Setting Code Lengths Via Serial Commands](#) on page B-6.



When setting lengths, single digit numbers must always be preceded by a leading zero.

**One Discrete Length** - This option limits decodes to only those Code 39 symbols containing a selected length. Lengths are selected from the numeric bar codes beginning on [page 8-73](#). For example, to decode only Code 39 symbols with 14 characters, scan **Code 39 - One Discrete Length**, then scan **1** followed by **4**. To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



Code 39 - One Discrete Length

**Two Discrete Lengths** - This option limits decodes to only those Code 39 symbols containing either of two selected lengths. Lengths are selected from the numeric bar codes beginning on [page 8-73](#). For example, to decode only those Code 39 symbols containing either 2 or 14 characters, select **Code 39 - Two Discrete Lengths**, then scan **0, 2, 1**, and then **4**. To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



Code 39 - Two Discrete Lengths

**Length Within Range** - This option limits decodes to only those Code 39 symbols within a specified range. For example, to decode Code 39 symbols containing between 4 and 12 characters, first scan **Code 39 - Length Within Range**. Then scan **0, 4, 1**, and **2**. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



Code 39 - Length Within Range

**Any Length** - Scan this option to decode Code 39 symbols containing any number of characters.

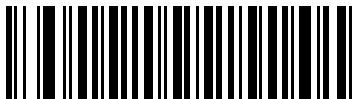


Code 39 - Any Length

## ***Code 39 Check Digit Verification***

### ***Parameter # 0x30***

When this feature is enabled, the scan engine checks the integrity of all Code 39 symbols to verify that the data complies with specified check digit algorithm. Only those Code 39 symbols which include a modulo 43 check digit are decoded. Only enable this feature if your Code 39 symbols contain a modulo 43 check digit.



**Verify Code 39 Check Digit  
(0x01)**



**\*Do Not Verify Code 39 Check Digit  
(0x00)**

## ***Transmit Code 39 Check Digit***

### ***Parameter # 0x2B***

Scan this symbol to transmit the check digit with the data.



**Transmit Code 39 Check Digit (Enable)  
(0x01)**

Scan this symbol to transmit data without the check digit.



**\*Do Not Transmit Code 39 Check Digit (Disable)  
(0x00)**

## ***Enable/Disable Code 39 Full ASCII***

### ***Parameter # 0x11***

Code 39 Full ASCII is a variant of Code 39 which pairs characters to encode the full ASCII character set. To enable or disable Code 39 Full ASCII, scan the appropriate bar code below.

Refer to Table B-3 on page B-5 for the mapping of Code 39 characters to ASCII values.



**Enable Code 39 Full ASCII  
(0x00)**



**\*Disable Code 39 Full ASCII  
(0x00)**



Note

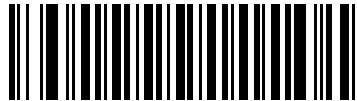
Trioptic Code 39 and Code 39 Full ASCII cannot be enabled simultaneously. If you get an error beep when enabling Code 39 Full ASCII, disable Trioptic Code 39 and try again.

## Code 93

### *Enable/Disable Code 93*

#### *Parameter # 0x09*

To enable or disable Code 93, scan the appropriate bar code below.



**Enable Code 93  
(0x01)**



**\*Disable Code 93  
(0x00)**

## Set Lengths for Code 93

### Parameter # L1 = 0x1A, L2 = 0x1B

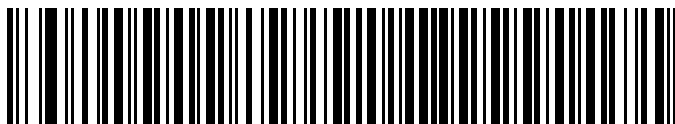
The length of a code refers to the number of characters (i.e., human readable characters), including check digit(s) the code contains. Lengths for Code 93 may be set for any length, one or two discrete lengths, or lengths within a specific range. To set lengths via serial commands, see [Setting Code Lengths Via Serial Commands](#) on page B-6.

**One Discrete Length** - Select this option to decode only those codes containing a selected length. For example, select **Code 93 One Discrete Length**, then scan **1, 4**, to limit the decoding to only Code 93 symbols containing 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



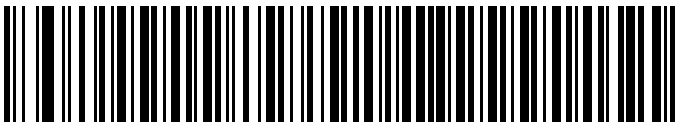
Code 93 - One Discrete Length

**Two Discrete Lengths** - Select this option to decode only those codes containing two selected lengths. For example, select **Code 93 Two Discrete Lengths**, then scan **0, 2, 1, 4**, to limit the decoding to only Code 93 symbols containing 2 or 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



Code 93 - Two Discrete Lengths

**Length Within Range** - This option sets the unit to decode a code type within a specified range. For example, to decode Code 93 symbols containing between 4 and 12 characters, first scan **Code 93 Length Within Range**, then scan **0, 4, 1** and **2** (single digit numbers must always be preceded by a leading zero). Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



Code 93 - Length Within Range

**Any Length** - Scan this option to decode Code 93 symbols containing any number of characters.



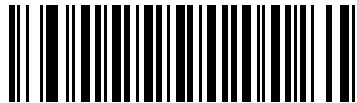
Code 93 - Any Length

## Code 11

### Enable/Disable Code 11

#### Parameter # 0x0A

To enable or disable Code 11, scan the appropriate bar code below.



**Enable Code 11**  
(0x01)



**\*Disable Code 11**  
(0x00)

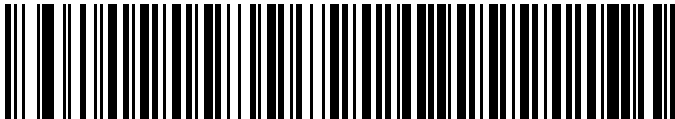
### Set Lengths for Code 11

#### Parameter # L1 = 0x1C, L2 = 0x1D

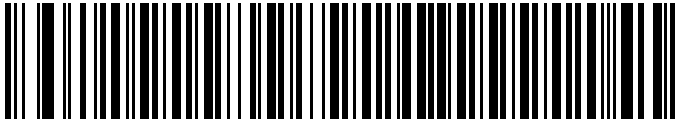
The length of a code refers to the number of characters (i.e., human readable characters), including check digit(s) the code contains. Set lengths for Code 11 to any length, one or two discrete lengths, or lengths within a specific range.

- **One Discrete Length** - Select this option to decode only Code 11 symbols containing a selected length. Select the length using the numeric bar codes beginning on [page 8-73](#). For example, to decode only Code 11 symbols with 14 characters, scan **Code 11 - One Discrete Length**, then scan **1** followed by **4**. To correct an error or to change the selection, scan the [Cancel](#) bar code on [page 8-74](#).
- **Two Discrete Lengths** - Select this option to decode only Code 11 symbols containing either of two selected lengths. Select lengths using the numeric bar codes beginning on [page 8-73](#). For example, to decode only those Code 11 symbols containing either 2 or 14 characters, select **Code 11 - Two Discrete Lengths**, then scan **0, 2, 1**, and then **4**. To correct an error or to change the selection, scan the [Cancel](#) bar code on [page 8-74](#).
- **Length Within Range** - Select this option to decode a Code 11 symbol with a specific length range. Select lengths using numeric bar codes beginning on [page 8-73](#). For example, to decode Code 11 symbols containing between 4 and 12 characters, first scan **Code 11 - Length Within Range**. Then scan **0, 4, 1**, and **2** (single digit numbers must always be preceded by a leading zero). To correct an error or change the selection, scan the [Cancel](#) bar code on [page 8-74](#).
- **Any Length** - Scan this option to decode Code 11 symbols containing any number of characters within the scan engine capability.

***Set Lengths for Code 11 (continued)***



**Code 11 - One Discrete Length**



**Code 11 - Two Discrete Lengths**



**Code 11 - Length Within Range**



**Code 11 - Any Length**



## Code 11 Check Digit Verification

### Parameter # 0x34

This feature allows the scan engine to check the integrity of all Code 11 symbols to verify that the data complies with the specified check digit algorithm. This selects the check digit mechanism for the decoded Code 11 bar code. The options are to check for one check digit, check for two check digits, or disable the feature.

To enable this feature, scan the bar code below corresponding to the number of check digits encoded in your Code 11 symbols.



**\*Disable  
(0x00)**



**One Check Digit  
(0x01)**



**Two Check Digits  
(0x02)**

## Transmit Code 11 Check Digits

### Parameter # 0x2F

This feature selects whether or not to transmit the Code 11 check digit(s).



**Transmit Code 11 Check Digit(s) (Enable)  
(0x01)**



**\*Do Not Transmit Code 11 Check Digit(s) (Disable)  
(0x00)**



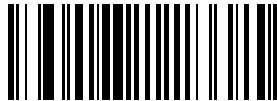
Code 11 Check Digit Verification must be enabled for this parameter to function.

## Interleaved 2 of 5

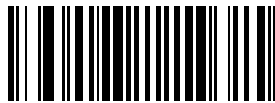
### *Enable/Disable Interleaved 2 of 5*

#### ***Parameter # 0x06***

To enable or disable Interleaved 2 of 5, scan the appropriate bar code below.



**\*Enable Interleaved 2 of 5  
(0x01)**



**Disable Interleaved 2 of 5  
(0x00)**

## Set Lengths for Interleaved 2 of 5

### Parameter # L1 = 0x16, L2 = 0x17

The length of a code refers to the number of characters (i.e., human readable characters), including check digit(s) the code contains. Lengths for I 2 of 5 may be set for any length, one or two discrete lengths, or lengths within a specific range. To set lengths via serial commands, see *Setting Code Lengths Via Serial Commands* on page B-8.



When setting lengths, single digit numbers must always be preceded by a leading zero.

**One Discrete Length** - Select this option to decode only those codes containing a selected length. For example, select **I 2 of 5 One Discrete Length**, then scan **1, 4**, to decode only I 2 of 5 symbols containing 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



**I 2 of 5 - One Discrete Length**

**Two Discrete Lengths** - Select this option to decode only those codes containing two selected lengths. For example, select **I 2 of 5 Two Discrete Lengths**, then scan **0, 6, 1, 4**, to decode only I 2 of 5 symbols containing 6 or 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).



**I 2 of 5 - Two Discrete Lengths**

## Set Lengths for Interleaved 2 of 5 (continued)

**Length Within Range** - Select this option to decode only codes within a specified range. For example, to decode I 2 of 5 symbols containing between 4 and 12 characters, first scan **I 2 of 5 Length Within Range**, then scan **0, 4, 1** and **2** (single digit numbers must always be preceded by a leading zero). Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



**I 2 of 5 - Length Within Range**

**Any Length** - Scan this option to decode I 2 of 5 symbols containing any number of characters.



Note

Selecting this option may lead to misdecodes for I 2 of 5 codes.



**I 2 of 5 - Any Length**

## ***1 2 of 5 Check Digit Verification***

### ***Parameter # 0x31***

When enabled, this parameter checks the integrity of an 1 2 of 5 symbol to ensure it complies with a specified algorithm, either USS (Uniform Symbology Specification), or OPCC (Optical Product Code Council).



**\*Disable  
(0x00)**



**USS Check Digit  
(0x01)**



**OPCC Check Digit  
(0x02)**

## ***Transmit I 2 of 5 Check Digit***

### ***Parameter # 0x2C***

Scan this symbol to transmit the check digit with the data.



**Transmit I 2 of 5 Check Digit (Enable)  
(0x01)**

Scan this symbol to transmit data without the check digit.



**\*Do Not Transmit I 2 of 5 Check Digit (Disable)  
(0x00)**

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

### ***Parameter # 0x52***

This parameter converts a 14 character I 2 of 5 code into EAN-13, and transmits to the host as EAN-13. To accomplish this, I 2 of 5 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.



**Convert I 2 of 5 to EAN-13 (Enable)  
(0x01)**



**\*Do Not Convert I 2 of 5 to EAN-13 (Disable)  
(0x00)**

## Discrete 2 of 5

### *Enable/Disable Discrete 2 of 5*

#### *Parameter # 0x05*

To enable or disable Discrete 2 of 5, scan the appropriate bar code below.



**Enable Discrete 2 of 5  
(0x01)**



**\*Disable Discrete 2 of 5  
(0x00)**

## Set Lengths for Discrete 2 of 5

### Parameter # L1 = 0x14, L2 = 0x15

The length of a code refers to the number of characters (i.e., human readable characters), including check digit(s) the code contains. Lengths for D 2 of 5 may be set for any length, one or two discrete lengths, or lengths within a specific range. To set lengths via serial commands, see *Setting Code Lengths Via Serial Commands* on page B-8.

**One Discrete Length** - Select this option to decode only those codes containing a selected length. For example, select **D 2 of 5 One Discrete Length**, then scan **1, 4**, to decode only D 2 of 5 symbols containing 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



D 2 of 5 - One Discrete Length

**Two Discrete Lengths** - Select this option to decode only those codes containing two selected lengths. For example, select **D 2 of 5 Two Discrete Lengths**, then scan **0, 2, 1, 4**, to decode only D 2 of 5 symbols containing 2 or 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



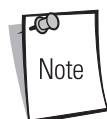
D 2 of 5 - Two Discrete Lengths

**Length Within Range** - Select this option to decode codes within a specified range. For example, to decode D 2 of 5 symbols containing between 4 and 12 characters, first scan **D 2 of 5 Length Within Range**, then scan **0, 4, 1** and **2** (single digit numbers must be preceded by a leading zero). Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



D 2 of 5 - Length Within Range

**Any Length** - Scan this option to decode D 2 of 5 symbols containing any number of characters.



Selecting this option may lead to misdecodes for D 2 of 5 codes.



D 2 of 5 - Any Length



## Chinese 2 of 5

### *Enable/Disable Chinese 2 of 5*

#### ***Parameter # 0xF0 0x98***

To enable or disable Chinese 2 of 5, scan the appropriate bar code below.



**Enable Chinese 2 of 5  
(0x01)**



**\*Disable Chinese 2 of 5  
(0x00)**

## Codabar

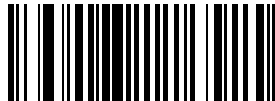
### *Enable/Disable Codabar*

#### *Parameter # 0x07*

To enable or disable Codabar, scan the appropriate bar code below.



**Enable Codabar  
(0x01)**



**\*Disable Codabar  
(0x00)**

## Set Lengths for Codabar

### Parameter # L1 = 0x18, L2 = 0x19

The length of a code refers to the number of characters (i.e., human readable characters), including check digit(s) the code contains. Lengths for Codabar may be set for any length, one or two discrete lengths, or lengths within a specific range. To set lengths via serial commands, see *Setting Code Lengths Via Serial Commands* on page B-8.

**One Discrete Length** - Select this option to decode only those codes containing a selected length. For example, select **Codabar One Discrete Length**, then scan **1, 4**, to decode only Codabar symbols containing 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



**Codabar - One Discrete Length**

**Two Discrete Lengths** - This option sets the unit to decode only those codes containing two selected lengths. For example, select **Codabar Two Discrete Lengths**, then scan **0, 2, 1, 4**, to decode only Codabar symbols containing 6 or 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



**Codabar - Two Discrete Lengths**

**Length Within Range** - Select this option to decode a code within a specified range. For example, to decode Codabar symbols containing between 4 and 12 characters, first scan **Codabar Length Within Range**, then scan **0, 4, 1** and **2** (single digit numbers must always be preceded by a leading zero). Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



**Codabar - Length Within Range**

**Any Length** - Scan this option to decode Codabar symbols containing any number of characters.



**Codabar - Any Length**

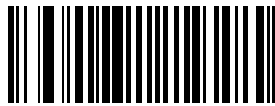
## ***CLSI Editing***

### ***Parameter # 0x36***

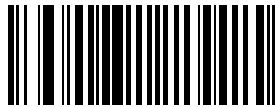
When enabled, this parameter strips the start and stop characters and inserts a space after the first, fifth, and tenth characters of a 14-character Codabar symbol.



Symbol length does not include start and stop characters.



**Enable CLSI Editing  
(0x01)**

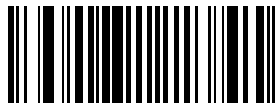


**\*Disable CLSI Editing  
(0x00)**

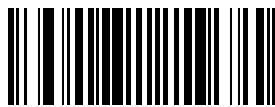
## ***NOTIS Editing***

### ***Parameter # 0x37***

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



**Enable NOTIS Editing  
(0x01)**



**\*Disable NOTIS Editing  
(0x00)**

## MSI

### *Enable/Disable MSI*

#### *Parameter # 0x0B*

To enable or disable MSI, scan the appropriate bar code below.



**Enable MSI**  
**(0x01)**



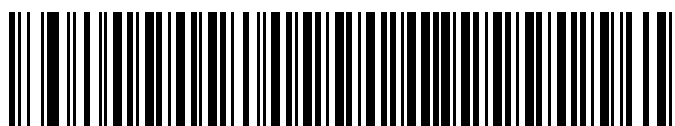
**\*Disable MSI**  
**(0x00)**

## Set Lengths for MSI

### Parameter # L1 = 0x1E, L2 = 0x1F

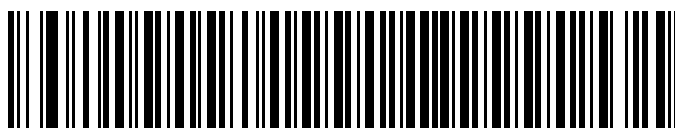
The length of a code refers to the number of characters (i.e., human readable characters) the code contains, and includes check digits. Lengths for MSI can be set for any length, one or two discrete lengths, or lengths within a specific range. See Table B-5 on page B-9 for ASCII equivalents. To set lengths via serial commands, see *Setting Code Lengths Via Serial Commands* on page B-8.

**One Discrete Length** - Select this option to decode only those codes containing a selected length. For example, select **MSI Plessey One Discrete Length**, then scan **1, 4**, to decode only MSI Plessey symbols containing 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



MSI - One Discrete Length

**Two Discrete Lengths** - Select this option to decode only those codes containing two selected lengths. For example, select **MSI Plessey Two Discrete Lengths**, then scan **0, 6, 1, 4**, to decode only MSI Plessey symbols containing 6 or 14 characters. Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



MSI - Two Discrete Lengths

**Length Within Range** - Select this option to decode codes within a specified range. For example, to decode MSI symbols containing between 4 and 12 characters, first scan **MSI Length Within Range**, then scan **0, 4, 1** and **2** (single digit numbers must always be preceded by a leading zero). Numeric bar codes begin on [page 8-73](#). To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



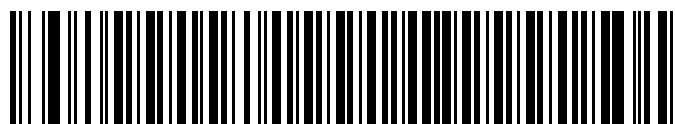
MSI - Length Within Range

**Any Length** - Scan this option to decode MSI Plessey symbols containing any number of characters.



Note

Selecting this option may lead to misdecodes for MSI codes.



MSI - Any Length

## **MSI Check Digits**

### **Parameter # 0x32**

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.



**\*One MSI Check Digit  
(0x00)**

If two check digits are selected, also select an [MSI Check Digit Algorithm on page 8-58](#).



**Two MSI Check Digit  
(0x01)**

## **Transmit MSI Check Digit**

### **Parameter # 0x2E**

Scan this symbol to transmit the check digit with the data.



**Transmit MSI Check Digit (Enable)  
(0x01)**

Scan this symbol to transmit data without the check digit.



**\*Do Not Transmit MSI Check Digit (Disable)  
(0x00)**

## **MSI Check Digit Algorithm**

### **Parameter # 0x33**

When the Two MSI check digits option is selected, an additional verification is required to ensure integrity. Select one of the following algorithms.



**MOD 10/ MOD 11**  
**(0x00)**



**\*MOD 10/ MOD 10**  
**(0x01)**

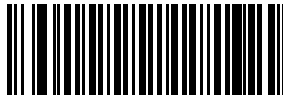


## RSS

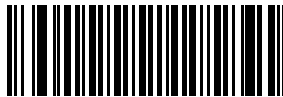
### ***Enable/Disable RSS-14***

#### ***Parameter # 0xF0 0x52***

To enable or disable RSS-14, scan the appropriate bar code below.



**Enable RSS-14  
(0x01)**



**\*Disable RSS-14  
(0x00)**

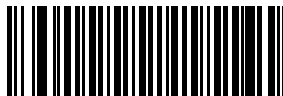
### ***Enable/Disable RSS-Limited***

#### ***Parameter # 0xF0 0x53***

To enable or disable RSS-Limited, scan the appropriate bar code below.



**Enable RSS-Limited  
(0x01)**

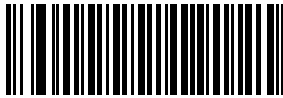


**\*Disable RSS-Limited  
(0x00)**

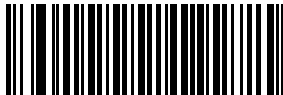
## ***Enable/Disable RSS-Expanded***

### ***Parameter # 0xF0 0x54***

To enable or disable RSS-Expanded, scan the appropriate bar code below.



**Enable RSS-Expanded  
(0x01)**



**\*Disable RSS-Expanded  
(0x00)**

## ***Convert RSS to UPC/EAN***

### ***Parameter # 0xF0 0x8D***

This parameter only applies to RSS-14 and RSS Limited symbols. When this conversion is enabled, RSS-14 and RSS Limited symbols encoding a single zero as the first digit have the leading '010' stripped and the bar code reported as EAN-13.

Bar codes beginning with two or more zeros but not six zeros have the leading '0100' stripped and the bar code reported as UPC-A. The UPC-A Preamble parameter to transmit the system character and country code applies to converted bar codes. Note that neither the system character nor the check digit can be stripped.



**Enable Convert RSS to UPC/EAN**



**\*Disable Convert RSS to UPC/EAN**

## Transmit Code ID Character

### ***Parameter # 0x2D***

A code ID character identifies the code type of a scanned bar code. This can be useful when decoding more than one code type. The code ID character is inserted between the prefix character (if selected) and the decoded symbol.

Select no code ID character, a Symbol Code ID character, or an AIM Code ID character. The Symbol Code ID characters are listed below; see [Appendix B, Miscellaneous Code Information](#) for [AIM Code Identifiers](#).

- A = UPC-A, UPC-E, UPC-E1, EAN-8, EAN-13
- B = Code 39, Code 32
- C = Codabar
- D = Code 128, ISBT 128
- E = Code 93
- F = Interleaved 2 of 5
- G = Discrete 2 of 5
- J = MSI
- K = UCC/EAN-128
- L = Bookland EAN
- M = Trioptic Code 39
- N = Coupon Code
- R = RSS-14, RSS-Limited, RSS-Expanded



**Symbol Code ID Character  
(0x02)**



**Aim Code ID Character  
(0x01)**



**\*None  
(0x00)**

## Prefix/Suffix Values

### **Parameter # P = 0x69, S1 = 0x68, S2 = 0x6A**

A prefix and/or one or two suffixes can be appended to scan data for use in data editing. To set these values, scan a four-digit number (i.e. four bar codes) that corresponds to ASCII values. See the [Table B-5 on page B-7](#) and [Numeric Bar Codes on page 8-73](#). To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#). To set the Prefix/Suffix values via serial commands, see [Setting Prefixes and Suffixes Via Serial Commands on page B-7](#).



In order to use Prefix/Suffix values, the [Scan Data Transmission Format on page 8-63](#) must be set. See [page 8-62](#).



**Scan Prefix**



**Scan Suffix 1**



**Scan Suffix 2**



**Data Format Cancel**

## Scan Data Transmission Format

### ***Parameter # 0xEB***

To change the Scan Data Transmission Format, scan one of the eight bar codes corresponding to the desired format.



**\*Data As Is  
(0x00)**



**<DATA> <SUFFIX 1>  
(0x01)**



**<DATA> <SUFFIX 2>  
(0x02)**



**<DATA> <SUFFIX 1> <SUFFIX 2>  
(0x03)**



**<PREFIX> <DATA>  
(0x04)**

**Scan Data Transmission Format (continued)**



<PREFIX> <DATA> <SUFFIX 1>  
(0x05)



<PREFIX> <DATA> <SUFFIX 2>  
(0x06)



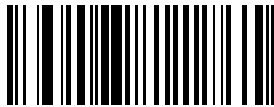
<PREFIX> <DATA> <SUFFIX 1> <SUFFIX 2>  
(0x07)

## Serial Parameters

### ***Baud Rate***

#### ***Parameter # 0x9C***

Baud rate is the number of bits of data transmitted per second. The scan engine's baud rate setting should match the data rate setting of the host device. If not, data may not reach the host device or may reach it in distorted form.



**Baud Rate 300  
(0x01)**



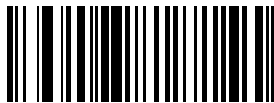
**Baud Rate 600  
(0x02)**



**Baud Rate 1200  
(0x03)**

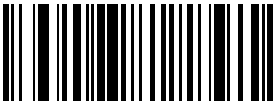


**Baud Rate 2400  
(0x04)**



**Baud Rate 4800  
(0x05)**

**Baud Rate (continued)**



**\*Baud Rate 9600  
(0x06)**



**Baud Rate 19,200  
(0x07)**



**Baud Rate 38,400  
(0x08)**



## Parity

### Parameter # 0x9E

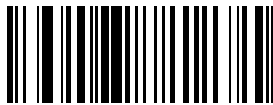
A parity check bit is the most significant bit of each ASCII coded character. Select the parity type according to host device requirements.

If you select **ODD** parity, the parity bit has a value 0 or 1, based on data, to ensure than an odd number of 1 bits is contained in the coded character.



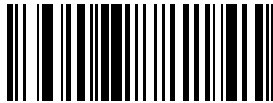
**Odd**  
(0x00)

If you select **EVEN** parity, the parity bit has a value 0 or 1, based on data, to ensure than an even number of 1 bits is contained in the coded character.



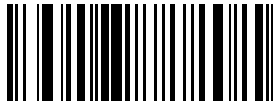
**Even**  
(0x01)

Select **MARK** parity and the parity bit is always 1.



**Mark**  
(0x02)

Select **SPACE** parity and the parity bit is always 0.



**Space**  
(0x03)

If no parity is required, select **NONE**.



**\*None**  
(0x04)

## **Software Handshaking**

### **Parameter # 0x9F**

This parameter offers control of the data transmission process in addition to that offered by hardware handshaking. Hardware handshaking is always enabled and cannot be disabled by the user.

#### **Disable ACK/NAK Handshaking**

When this option is selected, the decoder neither generates nor expects ACK/NAK handshaking packets.



**Disable ACK/NAK**  
**(0x00)**

#### **Enable ACK/NAK Handshaking**

When this option is selected, after transmitting data, the scan engine expects either an ACK or NAK response from the host. The scan engine also ACKs or NAKs messages from the host.

The scan engine waits up to the programmable Host Serial Response Time-out to receive an ACK or NAK. If the scan engine does not get a response in this time, it resends its data up to two times before discarding the data and declaring a transmit error.



**\*Enable ACK/NAK**  
**(0x01)**

## ***Decode Data Packet Format***

### ***Parameter # 0xEE***

This parameter selects whether decoded data is transmitted in raw format (unpacketed), or transmitted with the packet format as defined by the serial protocol.

If the raw format is selected, ACK/NAK handshaking is disabled for decode data.



**\*Send Raw Decode Data  
(0x00)**



**Send Packeted Decode Data  
(0x01)**

## ***Host Serial Response Time-out***

### ***Parameter # 0x9B***

This parameter specifies how long the decoder waits for an ACK or NAK before resending. Also, if the decoder wants to send, and the host has already been granted permission to send, the decoder waits for the designated time-out before declaring an error.

The delay period can range from 0.0 to 9.9 seconds in 0.1 second increments. After scanning the bar code below, scan two numeric bar codes beginning on [page 8-73](#). Values less than 10 require a leading zero. To change the selection or cancel an incorrect entry, scan the [Cancel](#) bar code on [page 8-74](#).

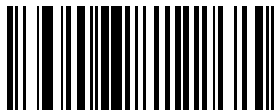


**Host Serial Response Time-out  
(Default: 2.0 sec.)**

## Stop Bit Select

### Parameter # 0x9D

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. Set the number of stop bits (one or two) to match host device requirements.



**\*1 Stop Bit  
(0x01)**



**2 Stop Bits  
(0x02)**

## Intercharacter Delay

### Parameter # 0x6E

The intercharacter delay gives the host system time to service its receiver and perform other tasks between characters. Select the intercharacter delay option matching host requirements. The delay period can range from no delay to 99 msec in 1 msec increments. After scanning the bar code below, scan two bar codes beginning on [page 8-71](#) to set the desired time-out. To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



**Intercharacter Delay  
(Default: 0 sec.)**

## Host Character Time-out

### Parameter # 0xEF

This parameter determines the maximum time the decoder waits between characters transmitted by the host before discarding the received data and declaring an error. The time-out is set in 0.01 second increments from 0.01 seconds to 0.99 seconds. After scanning the bar code below, scan two bar codes beginning on [page 8-71](#) to set the desired time-out. To change the selection or cancel an incorrect entry, scan the *Cancel* bar code on [page 8-74](#).



**Host Character Time-out  
(Default: 200 msec.)**

## Event Reporting

The host can request the decoder to furnish certain information (events) relative to the decoder's behavior. Enable or disable the events listed in [Table 8-2](#) by scanning the appropriate bar codes on the pages that follow. Parameter number format for these parameters follow those shown in [Table 10-9 on page 10-17](#) for parameters numbered 256 or higher.

**Table 8-2. Event Codes**

Event Class	Event	Code Reported
<b>Decode Event</b>	Non parameter decode	0x01
<b>Boot Up Event</b>	System power-up	0x03
<b>Parameter Event</b>	Parameter entry error	0x07
	Parameter stored	0x08
	Defaults set (and parameter event is enabled by default)	0x0A
	Number expected	0x0F

### Decode Event

#### Parameter # 0xF0 0x00

When enabled, the decoder generates a message to the host whenever a bar code is successfully decoded. When disabled, no notification is sent.



**Enable  
(0x01)**



**\*Disable  
(0x00)**

**Boot Up Event**

**Parameter # 0xF0 0x02**

When enabled, the decoder sends a message to the host whenever power is applied. When disabled, no message is sent.



**Enable**  
**(0x01)**



**\*Disable**  
**(0x00)**

**Parameter Event**

**Parameter # 0xF0 0x03**

When enabled, the decoder sends a message to the host when one of the events specified in [Table 8-2 on page 8-71](#) occurs. When disabled, no message is sent.



**Enable**  
**(0x01)**



**\*Disable**  
**(0x00)**

## Numeric Bar Codes

For parameters requiring specific numeric values, scan the appropriately numbered bar code(s).



0



1



2



3



4



5

**Numeric Bar Codes (continued)**



6



7



8



9

***Cancel***

To change the selection or cancel an incorrect entry, scan the bar code below.



**Cancel**



## ***Remote Scanner Management***

### **Introduction**

Symbol Technologies' Remote Scanner Management (RSM) technology enables a host to manage a Symbol scanner or scan engine. The scanner or scan engine can provide asset-tracking information (attributes) to the host such as serial number, date of manufacture and firmware version. In addition, Symbol's Remote Scanner Management technology provides the ability to automate the configuration process, monitor and optimize scan engine operation, and enables firmware upgrade to support new features.

The SE955 scan engine supports the ability to be remotely managed by supporting discovery, parameter configuration, and firmware updates electronically through Simple Serial Interface (SSI).

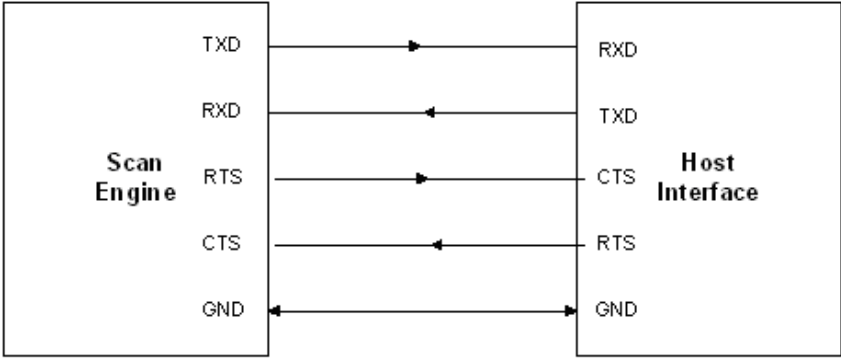
Discovery is defined as the ability to query "born on" information such as the model number, scan engine serial number, date of manufacture, and firmware version. This information can be retrieved electronically from the scan engine (through the SSI host interface) so that an application or Management Agent can publish the "discovered" information to the enterprise backend.

Parameter configuration allows the ability to query and set the device settings (i.e., scan amplitude) electronically.

For more information about SSI that supports the Remote Scanner Management architecture, see [\*Chapter 10, Simple Serial Interface\*](#).

# Hardware Signals

The basic SE955 scan engine & host interconnection diagram for the RSM transaction is shown in [Figure 9-1](#) below.



RSM Via RS232 - Scan Engine and Host Interconnection

**Figure 9-1. RSM via RS232 Interconnection**

# Protocol Commands

[Table 9-1](#) identifies the required support based upon Device Class.

**Table 9-1. Device Class**

Device Class	Supported Interfaces	Features
SE955 Scan Engine	RS232 - SSI	RMD_GET_PACKETSIZE ATTRIBUTE_GETALL ATTRIBUTE_GET ATTRIBUTE_GET_NEXT ATTRIBUTE_GET_OFFSET ATTRIBUTE_SET ATTRIBUTE_STORE

## Attribute Support

Table 9-2 identifies the list of attribute numbers supported in the SE955 scan engine.



Write access to the fields in this above RMD command list shall occur in supervisor mode (i.e., FAT or Manufacturing only).

**Table 9-2. Supported Attribute Numbers**

Attribute Number	Attribute Name	User Mode Access	Supervisor Mode Access	Size (Bytes)	Description
0 - 9999	System Parameters	R/W	R/W	Variable	All system parameters are accessible via 'GET' and 'SET' attribute commands.
191	Motor Scan Angle	R/W	R/W	1	Scan Angle setting that is currently in use by engine.
10008	Time to Decode	R	R	2	Time for the most recent decode is ms.
10009	Internal Temperature	R	R	2	Temperature of device - value is raw AD value.
10010	Motor Frequency	R	R	1	The motor frequency of the scan engine.
20000	Part Number	R	R/W	18	The part number of the device. This attribute shall exist for both the base and scan engine. This field shall be programmed during Manufacturing.  The format shall be: "LSXXXX-YYYY.Y"
20001	Serial Number	R	R/W	16	This is the serial number of the device. This attribute shall exist for both the base and scan engine. This attribute shall exist for both the base and scan engine. This field shall be programmed during Manufacturing.
20002	Date of Manufacture	R	R/W	7	The date of the manufacture DD- Day MMM- Month, Jan, Feb, Apr YY -Year This attribute shall exist for both the base and scan engine. This field shall be programmed during Manufacturing.
20003	Date of Last Service	R	R/W	7	The date of the manufacture DD- Day MMM- Month, Jan, Feb, Apr YY -Year This attribute shall exist for both the base and scan engine. This field shall be programmed during Manufacturing and the Service Depot.
20004	Firmware Version	R	R	8	This attribute represents the firmware version of the device. This attribute shall exist for both the base and scan engine. This field shall be determined at software build time.
20005	Engine ID	R	R/W	1	The engine ID of the attached scan engine.
20006	Hardware Version	R	R/W	1	The version or versions of the boards in the system.

## RMD Protocol Over RS232 (SSI)

### ***Encapsulation of RMD Commands/Responses over SSI***

The SSI protocol allows the host to send a command that is variable in length up to 255 bytes. Although there is a provision in the protocol to multi-packet commands from the host, it is not supported in the scan engine. It is required that the host fragment packets using the provisions supplied in the RMD protocol (ATTRIBUTE\_SET\_OFFSET, ATTRIBUTE\_GET\_OFFSET).

## Command Structure

Bit								
Byte	7	6	5	4	3	2	1	0
0	Length (Not including the checksum)							
1	SSI_MGMT_COMMAND (0x80)							
2	Message Source (4 - Host)							
3	Reserved (0)				Reserved (0)	Reserved (0)	Cont'd packet	Retransmit
4	Management Payload							
...								
...								
...								
...								
Length -1								
Length	2's compliment checksum (MSB)							
Length +1	2's compliment checksum (LSB)							

The expected response in the positive case is SSI\_MGMT\_COMMAND that may be a multi-packet response. For devices that do not support the SSI\_MGMT\_COMMAND, the response is the standard SSI\_NAK (NAK\_BADCONTEXT).

## Response Structure

Byte	7	6	5	4	3	2	1	0
0	Length (Not including the checksum)							
1	SSI_MGMT_COMMAND (0x80)							
2	Message Source (0 - Decoder)							
3	Reserved (0)			Reserved (0)		Reserved (0)	Cont'd packet	Retransmit
4	Management Payload							
...								
...								
...								
...								
Length -1								
Length	2's compliment checksum (MSB)							
Length +1	2's compliment checksum (LSB)							

## ***Example Transaction***

The following example shows how to retrieve the serial number (Attribute # 534 decimal) from the scanner.

### **Command from Host to Retrieve the Serial Number**

Entire command: 0A 80 04 00 00 06 02 00 02 16 FF 52

Where:

- 0A 80 04 00 - Encapsulation of RMD Commands/Responses over SSI command header
- 00 06 02 00 02 16 - Attribute Get command requesting attribute 534 decimal
- FF 52 - Encapsulation of RMD Commands/Responses over SSI command checksum

### **Response from Scanner with Serial Number**

Entire response: 23 80 00 00 00 1F 02 00 02 16 53 01 00 11 00 00 4D 31 4A 36 39 57 32 39 47 20 20 20 20 20 20 00 FF FF F9 A1

Where:

- 23 80 00 00 - Encapsulation of RMD Commands/Responses over SSI command header
- 00 1F 02 00 02 16 53 01 00 11 00 00 4D 31 4A 36 39 57 32 39 47 20 20 20 20 20 20 00 FF FF - Attribute get response which returns attribute 534 as a string response
- F9 A1 - Encapsulation of RMD Commands/Responses over SSI command checksum

## RSM Protocol (Management Payload)

The RMS protocol is defined as a host initiated command/response type protocol. Unsolicited responses are not currently addressed or supported by this protocol. The functionality of this protocol can be extended, if required, in the future.

The primitive commands supported by the SE955 RMS protocol are listed in [Table 9-3](#).

**Table 9-3. Supported RMS Protocol**

Command	Opcode
RMS_GET_PACKETSIZE	0x20
ATTRIBUTE_GETALL	0x01
ATTRIBUTE_GET	0x02
ATTRIBUTE_GET_NEXT	0x03
ATTRIBUTE_GET_OFFSET	0x04
ATTRIBUTE_SET	0x05
ATTRIBUTE_STORE	0x06

All Symbol scanning devices must support the commands prefixed with ATTRIBUTE\_ in order to be RMS compliant. Cascaded legacy devices (such as scan engines) may be compliant as long as they are attached to a root that is RMS compliant.

The commands prefixed with TUNNEL\_ are exclusively used for devices that fit the cascaded model (cordless multi-point scanners and table-top with auxiliary scanners).

### Error Commands

All RMS commands that fail receive the following response from the device.

### Response Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (MGMT_ERROR) (0x50)							
3	Status							

### Key

- Length: Length of the message including the length itself.
- Opcode
- Status:
  - UNKNOWN\_CMD (1)
  - TUNNEL\_ERROR (2)
  - RMS\_GET\_PACKETSIZE not received (3)

## ***RMS\_GET\_PACKETSIZE***

The RMS\_GET\_PACKETSIZE command allows the host to query packet size that is supported by the device. This command is required because each device has a different resource availability.



It is required that the host sends this command before it issues any other RMS command. If this command is not received, the host responds with an `MGMT_ERROR` message.

## Command Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (RMS_GET_PACKETSIZE) (0x20)							
3	Reserved (0)							
4	Maximum response size (MSB)							
5	Maximum response size (LSB)							

## Response Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (RMS_GET_PACKETSIZE) (0x20)							
3	Status							
4	Device Receive Packet Size (MSB)							
5	Device Receive Packet Size (LSB)							
6	Device Transmit Packet Size (MSB)							
7	Device Transmit Packet Size (LSB)							

**ATTRIBUTE\_GETALL**

The ATTRIBUTE\_GETALL command provides the ability to read out the supported attributes of the device being queried. The command itself does not report the values of the attributes but rather, the attribute numbers supported by the device itself. This command was added as an optimization for the host to report the "discoverable" device attributes.



The maximum command/response frame is limited to 240 bytes due to the memory constraints of the devices queried. If the response does not fit in the 240-byte response frame, then it is up to the application to parse the response and determine where the command should pick up. The command set has provisions that support this feature.

## Command Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GETALL) (0x01)							
3	Reserved (0)							
4	Start Attribute Number (MSB)							
5	Start Attribute Number (LSB)							

### **Key**

- Length: Length of the command 7 bytes.
- Opcode
- Start Attribute Number: The seed for the ATTRIBUTE\_GETALL command. The first attribute retrieved is equal to or greater than this field.

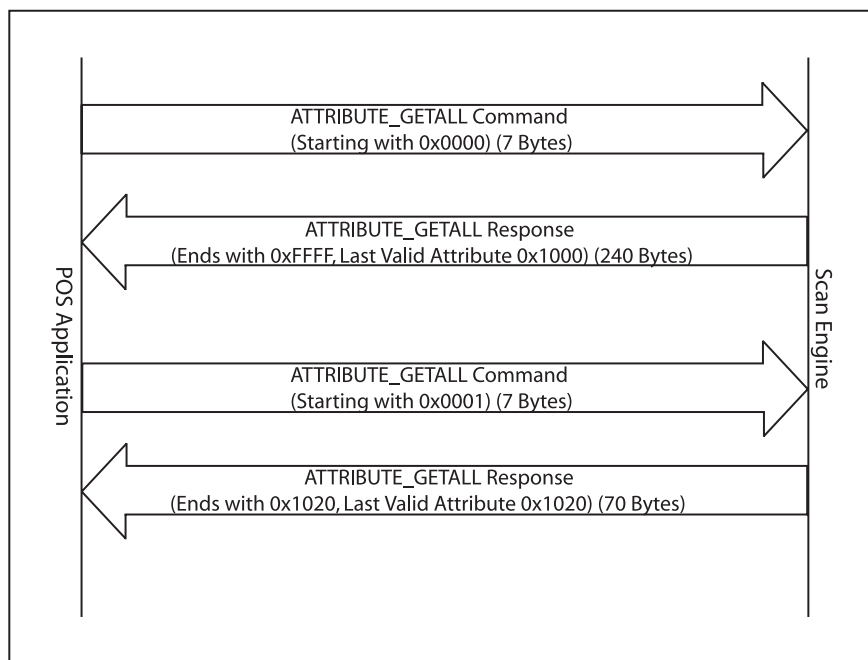
## Response Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GETALL) (0x01)							
3	Status							
4	Attribute Number 1 (MSB)							
5	Attribute Number 1 (MSB)							
...								
...								
237	Attribute Number N							



**Key**

- Length: Length of the response frame.
- Opcode
- Status:
  - 0 = Command Successful
- Attribute Number: A 16 bit value that represents the attribute number. An attribute value of 0xFFFF indicates that the end of the attribute table was reached. If the attribute value at the end of the list is not 0xFFFF, it is the responsibility of the host software to send an ATTRIBUTE\_GETALL command that starts off with the last attribute number that reported + 1.

**Transaction Diagram****Figure 9-2. Transaction Diagram**

**ATTRIBUTE\_GET**

The ATTRIBUTE\_GET command is used to retrieve a scan engine attribute when the attribute number is provided. The command supports the ability to request one or more parameters that are only limited by the size of the command that it can send. The current size limit of this command is 240 bytes.

**Command Structure**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GET) (0x02)							
3	Reserved (0)							
4	First Attribute (MSB)							
5	First Attribute (LSB)							
...	...							
237	Last Attribute (LSB)							

**Response Structure**

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GETALL) (0x02)							
3	Status							
4...4 + length of attribute	First Attribute Value							
...	...							
...								
237 - length of attribute ...237	Last Attribute Value							



This packet terminates when the last value is 0xFFFF.

## Attribute Value Sub-structure

The Attribute values are stored in a structure that self describes the data type as well as the amount of storage that each attribute supports. The tag that differentiates the types is human readable. For all types, an upper case letter indicates that the value is unsigned. The lower case letter indicates that the value is signed. This rule does not apply to the string 'S' and array 'A' data types.

### Generic Structure

Attribute	Type	Properties	Value
-----------	------	------------	-------

Type	Definition
'B'	Byte - unsigned char
'C'	Char - signed byte
'F'	Bit Flags
'W'	WORD - short unsigned integer (16 bits)
'I'	SWORD - short signed integer (16 bits)
'D'	DWORD - long unsigned integer (32 bits)
'L'	SDWORD - long signed integer (32 bits)
'A'	Array
'S'	String

### Properties

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Persistent	Write	Read

### Key

- Write - Attribute value is writable.
- Read - Attribute value is readable.
- Persistent - Attribute value is non-volatile.

### Value

- Variable depends on the 'TYPE' field.

**Byte Value Structure**

0	1	2	3	4
Attribute	Attribute	Type	Properties	Data
MSB	LSB	'B'	<val>	<val>

**Char Value Structure**

0	1	2	3	4
Attribute	Attribute	Type	Properties	Data
MSB	LSB	'C'	<val>	<val>

**Flag Value Structure**

0	1	2	3	4
Attribute	Attribute	Type	Properties	Data
MSB	LSB	'F'	<val>	<val>

**Short Unsigned Integer Value Structure**

0	1	2	3	4	5
Attribute	Attribute	Type	Flags	Properties	Data
MSB	LSB	'W'	<val>	MSB <val>	LSB <val>

**Short Signed Integer Value Structure**

0	1	2	3	4	5
Attribute	Attribute	Type	Flags	Properties	Data
MSB	LSB	'I'	<val>	MSB <val>	LSB <val>

**Long Unsigned Integer Value Structure**

0	1	2	3	4	5	6	7
Attribute	Attribute	Type	Properties	Data	Data	Data	Data
MSB	LSB	'D'	<val>	MSB <val>	<val>	<val>	LSB <val>

**Long Signed Integer Value Structure**

0	1	2	3	4	5	6	7
Attribute	Attribute	Type	Properties	Data	Data	Data	Data
MSB	LSB	'L'	<val>	MSB <val>	<val>	<val>	LSB <val>

**String Value Structure**

0	1	2	3	4	5	6	7	8	9	10	11
Attribute	Attribute	Properties	Flags	Length	Offset	Offset	Value	Value	Value	Value	Value
MSB	LSB	'S'	<val>	MSB	LSB	MSB	LSB	'S'	'T'	'R'	'\0'

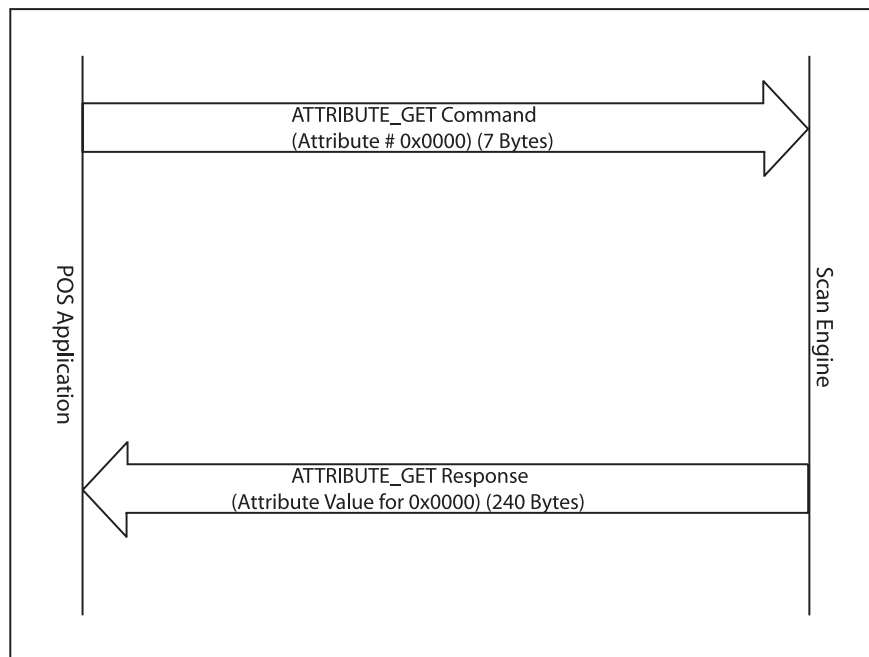


String values returned are NULL terminated. However, the size that is returned always represents the actual size of the data storage. It may seem wasteful however the get functionality serves as a way to self-describe the variable including the storage limits.

**Array Value Structure**

0	1	2	3	4	5	6	7	8	9	10	11	12
Attribute	Attribute	Type	Properties	Sub Type	Length	Length	Offset	Offset	Value	Value	Value	Value
MSB	LSB	'A'	<val>	'W'	MSB (00)	LSB (0x04)	MSB (0x00)	LSB (0x00)	0xDE	0xAD	0xBE	0xEF

The length and offset are represented in byte values. They do not indicate the element count or number of elements.

**Transaction Diagram**

**Figure 9-3. Transaction Diagram**

## ATTRIBUTE\_GETNEXT

The `ATTRIBUTE_GETNEXT` command provides a way for the application to retrieve the next attribute in the attribute table. This command takes an attribute number as a starting point for which the search commences.

This command is very similar to the `ATTRIBUTE_GET` command ([page 9-10](#)) in structure except that the command only supports a single attribute request.

## Command Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GETNEXT) (0x03)							
3	Reserved (0)							
4	Starting Attribute (MSB)							
5	Starting Attribute (LSB)							

## Response Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GETALL) (0x02)							
3	Status							
4...4 + length of attribute	First Attribute Value							
...	...							
...								
237 - length of attribute ...237	Last Attribute Value							

**ATTRIBUTE\_GET\_OFFSET**

The `ATTRIBUTE_GET_OFFSET` command provides the ability to retrieve string/array attribute values that do not fit within a packet. It is the responsibility of the application to determine what the starting offset is.

## Command Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GET_OFFSET) (0x04)							
3	Reserved (0)							
4	First Attribute (MSB)							
5	First Attribute (LSB)							
6	Offset (MSB)							
7	Offset (LSB)							

## Response Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_GET_OFFSET) (0x04)							
3	Status							
4...4 + length of attribute	GET_OFFSET Attribute Value							

ATTRIBUTE\_SET

The ATTRIBUTE\_SET command provides a mechanism for the application to change attributes on the device. The values altered by the ATTRIBUTE\_SET are by definition volatile. These values do not persist when a power cycle occurs. See [ATTRIBUTE\\_STORE on page 9-17](#) for non-volatile storage.

The number of attribute "sets" is limited by the size of the packet that can be sent to the device (currently 240 bytes).

Command Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_SET) (0x05)							
3	Reserved (0)							
4	Attribute Value							
...	Attribute Value N							

Response Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_SET) (0x05)							
3	Status							

Status Key

- 0 = Command Successful
- 1 = Command Unsuccessful.



## ATTRIBUTE\_STORE

The ATTRIBUTE\_STORE command provides a mechanism for the application to change attributes on the device. The values altered by the ATTRIBUTE\_STORE are by definition non-volatile.



*caution The number of non-volatile writes are limited.*



This command works in supervisor mode (i.e., FAT or Manufacturing) only.

The number of attribute "sets" is limited by the size of the packet that can be sent to the device (currently 240 bytes).

### Command Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_STORE) (0x06)							
3	Reserved (0)							
4	Attribute Value							
...	Attribute Value N							

### Response Structure

Byte	Bit							
	7	6	5	4	3	2	1	0
0	Length (MSB)							
1	Length (LSB)							
2	Opcode (ATTRIBUTE_STORE) (0x06)							
3	Status							

### Status Key

- 0 = Command Successful
- 1 = Command Partially Successful.
- 2 = Command Unsuccessful.

## Remote Monitoring Commands

The function of Remote Monitoring commands is to support the Symbol Technologies Inc. Remote Scanner Management architecture. These commands are used by the Host to query the scan engine for important information, i.e., software revision and serial number. In addition, these commands can be used to determine the status of several key scan engine subsystems.

The opcodes/attribute of the Remote Monitoring Commands are in the range of [10002 - 10014 and 20000 - 20006]. The scan engine responds with the requested data byte(s). However, there is an exception; for those commands that require a parameter (i.e., REQUEST\_LASER\_CURRENT), the scan engine responds with CMD\_NACK if an invalid parameter is received.

### *Time to Decode*

#### Attribute Number 10008

##### *Description*

This command is used by the host to request the time to decode of the scan engine in msec. This is read only parameter.

Response Format for ATTRIBUTE_GET command		
Number of Bytes Returned to Host	Byte 12(MSB) & Byte 13(LSB)	Byte 16 & Byte 17
16 Bytes (not including checksum)	Decode time in msec	Checksum on Byte 0 through Byte 15

### *Internal Temperature*

#### Attribute Number 10009

##### *Description*

This command is used by the host to request the temperature of the scan engine. This is read only parameter. The value returned to the Host represents the temperature of the last measurement and a measurement is taken only when the engine is scanning. The value returned to the host immediately after power-on is the birth temperature value.

The 10-bit value returned by the host represents the voltage of the sensor output. The sensor output has a range of -40°C to 105°C, where the sensor slope is -7.75 mV / °C.

The sensor transfer function is linear according to the following: -40°C = 1.9V, 25°C = 1.4V and 105°C = 0.77V

To convert the 10-bit data value from scan engine to a temperature:

$$V = (10\text{-bit data value}/1023) * 2.68 \text{ Volts}$$

$$^{\circ}\text{C} = \{V - 1.59\} / \{-0.00775\}$$

Response Format for ATTRIBUTE_GET command		
Number of Bytes Returned to Host	Byte 12(MSB) & Byte 13(LSB)	Byte 16 & Byte 17
16 Bytes (not including checksum)	Temperature - ADC value (10-bit)	Checksum on Byte 0 through Byte 15

**Example**

Host wants to request the temperature of the scan engine.

Host sends the following packet to the scan engine:

0x0A, 0x80, 0x04, 0x00, 0x00, 0x06, 0x02, 0x00, 0x27, 0x19, 0xFF, 0x2A

Host receives the "temperature" from scan engine:

0x10, 0x80, 0x00, 0x00, 0x00, 0x0C, 0x02, 0x00, 0x27, 0x19, 0x57, 0x01, 0x02, 0x2E, 0xFF, 0xFF, 0xFC, 0x91

This corresponds to a temperature of:

$$558/1023 * 2.68V = 1.46V$$

$$^{\circ}C = \{1.46 - 1.59\} / \{-0.00775\} = 16.7^{\circ} C$$

**Motor Frequency****Attribute Number 10010****Description**

This command is used by the host to request the motor frequency of the scan engine represented in Hertz (Hz). This is read only parameter.

Response Format for ATTRIBUTE_GET command		
Number of Bytes Returned to Host	Byte 12	Byte 15 & Byte 16
15 Bytes (not including checksum)	Motor Frequency in Hz	Checksum on Byte 0 through Byte 14

**Part Number****Attribute Number 20000****Description**

This command is used by the host to request the part number of the scan engine. This is read/write parameter. Write can be performed only in FAT or Manufacturing mode.

Response Format for ATTRIBUTE_GET command		
Number of Bytes Returned to Host	Byte 16 through Byte 33	Byte 37 & Byte 38
37 Bytes (not including checksum)	Part Number String Value	Checksum on Byte 0 through Byte 36

## ***Serial Number***

### **Attribute Number 20001**

#### ***Description***

This command is used by the host to request the serial number of the scan engine. This is read/write parameter. Write can be performed only in FAT or Manufacturing mode.

<b>Response Format for ATTRIBUTE_GET command</b>		
Number of Bytes Returned to Host	Byte 16 through Byte 31	Byte 35 & Byte 36
35 Bytes (not including checksum)	Serial Number String Value	Checksum on Byte 0 through Byte 34

## ***Date of Manufacture***

### **Attribute Number 20002**

#### ***Description***

This command is used by the host to request the date of manufacturing of the scan engine. This is read/write parameter. Write can be performed only in FAT or Manufacturing mode.

<b>Response Format for ATTRIBUTE_GET command</b>		
Number of Bytes Returned to Host	Byte 16 through Byte 22	Byte 26 & Byte 27
26 Bytes (not including checksum)	Date of Manufacture Value	Checksum on Byte 0 through Byte 25

## ***Date of Last Service***

### **Attribute Number 20003**

#### ***Description***

This command is used by the host to request the date of service of the scan engine. This is read/write parameter. Write can be performed only in FAT or Manufacturing mode.

<b>Response Format for ATTRIBUTE_GET command</b>		
Number of Bytes Returned to Host	Byte 16 through Byte 22	Byte 26 & Byte 27
26 Bytes (not including checksum)	Date of Service Value	Checksum on Byte 0 through Byte 25

## ***Firmware Version***

### **Attribute Number 20004**

#### ***Description***

This command is used by the host to request the firmware version loaded on the scan engine. This is read only parameter.

<b>Response Format for ATTRIBUTE_GET command</b>		
Number of Bytes Returned to Host	Byte 16 through Byte 23	Byte 27 & Byte 28
27 Bytes (not including checksum)	Firmware Version	Checksum on Byte 0 through Byte 26

## Engine ID

### Attribute Number 20005

#### Description

This command is used by the host to request the Engine ID (Class I or Class II) of the scan engine. This is read/write parameter. Write can be performed only in FAT or Manufacturing mode only.

Class I Engine - 0x98

Class II Engine - 0x99

Response Format for ATTRIBUTE_GET command		
Number of Bytes Returned to Host	Byte 12	Byte 15 & Byte 16
15 Bytes (not including checksum)	Engine ID Value	Checksum on Byte 0 through Byte 14

## Hardware Version

### Attribute Number 20006

#### Description

This command is used by the host to request the hardware version of the scan engine. This is read/write parameter. Write can be performed only in FAT or Manufacturing mode only.

Response Format for ATTRIBUTE_GET command		
Number of Bytes Returned to Host	Byte 12	Byte 15 & Byte 16
15 Bytes (not including checksum)	Hardware Version Value	Checksum on Byte 0 through Byte 14



String values returned are NULL terminated.



## ***Simple Serial Interface***

### **Introduction**

This chapter describes the system requirements of the Simple Serial Interface, which provides a communications link between Symbol Technologies decoders (e.g., SE955 scan engine, slot scanners, hand-held scanners, two-dimensional scanners, hands free scanners, and RF base stations) and a serial host. SSI allows the host to control the decoder.

### **Communication**

All communication between the decoder and host occur over the hardware interface lines using the SSI protocol. The [Serial Interface Specification](#) (SIF) is described in [Appendix A, Serial Interface Specification](#).

The host and the decoder exchange messages in packets. (A packet is a collection of bytes framed by the proper SSI protocol formatting bytes.) The maximum number of bytes per packet allowed by the SSI protocol for any transaction is 257 (255 bytes + 2 byte checksum).

Decode data may be sent as ASCII data (unpacketed), or as part of a larger message (packeted), depending on the decoder configuration.

SSI performs the following functions for the host device:

- Maintains a bi-directional interface with the decoder
- Allows the host to send commands which can control the decoder
- Passes data from the decoder to a host device in the formatted SSI packet format or straight decode message.

The SSI environment consists of a decoder, a serial cable which attaches to the host device, and in some instances, a power supply.

The SSI interface transmits all decode data including special formatting (e.g., AIM ID). The format of this data can be controlled via parameter settings. The decoder may also send parameter information, product identification information or event codes to the host.

All commands sent between the decoder and host must use the format described in [SSI Message Formats on page 10-4](#). [SSI Transactions on page 10-27](#) describes the required sequence of messages in specific cases.

[Table 10-1](#) lists all the SSI Opcodes supported by the SE955. It identifies the SSI partner allowed to send a message of each type. The host transmits type **H** opcodes, the decoder transmits type **D** opcodes, and either partner can transmit Host/Decoder (**H/D**) types.

**Table 10-1. SSI Commands**

Name	Type	Opcode	Description	Page
<b>AIM_OFF</b>	H	0xC4	Deactivate aim pattern.	<a href="#">10-4</a>
<b>AIM_ON</b>	H	0xC5	Activate aim pattern.	<a href="#">10-5</a>
<b>BEEP</b>	H	0xE6	Sound the beeper.	<a href="#">10-6</a>
<b>CMD_ACK</b>	H/D	0xD0	Positive acknowledgment of received packet.	<a href="#">10-7</a>
<b>CMD_NAK</b>	H/D	0xD1	Negative acknowledgment of received packet.	<a href="#">10-8</a>
<b>DECODE_DATA</b>	D	0xF3	Decode data in SSI packet format.	<a href="#">10-9</a>
<b>EVENT</b>	D	0xF6	Event indicated by associated event code.	<a href="#">10-11</a>
<b>LED_OFF</b>	H	0xE8	De-activate LED output.	<a href="#">10-12</a>
<b>LED_ON</b>	H	0xE7	Activate LED output.	<a href="#">10-13</a>
<b>PARAM_DEFAULTS</b>	H	0xC8	Set parameter default values.	<a href="#">10-14</a>
<b>PARAM_REQUEST</b>	H	0xC7	Request values of certain parameters.	<a href="#">10-15</a>
<b>PARAM_SEND</b>	H/D	0xC6	Send parameter values.	<a href="#">10-17</a>
<b>REPLY_REVISION</b>	D	0xA4	Reply to REQ_REV contains decoder's software/hardware configuration.	<a href="#">10-18</a>
<b>REQUEST_REVISION</b>	H	0xA3	Request the decoder's configuration.	<a href="#">10-20</a>
<b>SCAN_DISABLE</b>	H	0xEA	Prevent the operator from scanning bar codes.	<a href="#">10-20</a>
<b>SCAN_ENABLE</b>	H	0xE9	Permit bar code scanning.	<a href="#">10-21</a>
<b>SLEEP</b>	H	0xEB	Request to place the decoder into low power.	<a href="#">10-22</a>
<b>START_DECODE</b>	H	0xE4	Tell decoder to attempt to decode a bar code.	<a href="#">10-23</a>
<b>STOP_DECODE</b>	H	0xE5	Tell decoder to abort a decode attempt.	<a href="#">10-24</a>
<b>WAKEUP</b>	H	N/A	Wakeup decoder after it's been powered down.	<a href="#">10-25</a>
<b>CUSTOM_DEFAULTS</b>	H	0x12	Custom defaults option to write/restore parameter values.	<a href="#">10-25</a>
<b>Note:</b> D = Decoder, H = Host, H/D = Host/Decoder				



Figure 10-1 shows the general packet format for SSI messages, and Table 10-2 lists the descriptions of fields that occur in all messages. These descriptions are repeated for each Opcode in the SSI message formats section. For messages that use the *Data* field, the specific type of data is shown in that field.

Length	Opcode	Message Source	Status	Data	Checksum
--------	--------	----------------	--------	------	----------

**Figure 10-1. General Packet Format**

**Table 10-2. Field Descriptions**

Field Name	Format	Sub-Field	Meaning
<b>Length</b>	1 Byte	Length	Length of message not including the check sum bytes. Maximum value is 0xFF.
<b>Opcode</b>	1 Byte	See Table 10-1 for details.	Identifies the type of packet data being sent.
<b>Message Source</b>	1 Byte	0 = Decoder 04 = Host	Identifies where the message is coming from.
<b>Status</b>	Bit 0	Retransmit	0 = First time packet is sent 1 = Subsequent transmission attempts
	Bit 1	Reserved	Always set to zero
	Bit 2	Reserved	Always set to zero
	Bit 3	Change Type (applies to parameters)	0 = Temporary change 1 = Permanent change
	Bits 4 - 7		Unused bits must be set to 0.
<b>Data...</b>	Variable number of bytes	See individual sections for details.	
<b>Checksum</b>	2 Bytes	2's complement sum of message contents excluding checksum.	Checksum of message formatted as HIGH BYTE LOW BYTE
<b>Note:</b> The checksum is a 2 byte checksum and must be sent as HIGH BYTE followed by LOW BYTE.			

## SSI Message Formats

The following sections describe each of the SSI messages that can be communicated between the decoder and host. See [SSI Transactions on page 10-27](#) for the protocol required to transmit these messages.

The messages are separated into two categories:

- Engine Control Commands - Commands that configure and control various scan engine features. These commands are considered public commands and are available to all hosts. These opcodes are in the range of [0x04 - 0xF6].
- Remote Monitoring Commands - Commands that are used for remote monitoring of the scan engine attributes. These commands are also considered public commands and are available to all hosts. These opcodes are in the range of [0x60 - 0x7F].

### Engine Control Commands

#### AIM\_OFF

**Description:** Turn off aiming pattern

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xC4	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xC4	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Data</b>			None
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

#### Host Requirements

This command applies only to decoders which support an aim pattern (see [Table 10-10 on page 10-19](#)).

If the TRIGGER\_MODE parameter is set to HOST, the host can use this command instead of a AIM pull.

#### Decoder Requirements

The decoder turns off the aim pattern, and responds with a CMD\_ACK (if ACK/NAK handshaking is enabled).

If the aim pattern is not supported, the decoder responds with NAK\_DENIED (if ACK/NAK handshaking is enabled).

**AIM\_ON****Description:** *Turn on aiming pattern*

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xC5	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xC5	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Data</b>			None
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

**Host Requirements**

This command applies only to decoders which support an aim pattern (see [Table 10-10 on page 10-19](#)).

If the TRIGGER\_MODE parameter is set to HOST, the host can use this command instead of a AIM pull.

**Decoder Requirements**

The decoder turns on the aim pattern, and responds with a CMD\_ACK (if ACK/NAK handshaking is enabled).

If the aim pattern is not supported, the decoder responds with NAK\_DENIED (if ACK/NAK handshaking is enabled).

The Aim Duration parameter controls the amount of time the aiming pattern stays on during a trigger pull. The valid values for this parameter are 0 - 99, which equal 0.1 to 9.9 seconds in 100 msec increments. [Table 10-3](#) lists Aim mode behavior in various situations.

**Table 10-3. Aim Mode**

Command Sequence	Action performed	Aim duration parameters
<b>AIM_ON</b>	Turns on the aiming pattern indefinitely.	aim duration = 0
<b>AIM_OFF</b>	Turns off the aiming pattern.	aim duration = 0
<b>AIM_ON, START_DECODE</b>	Turns on the aiming pattern. When START_DECODE received turns on scan pattern and begin decoding.	aim duration = 0
<b>AIM_ON, AIM_OFF, START_DECODE</b>	Turns on aiming pattern, turns off aiming pattern, turns on scan pattern and begin decoding.	aim duration = 0
<b>START_DECODE</b>	Turns on aiming pattern for aim duration time, turns on scan pattern and begin decoding.	aim duration > 0

## BEEP

**Description:** *Sound the beeper*

Packet Format

Length	Opcode	Message Source	Status	Beep Code	Checksum
0x05	0xE6	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xE6	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: unused	1 Byte	Identifies the transmission status. All unused bits must be set to 0.
<b>Beep Code</b>	See <a href="#">Table 10-4</a> .	1 Byte	Number that identifies a beep sequence.
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

This Opcode instructs the receiver to sound the beep sequence indicated by the Beep Code field.

For [Table 10-4](#), Duration (a relative term) is the length of a sound, Pitch (a relative term) is the pitch of the sound, and Number of Beeps indicates the number of times a beep pitch is repeated at the specified duration.

**Table 10-4. Beep Code Definitions**

Beep Code	Duration	Pitch	Number of Beeps	Beep Code	Duration	Pitch	Number of Beeps
0x00	Short	High	1	0x0D	Long	High	4
0x01	Short	High	2	0x0E	Long	High	5
0x02	Short	High	3	0x0F	Long	Low	1
0x03	Short	High	4	0x10	Long	Low	2
0x04	Short	High	5	0x11	Long	Low	3
0x05	Short	Low	1	0x12	Long	Low	4
0x06	Short	Low	2	0x13	Long	Low	5
0x07	Short	Low	3	0x14	Fast Warble	Hi-Lo-Hi-Lo	4
0x08	Short	Low	4	0x15	Slow Warble	Hi-Lo-Hi-Lo	4
0x09	Short	Low	5	0x16	Mix 1	Hi-Lo	2
0x0A	Long	High	1	0x17	Mix 2	Lo-Hi	2
0x0B	Long	High	2	0x18	Mix 3	Hi-Lo-Hi	3
0x0C	Long	High	3	0x19	Mix 4	Lo-Hi-Lo	3

**Host Requirements**

The host sends this command to cause the decoder to beep. The host may also send these beep codes as part of the PARAM\_SEND directive.

**Decoder Requirements**

When the decoder receives this command, it beeps the sequence provided in the BEEP directive. If ACK/NAK handshaking is enabled, the decoder ACKs if a valid beep code is requested. Otherwise it sends NAK\_DENIED.

**CMD\_ACK****Description: Positive acknowledgment of received packet**

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xD0				

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xD0	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	0 = Decoder 4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: unused	1 Byte	Identifies the transmission status. All unused bits must be set to 0.
<b>Data</b>			None
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

This message is sent to the SSI packet transmitter when the received packet passes the checksum check and no negative acknowledgment conditions apply (see CMD\_NAK). If the data is in response to a command (e.g., PARAM\_REQUEST, REQUEST\_REVISION, etc.), no ACK is sent.



ACK/NAK handshaking can be disabled, but this is not recommended.

It is not necessary to respond to a valid ACK or NAK message.

**Host Requirements**

The decoder must send a CMD\_ACK or response data within the programmable Serial Response Time-out to acknowledge receipt of all messages, unless noted otherwise in the message description section. If the host sends data and does not receive a response within the programmable serial response time-out, it resends the message (with the retransmit status bit set) before declaring a failure. The host should limit the number of retries.

**Decoder Requirements**

The decoder must send a CMD\_ACK or response data within the programmable Serial Response Time-out to acknowledge receipt of all messages, unless noted otherwise in the message description section. If the decoder does not receive an ACK within this time period, it sends the previous message again. The decoder retries twice more (with the retransmit status bit set) before declaring a transmit error.

**CMD\_NAK****Description:** *Negative acknowledgment of received packet*

Packet Format

Length	Opcode	Message Source	Status	Cause	Checksum
0x05	0xD1				

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xD1	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	0 = Decoder 4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Cause</b>	Reason code	1 Byte	Identifies the reason the NAK occurred: 0 = Reserved 1 = (RESEND) Checksum failure 2 = (BAD_CONTEXT) Unexpected or Unknown message 3 = Reserved 4 = Reserved 5 = Reserved 6 = (DENIED) Host Directive Denied 7 = Reserved 8 = Reserved 9 = Reserved
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

This message is sent when the received packet fails the checksum verification or some error occurred while handling the message.



ACK/NAK handshaking can be disabled, but this is not recommended.

It is not necessary to respond to a valid ACK or NAK message.

Table 10-5 describes NAK types supported by the SE955.

**Table 10-5. Decoder-Supported NAK Types**

NAK Type	Meaning	Receiver Action
<b>NAK_RESEND</b>	Checksum incorrect.	Ensure checksum is correct. Limit number of resends. Send packet again with resend bit set.
<b>NAK_DENIED</b>	Host is unable to comply with the requested message (e.g., beep code is out of range).	Do not send data with this message again. Developer should check values with specified values. Developer should ensure the proper character is sent, if using wake-up character.
<b>NAK_BAD_CONTEXT</b>	Host does not recognize the command.	

The decoder only resends a message twice. If the message is not sent successfully either time, the decoder declares a transmit error and issues transmit error beeps (LO-LO-LO-LO).

## DECODE\_DATA

**Description:** Decode data in SSI packet format

Packet Format

Length	Opcode	Message Source	Status	Bar code Type	Decode Data	Checksum
	0xF3	0x00				

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xF3	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	0 = Decoder	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bits 1-7: unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Bar Code Type</b>	See Table 10-6	1 Byte	Identifies the scanned data code type.
<b>Decode Data</b>	<data>	Variable	Data is decoded data including prefix and suffix sent in ASCII format.
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

The decoder uses this opcode when packeted data is selected to send decoded bar code data to the host. The decoded message is contained in the Decode Data field.

Table 10-6 lists all SE955 supported code types. The associated hex value for each code (as required) is entered in the Code Type field.

**Table 10-6. Supported Code Types**

<b>Not Applicable</b>	0x00	<b>EAN 13 with 5 Supps.</b>	0x8B
<b>Code 39</b>	0x01	<b>EAN 13</b>	0x0B
<b>Codabar</b>	0x02	<b>EAN 13 with 2 Supps.</b>	0x4B
<b>Code 128</b>	0x03	<b>EAN 13 with 5 Supps.</b>	0x8B
<b>Discrete 2 of 5</b>	0x04	<b>MSI</b>	0x0E
<b>IATA 2 of 5</b>	0x05	<b>EAN 128</b>	0x0F
<b>Interleaved 2 of 5</b>	0x06	<b>UPC E1</b>	0x10
<b>Code 93</b>	0x07	<b>UPC E1 with 2 Supps.</b>	0x50
<b>UPC A</b>	0x08	<b>UPC E1 with 5 Supps.</b>	0x90
<b>UPC A with 2 Supps.</b>	0x48	<b>Trioptic Code 39</b>	0x15
<b>UPC A with 5 Supps.</b>	0x88	<b>Bookland EAN</b>	0x16
<b>UPC E0</b>	0x09	<b>Coupon Code</b>	0x17
<b>UPC E0 with 2 Supps.</b>	0x49	<b>RSS-Limited</b>	0x23
<b>UPC E0 with 5 Supps.</b>	0x89	<b>RSS-14</b>	0x24
<b>EAN 8</b>	0x0A	<b>RSS-Expanded</b>	0x25

#### **Host Requirements**

If DECODE\_EVENT reporting is enabled, the beep event message is received prior to the DECODE\_DATA message. If ACK/NAK handshaking is enabled, the host responds to each of these messages.

#### **Decoder Requirements**

Decode data is sent in this format if packeted decode data is selected via parameter. The host responds to this message with a CMD\_ACK, if ACK/NAK handshaking is enabled.



## EVENT

**Description:** *Indicate selected events occurred*

Packet Format

Length	Opcode	Message Source	Status	Event Code	Checksum
0x05	0xF6	0x00			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xF6	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	0 = Decoder	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Event Code</b>	Type of Event Code.	1 Byte	See <a href="#">Table 8-2 on page 8-71</a>
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

The decoder sends this message when an enabled event occurs. Use [Table 8-2 on page 8-71](#), and parameters 0xF0 0X00 through 0xF0 0X07 to determine which events should be reported.

### Host Requirements

The host receives this message when a selected event occurs.

### Decoder Requirements

Generate this message when a selected event occurs.

LED\_OFF

**Description:** De-activate LED output

Packet Format

Length	Opcode	Message Source	Status	LED Selection	Checksum
0x05	0xE8	0x04		0x01	

Field Descriptions

Field Name	Format	Size	Description
Length	Length of message (not including checksum).	1 Byte	Length Field
Opcode	0xE8	1 Byte	Identifies this Opcode type.
Message Source	4 = Host	1 Byte	Identifies where the message is coming from.
Status	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
LED Selection	Bit 0 - 7: LED bit numbers to turn off.	1 Byte	Bit 0 = decode LED All other bits should be set to 0.
Checksum	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

The host sends this message to turn off the decode LED.

**Host Requirements**

None.

**Decoder Requirements**

The decoder turns off the decode LED.

**LED\_ON****Description:** *Activate LED output*

Packet Format

Length	Opcode	Message Source	Status	LED Selection	Checksum
0x05	0xE7	0x04		0x01	

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xE7	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>LED Selection</b>	Bit 0 - 7: LED bit numbers to turn on.	1 Byte	Bit 0 = decode LED All other bits should be set to 0.
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

The host sends this message to turn on the decode LED.

**Host Requirements**

None.

**Decoder Requirements**

The decoder turns on the decode LED.

PARAM\_DEFAULTS

*Description: Sets the parameters to their factory default values*

Packet Format

Length	Opcode	Message Source	Status	Checksum
0x04	0xC8	0x04		

Field Descriptions

Field Name	Format	Size	Description
Length	Length of message (not including checksum).	1 Byte	Length Field
Opcode	0xC8	1 Byte	Identifies this Opcode type.
Message Source	4 = Host	1 Byte	Identifies where the message is coming from.
Status	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
Checksum	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

This command returns all parameters to their factory default settings.

**Host Requirements**

The host sends this command to reset the decoders parameter settings to the factory default values.

**Decoder Requirements**

Upon receiving this command, the decoder resets all its parameters to the factory default values. The behavior is the same as scanning a **Set Factory Defaults** bar code.

## PARAM\_REQUEST

**Description:** Request values of selected parameters

Packet Format

Length	Opcode	Message Source	Status	Request Data	Checksum
	0xC7	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xC7	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Request Data</b>	<Param_num><Param_num> <Param_num>...	Variable	
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

The host uses this message to request selected parameters from the decoder.

### Host Requirements

The host requests the decoder's current values for specific parameters by listing the parameter numbers in the Request\_Data field. If the host asks for a parameter value not supported by the decoder, the decoder does not send a value for this unsupported param\_num. If none of the requested values is supported, the decoder transmits an empty PARAM\_SEND message. If the host requests the value of all the parameters, it sends a special param\_num called ALL\_PARAMS (0xFE) in the first position of the Request\_Data field.



The decoder's response to this command is PARAM\_SEND, not ACK. Depending on the time-out set, and the number of parameters requested, this reply may fall outside the programmable Serial Response Time-out. If this occurs, this is not a time-out error. To compensate, increase the time-out.

### Decoder Requirements

When the decoder receives this message, it processes the information by formatting a PARAM\_SEND message containing all requested parameters supported and their values. The programmable Serial Response Time-out can be exceeded when processing this message, depending on the time-out set and the number of parameters requested.

**Hints for requesting parameter values:**

Before forming a PARAM\_REQUEST, confirm that the decoder supports the requested parameters ([Table 10-7](#)). To find out what parameters are supported, send an 0xFE (request all parameters). The response to this is a PARAM\_SEND which contains all the supported parameters and their values.

**Table 10-7. Example of Supported Parameter Numbers**

Supported Parameter Number	Associated Parameter Values
01	00
02	01
9C	07
E6	63

0xFE must be in the first position of the request\_data field if used, or it is treated as an unsupported parameter. Unsupported parameters are not listed in the PARAM\_SEND response. Requesting unsupported parameters has no effect, but can cause delays in responding to requests for valid parameters. See [Table 10-8](#) for example requests and responses.

**Table 10-8. Example Requests and Replies**

PARAM_REQUEST message		Response PARAM_SEND message
#ALL	05 C7 04 00 FE FE 32	0D C6 00 00 FF 01 00 02 01 9C 07 E6 63 FC 3E
#1, 9C	06 C7 04 00 01 9C FE 92	09 C6 00 00 FF 01 00 9C 07 FD 8E
#All, 1, 9C	07 C7 04 00 FE 01 9C FD 93	0D C6 00 00 FF 01 00 02 01 9C 07 E6 63 FC 3E
#1, 9C, ALL	07 C7 04 00 01 9C FE FD 93	09 C6 00 00 FF 01 00 9C 07 FD 8E
#4	05 C7 04 00 04 FF 2C	05 C6 00 00 FF FE 36
#ALL - 3 times	07 C7 04 00 FE FE FE FC 34	0D C6 00 00 FF 01 00 02 01 9C 07 E6 63 FC 3E
#1 -3 times	07 C7 04 00 01 01 01 FF 2B	0B C6 00 00 FF 01 00 01 00 01 00 FE 2D

## PARAM\_SEND

**Description:** Respond to a *PARAM\_REQUEST*, change particular parameter values

Packet Format

Length	Opcode	Message Source	Status	Beep Code	Param data	Checksum
	0xC6					

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xC6	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	0 = Decoder 4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bits 1, 2: Unused Bit 3: Change Type Bits 4-7: Unused	1 Byte	Bit 0: 1 indicates a retransmit  Bit 3: 1 Permanent change 0 Temporary change - lost when power removed. Unused bits must be set to 0.
<b>Beep code</b>	See <a href="#">Table 10-4 on page 10-6</a> .	1 Byte	If no beep is required, set this field to 0xFF.
<b>Param_data</b>	See <a href="#">Table 10-9 on page 10-17</a> .		The parameter numbers and data to be sent to the requester.
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

This message is sent by the decoder in response to the *PARAM\_REQUEST* message, or by the host to change the decoder's parameter values.

Parameter numbers 0xF0 (+256), 0xF1 (+512), 0xF2 (+768) are used to access parameters whose numbers are 256 and higher. For example, to access the first parameter in the 256-511 range, use 0xF0 and 0x00.

**Table 10-9. Param Data Format**

Parameter Number	Data Format
<b>0 through 0xEF</b>	<param_num> <value>
<b>0xF0, 0xF1, 0xF2</b>	<extended parameter code> <param_num offset> <value>

### Host Requirements



Due to the processing time of interpreting and storing parameters contained in the message, the decoder may not be able to send an ACK within the programmable Serial Response time-out. This is not an error; to compensate, increase the time-out.

The host transmits this message to change the decoder's parameters. Be sure the Change Type bit in the Status byte is set as desired. If no beep is required, the beep code must be set to 0xFF, or the decoder beeps as defined in [Table 10-4](#).

### Decoder Requirements

When the decoder receives a PARAM\_SEND, it interprets and stores the parameters, then ACKs the command (if ACK/NAK handshaking is enabled). These parameters are stored permanently only if the Change Type (bit 3 of the Status byte) is set to 1. If bit 3 is set to 0 the changes are temporary, and are lost when the decoder is powered down.

If the PARAM\_SEND sent by the host contains a valid beep code, the decoder issues the requested beep sequence, and changes the requested parameter values.

The decoder issues a PARAM\_SEND in response to a PARAM\_REQUEST from the host. It responds to the PARAM\_REQUEST message by sending all supported parameter values. No value is sent for any unsupported param\_num. If none of the requested values is supported, the PARAM\_SEND message is transmitted with no parameters. When sending this command, the Change Type bit (bit 3 of Status byte) can be ignored.

### REPLY\_REVISION

**Description:** Reply to REQUEST\_REVISION command with software revision string

Packet Format

Length	Opcode	Message Source	Status	Revision	Checksum
	0xA4	0x00			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xA4	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	0 = Decoder	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Revision</b>	ASCII data	variable	Software revision in ASCII (see format below).
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

### Host Requirements

None.

### Decoder Requirements

The decoder sends its revision string to the host in the following format:

S/W\_REVISION <space> BOARD\_TYPE <space> SCANNER\_ID <space> PGM\_CHKSUM

where:

- **S/W\_REVISION** is the release name of the software
- **BOARD\_TYPE** is *N* for non-flash decoder board, *F* for flash
- **SCANNER\_ID** indicates the type of scan engine paired with the decoder
- **PGM\_CHKSUM** is the two-byte checksum of the program code.



Table 10-10 lists the scan engine codes.

**Table 10-10. Scan Engine Codes**

<b>Engine Code</b>	<b>Engine Description</b>	<b>Aiming Pattern</b>	<b>Blinking Trigger</b>	<b>Laser Clipping</b>
<b>0x00</b>	SE 1200 Standard	No	Yes	No
<b>0x01</b>	SE 1200LR (Long Range)	Yes	No	No
<b>0x02</b>	SE 1200WA (Wide Angle)	No	Yes	No
<b>0x03</b>	SE 1200HV (High Visibility)	Yes	No	No
<b>0x04</b>	SE 1200C1 (Class 1)	No	Yes	No
<b>0x05</b>	SE 1200VHD (Very High Density)	No	Yes	No
<b>0x28</b>	SE 923 Standard	Yes	No	Yes
<b>0x29</b>	Reserved	-	-	-
<b>0x2C</b>	Reserved	-	-	-
<b>0x2D</b>	Reserved	-	-	-
<b>0x2A</b>	SE 923C1 IEC Class 1	No	Yes	Yes
<b>0x2B</b>	Reserved	-	-	-
<b>0x2D</b>	Reserved	-	-	-
<b>0x37</b>	SE824	Yes	Yes	No
<b>0x37</b>	SE824 IEC825 Class 1	Yes	Yes	No
<b>0x90</b>	SE950 IEC825 Class 1	No	No	Yes
<b>0x91</b>	SE950 IEC825 Class 2	Yes	No	Yes
<b>0x98</b>	SE955 IEC825 Class 1	No	Yes	Yes
<b>0x99</b>	SE955 IEC825 Class 2	Yes	Yes	Yes

## REQUEST\_REVISION

**Description:** Request the software revision string from the decoder

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xA3	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xA3	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Data</b>			None
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

### Host Requirements

The host sends this message to request revision information from the decoder. The decoder responds with [REPLY\\_REVISION](#).

### Decoder Requirements

The decoder sends its revision string to the host. See [REPLY\\_REVISION](#) for format.

## SCAN\_DISABLE

**Description:** Prevent the decoder from scanning bar codes

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xEA	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xEA	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Data</b>			None
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

**Host Requirements**

All scan attempts are disabled by this command until either a SCAN\_ENABLE is sent, or the decoder is reset.

**Decoder Requirements**

When the decoder receives this command, it ignores all trigger/START\_DECODE requests until a SCAN\_ENABLE command is received.

**SCAN\_ENABLE****Description: Permit the decoder to scan bar codes**

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xE9	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xE9	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Data</b>			None
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

**Host Requirements**

The host sends the SCAN\_ENABLE command to enable scanning in the decoder. Scanning is enabled upon power-up, so this command need only be sent if a prior SCAN\_DISABLE command was sent.

**Decoder Requirements**

The decoder allows scanning and decoding upon receipt of this command.



At initial power-up, the decoder assumes SCAN\_ENABLED.

**SLEEP**

**Description:** *Request to place the decoder into Sleep power state*

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xEB	0x04			

Field Descriptions

Field Name	Format	Size	Description
Length	Length of message (not including checksum).	1 Byte	Length Field
Opcode	0xEB	1 Byte	Identifies this Opcode type.
Message Source	4 = Host	1 Byte	Identifies where the message is coming from.
Status	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
Data			None
Checksum	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

**Host Requirements**

The host sends this command to place the decoder into Sleep power state. If the low power mode parameter is enabled, the scanner goes into Sleep power state automatically, and the SLEEP command is not necessary.



The decoder does not sleep immediately upon acknowledging the command if it is processing data when the SLEEP command is sent.

**Decoder Requirements**

None.

## START\_DECODE

**Description:** Tell decoder to attempt to decode a bar code

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xE4	0x04			

Field Descriptions

Field Name	Format	Size	Description
<b>Length</b>	Length of message (not including checksum).	1 Byte	Length Field
<b>Opcode</b>	0xE4	1 Byte	Identifies this Opcode type.
<b>Message Source</b>	4 = Host	1 Byte	Identifies where the message is coming from.
<b>Status</b>	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
<b>Data</b>			None
<b>Checksum</b>	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

This command tells the decoder to start a scan and decode session. The decode session ends with a successful decode, a scan session time-out, or a STOP\_DECODE command.

### Host Requirements

If the TRIGGER\_MODE parameter is set to HOST, the host can use this command instead of a trigger pull.

### Decoder Requirements

None.

STOP\_DECODE

**Description:** Tell decoder to abort a decode attempt

Packet Format

Length	Opcode	Message Source	Status	Data	Checksum
0x04	0xE5	0x04			

Field Descriptions

Field Name	Format	Size	Description
Length	Length of message (not including checksum).	1 Byte	Length Field
Opcode	0xE5	1 Byte	Identifies this Opcode type.
Message Source	4 = Host	1 Byte	Identifies where the message is coming from.
Status	Bit 0: Retransmit Bit 1-7: Unused	1 Byte	Identifies the transmission status. Unused bits must be set to 0.
Data			None
Checksum	2's complement sum of message contents excluding checksum.	2 Bytes	Checksum of message.

This command tells the decoder to stop a scan and decode attempt.

**Host Requirements**

The TRIGGER\_MODE parameter must be set to HOST.

**Decoder Requirements**

None.

## WAKEUP

**Description:** Wakeup decoder after it's been put into Sleep power state

If the decoder is in Sleep power state, sending the single character, **NULL** (0x00) wakes up the decoder. This character is only needed when hardware handshaking is not used or is bypassed. (See [Power Management on page 1-4](#).)

## Host Requirements

Once the WAKEUP command is sent, the host must wait at least 10 msec, but less than 1 second before sending additional data, since the decoder is required to wait 1 second after waking up before going back to sleep (if low power mode is enabled).

### ***Decoder Requirements***

The decoder must not return to low power mode for at least 1 second after waking up.




The mechanism to wake up a decoder in this manner also works if characters other than WAKEUP are sent to the decoder. There is, however, no guarantee that these commands are interpreted correctly upon power-up. Therefore, it is not recommended that characters other than WAKEUP be used to awaken the decoder.

The WAKEUP character has no effect if sent when the scanner is awake. If the host is unsure of the scanner power state, it can send the wakeup character anytime it wants to communicate with the scanner.


## CUSTOM DEFAULTS

**Description:** Sets the parameters to their default values. This command returns all parameters to their default settings.

## Packet Format

	Bit							
Byte	7	6	5	4	3	2	1	0
0	Length = 6 (not including checksum)							
1	Opcode = 12h							
2	Message Source = 4							
3	MIMIC  Supported	Reserved						Retransmit
4	Action							
5-6	Checksum							

## Field Descriptions

Field Name	Description
<b>Length</b>	Length of message not including the checksum.
<b>Opcode</b>	The opcode for this message.
<b>Message Source</b>	Identifies the sender of the message: Host = 4
 <b>MIMIC Supported</b>	Identifies compliance to the MIMIC System Architecture.
<b>Retransmit</b>	Identifies if a message was resent or not. Values: 0 = First transmission 1 = Subsequent transmission
<b>Action</b>	Identifies the operation to perform on the custom defaults buffer. Values: 0 = Write to Custom Defaults 1 = Restore Custom Defaults
<b>Checksum</b>	16 bit twos complement checksum of message (two byte field size).

**Host Requirements**

The host sends this command to program or restore the products custom default values.

**Decoder Requirements**

Upon receiving this command, the scan engine writes/stores the current parameter settings to the custom defaults buffer. They can be recovered at any time by sending a restore action.

If the restore action is requested, reset all default parameters as follows:

- If custom defaults were set by sending **Write to Custom Defaults**, send **Restore Custom Defaults** to retrieve and restore the scan engine custom default settings.
- If no custom defaults were set, send **Restore Custom Defaults** to restore the factory default values.



## SSI Transactions

### *General data transactions*

#### **ACK/NAK Handshaking**

If ACK/NAK handshaking is enabled, all packeted messages must have a CMD\_ACK or CMD\_NAK response, unless the command description states otherwise. This parameter is enabled by default, and should remain enabled to provide feedback to the host. Raw decode data and WAKEUP do not use ACK/NAK handshaking since they are not packeted data.

Following is an example of a problem that can occur when ACK/NAK handshaking is disabled:

- The host sends a PARAM\_SEND message to the decoder to change the baud rate from 9600 to 19200.
- The decoder cannot interpret the message.
- The decoder does not implement the changes requested by the host.
- The host assumes that the parameter changes have occurred and acts accordingly.
- Communications are lost because the change did not occur on both sides.

If the ACK/NAK handshaking is enabled, the following occurs:

- The host sends a PARAM\_SEND message
- The decoder cannot interpret the message
- The decoder CMD\_NAKs the message
- The host resends the message
- The decoder receives the message successfully, responds with CMD\_ACK, and implements parameter changes.

**Transfer of Decode Data**

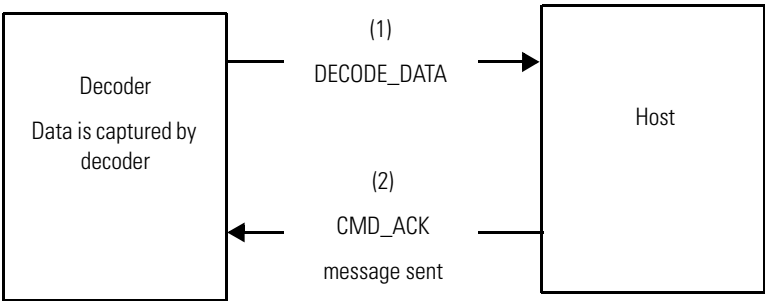
The Decode Data Packet Format parameter controls how decode data is sent to the host. When this parameter is set, the data is sent in a DECODE\_DATA packet. When the parameter is cleared, the data is transmitted as raw ASCII data.



When decode data is transmitted as raw ASCII data, ACK/NAK handshaking does not apply regardless of the state of the ACK/NAK handshaking parameter.

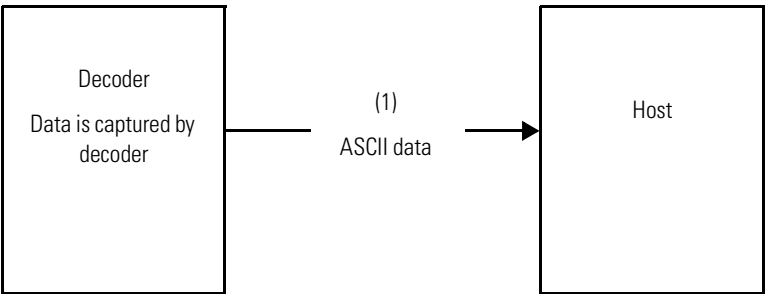
**ACK/NAK Enabled and Packeted Data**

The decoder sends a DECODE\_DATA message after a successful decode. The decoder waits for a programmable time-out for a CMD\_ACK response. If it does not receive the response, the decoder tries to send twice more before issuing a host transmission error. If the decoder receives a CMD\_NAK from the host, it may attempt a retry depending on the cause field of the CMD\_NAK message.



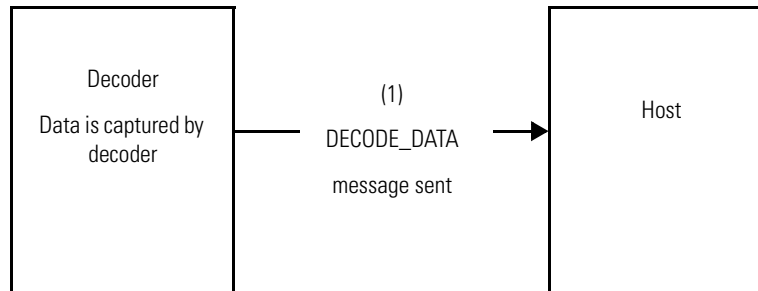
**ACK/NAK Enabled and Unpacketed ASCII Data**

Even though the ACK/NAK handshaking is enabled, no handshaking occurs because the handshaking applies only to packeted data. In this example the packeted\_decode parameter is disabled.

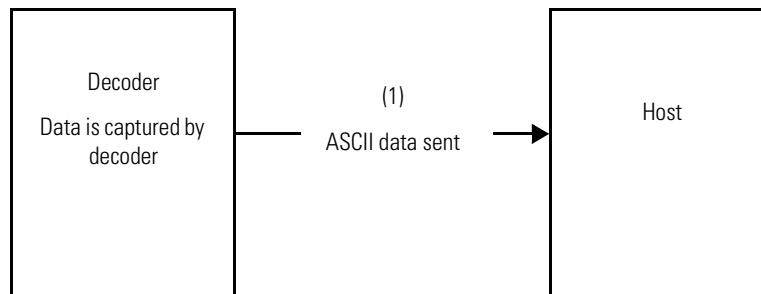


**ACK/NAK Disabled and Packeted `DECODE_DATA`**

In this example ACK/NAK does not occur even though `packeted_decode` is enabled because the ACK/NAK handshaking parameter is disabled.

**ACK/NAK Disabled and Unpacketed ASCII Data**

Data captured by the decoder is sent to the host.



## Communication Summary

### ***RTS/CTS Lines***

All communication must use RTS/CTS handshaking as described in [Appendix A, Serial Interface Specification](#).

### ***ACK/NAK Option***

ACK/NAK handshaking can be enabled or disabled. This handshaking is enabled by default; disabling this is not recommended as it can lead to communication problems, since handshaking is the only indication that a message was received and if it was received correctly. ACK/NAK is not used with unpacked decode data regardless of whether or not this option is enabled.

### ***Number of Data Bits***

All communication with the decoder must use eight bit data.

### ***Serial Response Time-out***

The Serial Response Time-out parameter determines how long to wait for a handshaking response before trying again, or aborting any further attempts. Both the host and decoder should use the same value for this parameter.



A temporary change may be made to the Serial Response Time-out when the host takes longer to process an ACK, or longer data string. Frequent permanent changes are not recommended due to limited write cycles of non volatile memory.

### ***Retries***

When sending data, the host should resend twice after the initial send if the decoder does not respond with an ACK or NAK (if ACK/NAK handshaking is enabled), or response data (e.g., PARAM\_SEND, REPLY\_REVISION). If the decoder replies with a NAK RESEND, the host resends the data. All resent messages must have the resend bit set in the Status byte.

The decoder resends data two times after the initial send if the host fails to reply with an ACK or NAK (if ACK/NAK handshaking is enabled).

### ***Baud Rate, Stop Bits, Parity, Response Time-out, ACK/NAK Handshake***

If the serial parameters above are changed using PARAM\_SEND, the ACK response to the PARAM\_SEND uses the previous values for these parameters. The new values then take effect for the subsequent transaction.

### ***Errors***

The decoder generates a communication error when:

- The CTS line is asserted when the decoder tries to transmit, and is still asserted on each of 2 successive retries
- Failure to receive an ACK or NAK after initial transmit and two resends.

## SSI Communication Notes

If hardware handshaking is not used, messages should be spaced sufficiently apart, and the host must not communicate with the SE955 when the SE955 is sending.

If hardware handshaking is used, frame each message properly with the handshaking signals. Do not try to send two commands within the same handshaking frame.

There is a permanent/temporary bit in the PARAM\_SEND message. Temporary changes are lost when power is removed from the SE955. Permanent changes are written to non-volatile memory. Frequent changes shorten the life of the non-volatile memory.

Do not scan parameter bar codes and send parameters via SSI simultaneously. All parameters can be accessed via SSI, so parameter bar code scanning is not necessary.





# ***Serial Interface Specification***

## **Introduction**

The Serial Interface Specification (SIF) describes the requirements that two digital systems must meet to exchange asynchronous serial data. SIF deals only with the physical flow control and asynchronous serial transmission of data between two digital systems. This specification does not impose any requirements on how the data is packaged and the number of characters exchanged.

SIF data exchange generates errors under certain conditions but does not specify the actions to take to correct the error. This is the responsibility of the software/hardware layer above SIF.

## **Terms and Definitions**

### ***Systems***

Unless otherwise noted, the systems described in this specification are digital systems.

### ***Inactive***

Each system interprets what physical quantity represents inactive. To communicate, two systems must have the same interpretation of inactive.

### ***The Decoder and the Host***

The two systems described in this specification are the decoder and the host. Only one host is allowed to exist at any time.

A Character

This chapter uses the term *character* to generalize a unit piece of information. This unit could be in bit, byte, word, etc.

Data

Data refers to a group of characters.

Tolerances

Unless otherwise noted, all numeric figures stated in this document have a tolerance of  $\pm 5\%$ .

Common Attributes

This section describes requirements common to the decoder and the host.



SIF is a half-duplex communication protocol. To maintain proper communication, the requirements in this section must be met.

All SIF systems have four signal lines. Two are for handshaking and two are for transmitting and receiving serial data.

Many communications packages do not properly use the handshaking lines for half duplex communications. If using a PC communications package such as Windows Terminal, disconnect the hardware handshaking lines from the interface.

The software application libraries included with the optional SE955 Developer Kit provide code to perform proper handshaking.

Table A-1 lists the decoder’s signal lines, and Table A-2 lists the host’s signal lines.

Table A-1. Decoder Signal Lines

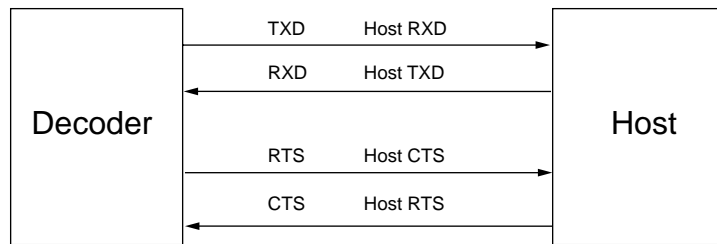
TXD	Serial data transmit output. Drives the serial data receive input of the host.
RXD	Serial data receive input. Driven by the serial data transmit output of the host.
RTS	Request-to-send handshaking line (output). See <a href="#">The Decoder</a> on page A-3.
CTS	Clear-to-send handshaking line (input). See <a href="#">The Decoder</a> on page A-3.

Table A-2. . Host Signal Lines

HOST RXD	Serial data receive input. Driven by the serial data transmit output of the decoder.
HOST TXD	Serial data transmit output. Drives the serial data receive input of the decoder.
HOST CTS	Decoder transmit request (input). See <a href="#">The Host</a> on page A-5.
HOST RTS	Decoder transmit request granted (output). See <a href="#">The Host</a> on page A-5.



Figure A-1 shows the decoder and host signal relationships.



**Figure A-1. Decoder and Host Signals**

## The Decoder

This section describes the requirements that are specific to the decoder.

### Transmitting Data

When the decoder needs to send information, it must first check the CTS line to see if the host is trying to transmit.

Once the bus is available, the decoder can transmit. The decoder is responsible for:

- any programmed intercharacter delays
- retrying if the host communicates during decoder transmission.

### Sample Code for Decoder Transmit Procedure

```

boolean decoder_xmit()
IF (host is requesting to send) THEN
    enable receiving
    give host permission to send
    set up serial response time out
    WHILE (host is still requesting to send) DO
        IF (character was received OR timed out) THEN
            RETURN (FALSE) /* abort transmit */
        END
    END
    disable receiving
    remove host's permission to send
END
WHILE (there are characters to send) DO
    IF (host is not requesting to send) THEN
        send next character
    ELSE
        enable receiving
        give host permission to send
        WHILE (host is still requesting to send) DO
            IF (character was received) THEN
                RETURN (FALSE) /* abort transmit */
            END
        END
        disable receiving
        remove host's permission to send
    END /* resume transmit */
END
RETURN (TRUE)

```

## Receiving Data

The decoder can receive data whenever it grants permission to the host to send its data. If the host is transmitting data, the maximum character-to-character delay allowed is determined by the Host Intercharacter Time-out parameter. The decoder may discard any received data if the host exceeds this time limit.

## Sample Code for Decoder Receive Procedure

```
void decoder_receive()  
  IF (host is requesting to send) THEN  
    give host permission to send  
    WHILE (no characters received) DO  
      IF (host is no longer requesting to send) THEN  
        remove host's permission to send  
        RETURN /* NULL xmit - do not NAK */  
      END  
    END  
    set up host character time out  
    WHILE (not timed out AND not the last character) DO  
      IF (a character was received) THEN  
        reset host character time out  
      END  
    END  
    WHILE (host is requesting to send) DO  
      wait /* for host to end handshake */  
    END  
    remove host's permission to send  
    process received message and prepare response  
  END  
RETURN
```

## The Host

This section describes the requirements specific to the host.

### Transmitting Data

The host only transmits after receiving permission from the decoder. There is no limit to the number of characters per transmit. However, the maximum character-to-character delay cannot exceed the Host Intercharacter Time-out parameter. The HOST RTS signal must return to inactive at the end of transmission (unless the host wants to temporarily prevent the decoder from transmitting).

If the transmit procedure fails, the host must wait for some randomly generated time period before trying again.

### Sample Code for Host Transmit Procedure

```
boolean host_transmit()
  request permission to send
  WHILE (the last character has not been sent) DO
    set up serial response time out
    WHILE (permission has not been granted) DO
      IF (serial response time out expired) THEN
        remove request to send /* transmit failed */
        RETURN (FALSE) /* calling function may retry transmit */
      END
    END
    transmit a character
  END
  remove request to send
  RETURN (TRUE) /* transmit successful */
```

### Receiving Data

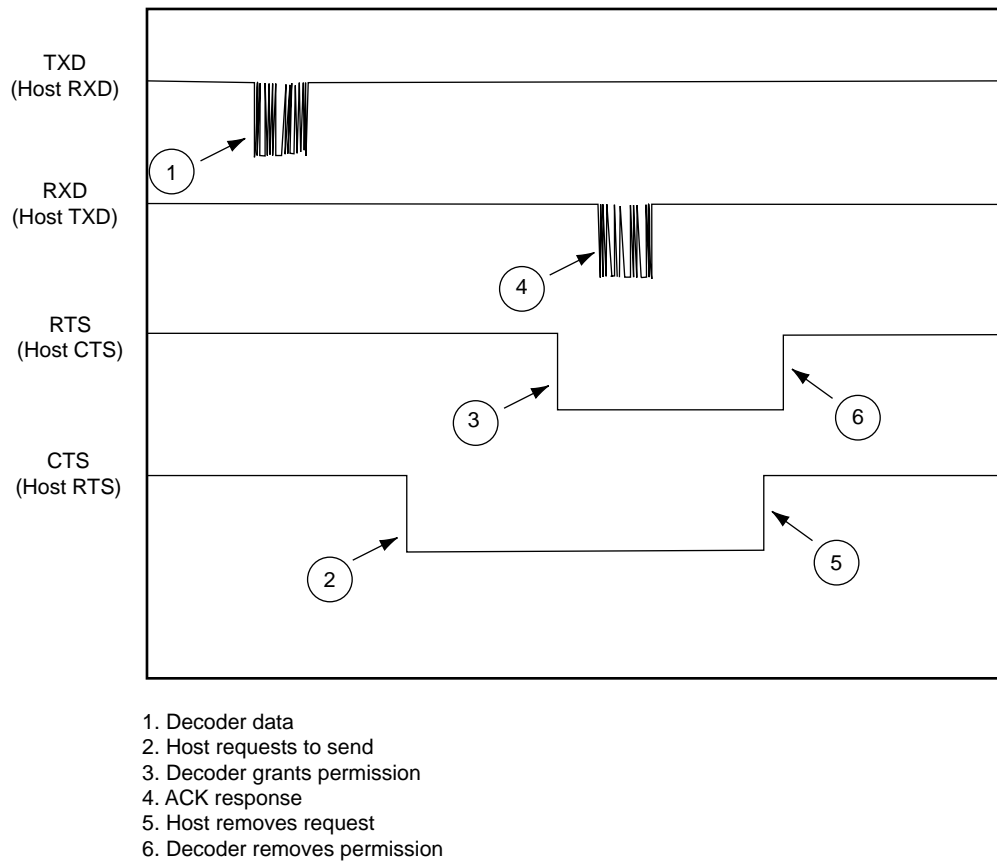
The host must be ready to receive data from the decoder anytime the host is not transmitting. The host can temporarily prevent the decoder from transmitting by using the Host RTS line.

### Sample Code for Host Receive Procedure

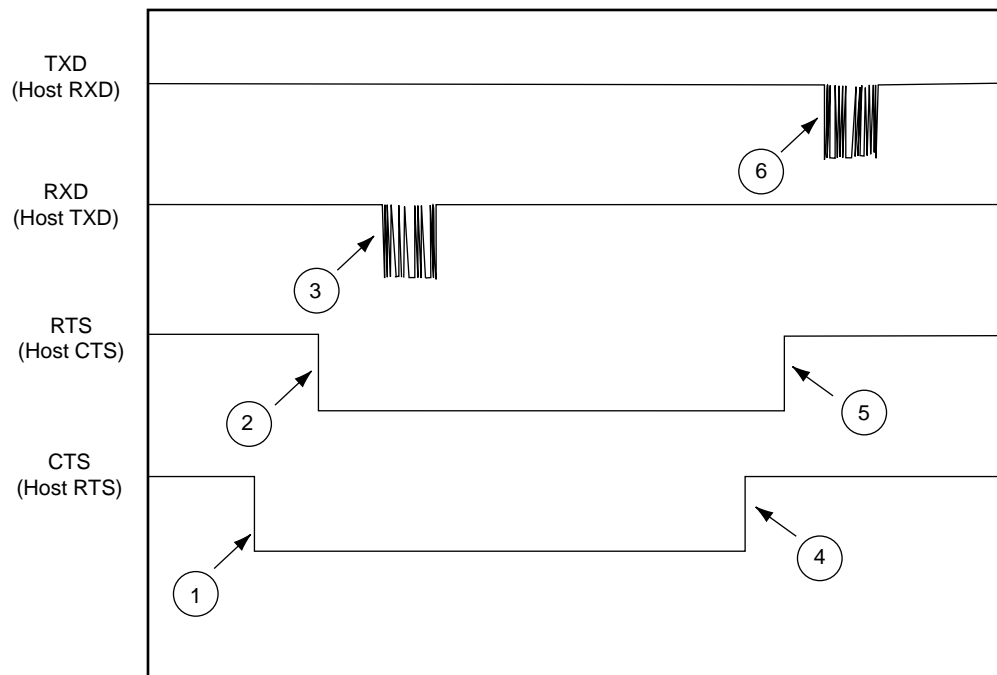
```
void host_receive()
  IF (a character has been received) THEN
    set up intercharacter time out
    WHILE (not timed out AND not the last character) DO
      IF (host can receive right now) THEN
        deassert host RTS /* in case host was holding off decoder */
        IF (a character was received) THEN
          reset intercharacter time out
        END
      ELSE
        IF (host wants to send to decoder) THEN
          RETURN /* so host can transmit */
        ELSE
          request to send /* to hold off the decoder */
          set up new intercharacter time-out
        END
      END
    END
    process received message and prepare response
    RETURN
  END
  RETURN
```

## Transaction Examples

Various transaction examples are shown in [Figure A-2](#) through [Figure A-9](#).

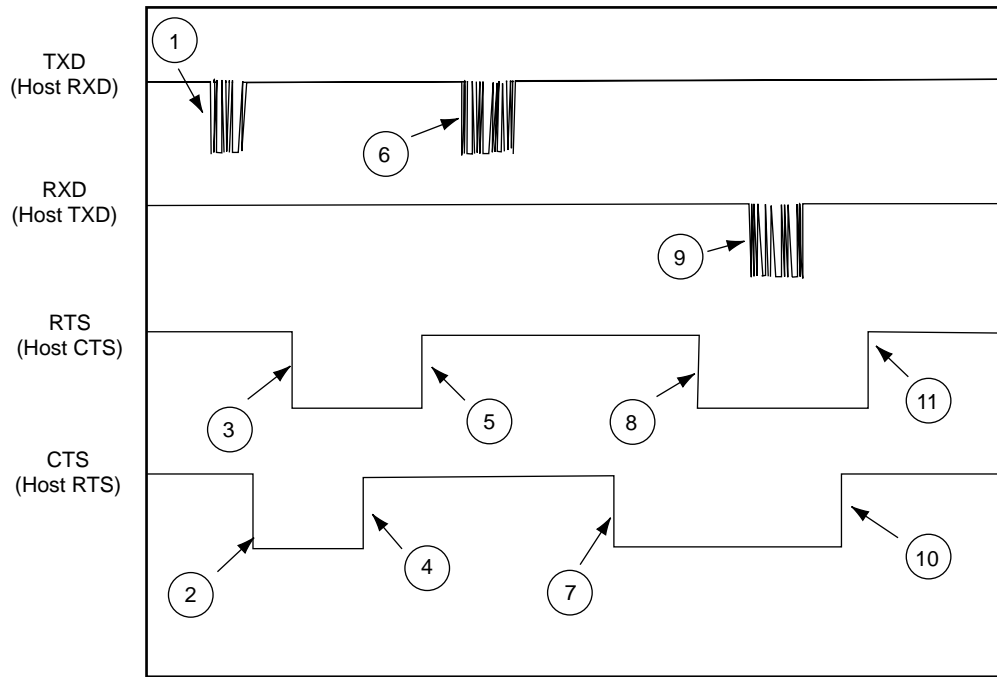


**Figure A-2. Basic Decoder Initiated Transaction**



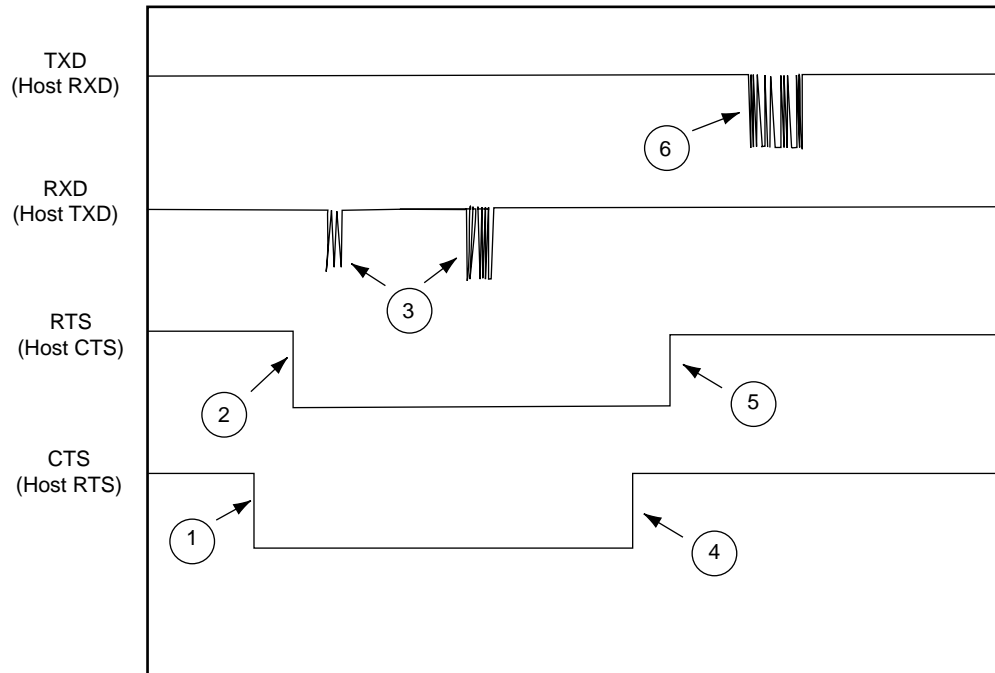
1. Host requests to send
2. Decoder grants permission
3. BEEP command sent
4. Host removes request
5. Decoder removes permission
6. Decoder ACKs

**Figure A-3. Basic Host Initiated Transaction**



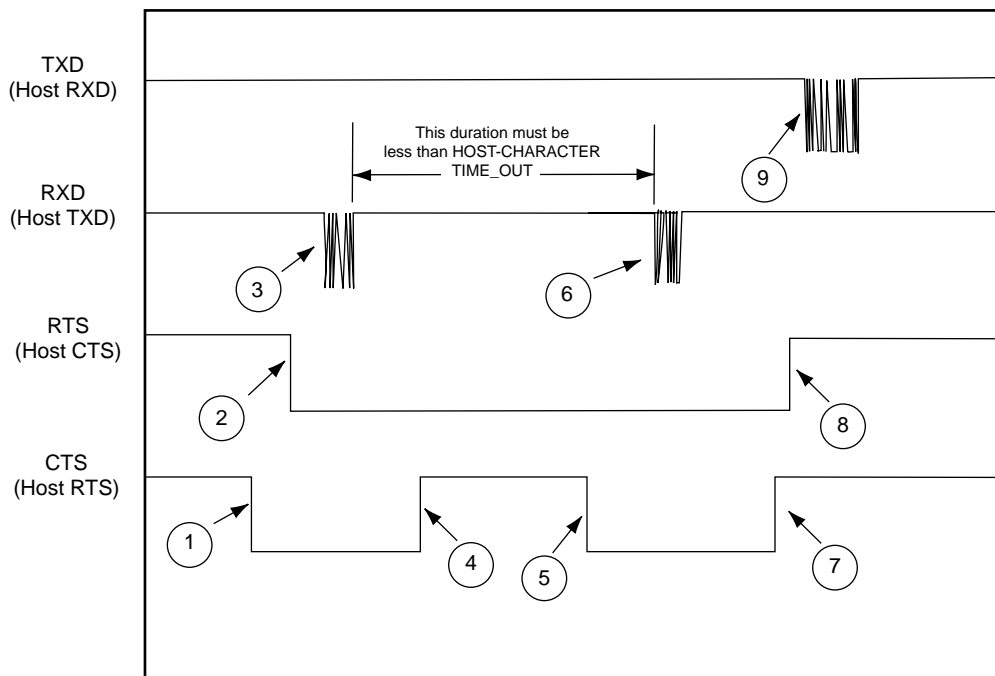
1. Decoder starts to transmit
2. Host asserts RTS causing transmission pause
3. Decoder grants permission for host to send
4. Host removes request without sending
5. Decoder removes permission
6. Decoder resumes transmission
7. Host requests permission to send ACK
8. Decoder grants permission
9. Host sends ACK
10. Host removes request when finished sending
11. Decoder removes permission

**Figure A-4. Host Interrupting Decoder's Transmission**



1. Host requests permission to send
2. Decoder grants permission
3. Host sends 3 nulls, then BEEP command
4. Host removes request when finished sending
5. Decoder removes permission
6. Decoder ACKs

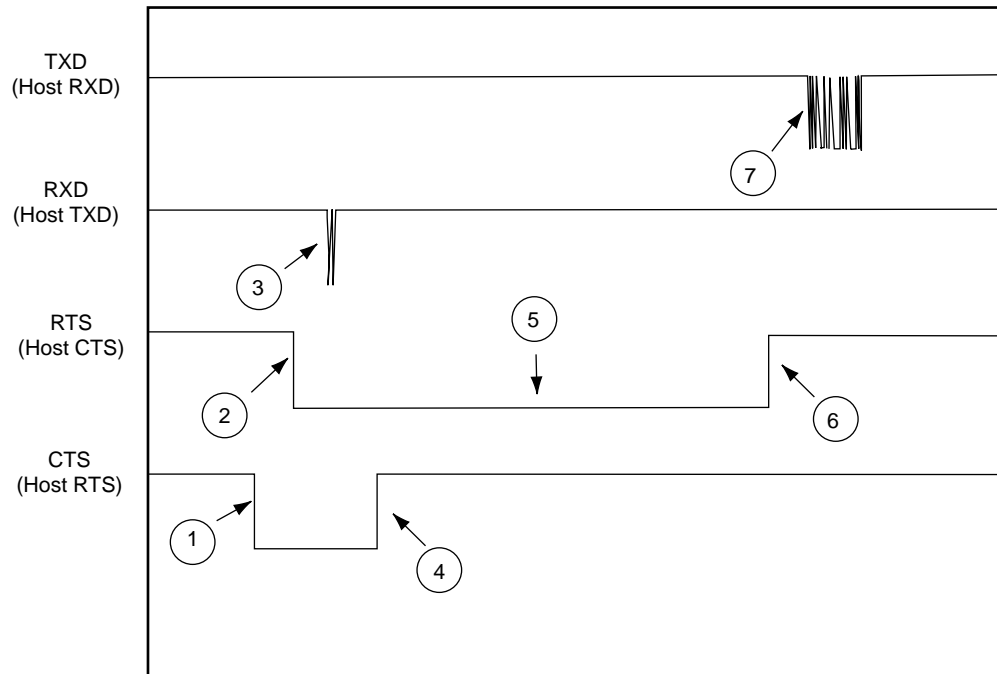
**Figure A-5. Host Initiated Transmission with Leading Nulls  
(Decoder in Continuous Power Mode)**



1. Host requests permission to send
2. Decoder grants permission
3. Host sends 1/2 BEEP command
4. Host removes request (ignored by decoder until transmit complete or timed out)
5. Host requests again (ignored by decoder until transmit complete or timed out)
6. Host sends remainder of BEEP command
7. Host removes request
8. Decoder removes permission
9. Decoder ACKs

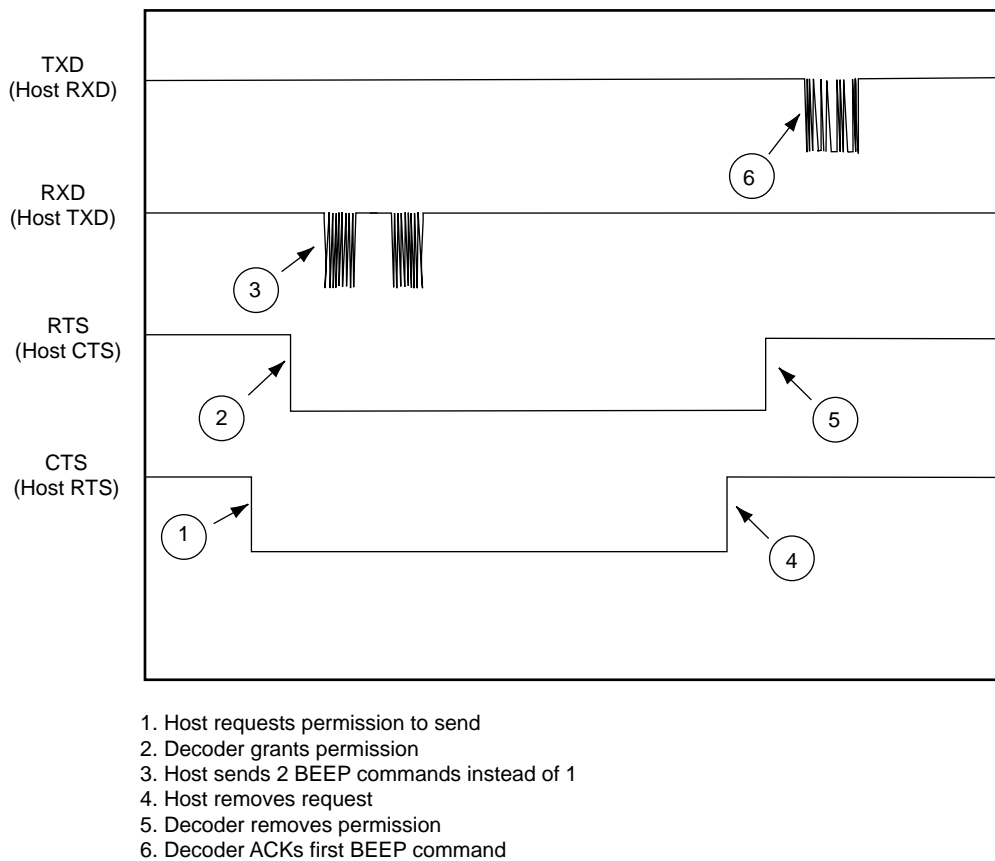
**Figure A-6. Host Initiated Transaction with Host Pausing and Releasing RTS During Transmission**



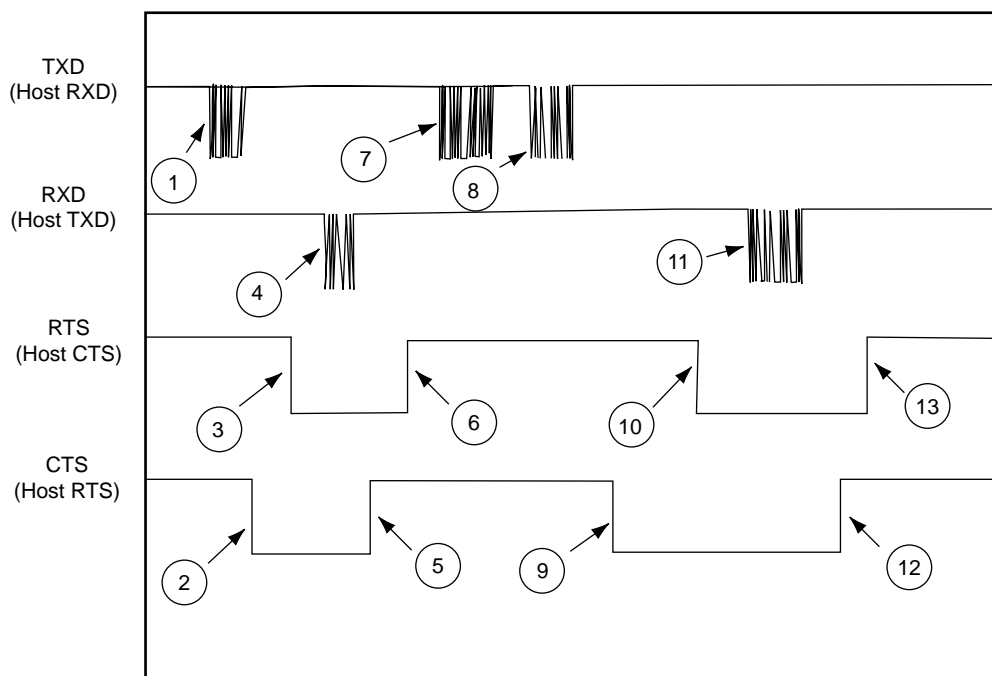


1. Host requests permission to send
2. Decoder grants permission
3. Host sends 2 characters of message
4. Host removes request
5. RTS remains low because decoder is still expecting data
6. Decoder times out waiting for a character and removes permission
7. Decoder sends a NAK resend

**Figure A-7. Error Transmission: Host Sends Only First 2 Characters of 6 Character Message**



**Figure A-8. Error Condition: Host Sends 2 Valid BEEP Commands Back to Back**



1. Decoder starts to transmit
2. Host requests permission
3. Decoder grants permission
4. Host causes abort by sending BEEP
5. Host removes request
6. Decoder removes permission
7. Decoder ACKs
8. Decoder resends data
9. Host requests permission
10. Decoder grants permission
11. Host ACKs
12. Host removes request
13. Decoder removes permission

**Figure A-9. Host Causes Decoder to Abort Transmission**



## ***Miscellaneous Code Information***

### **Introduction**

This Appendix provides information on the following:

- [\*UCC/EAN-128\*](#)
- [\*AIM Code Identifiers\*](#)
- Setting Code Lengths
- Setting Prefixes and Suffixes Via Serial Commands
- Character Equivalents

### **UCC/EAN-128**

UCC/EAN-128 is a convention for printing data fields with standard Code 128 bar code symbols. UCC/EAN-128 symbols are distinguished by a leading FNC 1 character as the first or second character in the symbol. Other FNC 1 characters are used to delineate fields.

When EAN-128 symbols are read, they are transmitted after special formatting strips off the leading FNC 1 character, and replaces other FNC 1 characters with the ASCII 29 (GS) control character.

When AIM symbology identifiers are transmitted, the modifier character indicates the position of the leading FNC 1 character according to AIM guidelines. For example, **lc1** indicates a UCC/EAN-128 symbol with a leading FNC1 character.

Standard Code 128 bar codes which do not have a leading FNC 1 may still be used, but are not encoded according to the EAN-128 convention. Standard Code 128 and UCC/EAN-128 may be mixed in an application. The SE955 autodiscriminates between these symbols, and can enable or disable one or both code types. [Table B-1](#) indicates the behavior of the SE955 in each of the four possible parameter settings.

**Table B-1. Reading Standard Code128 & UCC/EAN 128**

Standard Code 128	UCC/EAN-128	Effect and Example
Disable	Disable	No Code 128 symbols can be read.
Disable	Enable	Read only symbols with leading FNC 1. Examples: FNC1ABCD <sup>FNC1</sup> E are read as ABCD <sup>29</sup> E A <sup>FNC1</sup> BCD <sup>FNC1</sup> E are read as ABCD <sup>29</sup> E FNC1FNC1ABCD <sup>FNC1</sup> E are read as ABCD <sup>29</sup> E ABCD <sup>FNC1</sup> E cannot be read ABCDE cannot be read
Enable	Disable	Read only symbols without leading FNC 1. Examples: FNC1ABCD <sup>FNC1</sup> E cannot be read A <sup>FNC1</sup> BCD <sup>FNC1</sup> E cannot be read FNC1FNC1ABCD <sup>FNC1</sup> E cannot be read ABCD <sup>FNC1</sup> E is read as ABCD <sup>29</sup> E ABCDE is read as ABCDE
Enable	Enable	Read both types of symbols. Examples: FNC1ABCD <sup>FNC1</sup> E are read as ABCD <sup>29</sup> E A <sup>FNC1</sup> BCD <sup>FNC1</sup> E are read as ABCD <sup>29</sup> E FNC1FNC1ABCD <sup>FNC1</sup> E are read as ABCD <sup>29</sup> E ABCD <sup>FNC1</sup> E is read as ABCD <sup>29</sup> E ABCDE is read as ABCDE

## AIM Code Identifiers

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

- J = Flag Character (ASCII 93)
- c = Code Character (see [Table B-2](#))
- m = Modifier Character (see [Table B-3](#)).

**Table B-2. Code Characters**

Code Character	Code Type
A	Code 39
C	Code 128
E	UPC/EAN
F	Codabar
G	Code 93
H	Code 11
I	Interleaved 2 of 5
M	MSI
S	D2 of 5, IATA 2 of 5
X	Code 39 Trioptic, Bookland EAN
e	RSS

The modifier character is the sum of the applicable option values based on the following table.

**Table B-3. Modifier Characters**

Code Type	Option Value	Option
<b>Code 39</b>		
	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, <b>A+I+MI+DW</b> , is transmitted as <b>J<sup>A7</sup></b> Aimld where 7 = (3+4).	

**Table B-3. Modifier Characters (Continued)**

Code Type	Option Value	Option
Trioptic Code 39		
	0	No option specified at this time. Always transmit 0.
	Example: A trioptic bar code 412356 is transmitted as <b>JX0</b> 412356	
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, <b>FNC1</b> Aim Id is transmitted as <b>JC1</b> AimId	
I 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, is transmitted as <b>J10</b> 4123	
Codabar		
	0	No check digit processing.
	1	Reader has checked check digit.
	Example: A Codabar bar code without check digit, 4123, is transmitted as <b>JF0</b> 4123	
Code 93		
	0	No options specified at this time. Always transmit 0.
	Example: A Code 93 bar code 012345678905 is transmitted as <b>JG00</b> 12345678905	
MSI		
	0	Mod 10 check digit checked and transmitted.
	1	Mod 10 check digit checked but not transmitted.
	Example: An MSI bar code 4123, with a single check digit checked, is transmitted as <b>JM0</b> 4123	
D 2 of 5		
	0	No options specified at this time. Always transmit 0.
	Example: A D 2 of 5 bar code 4123, is transmitted as <b>JS0</b> 4123	



**Table B-3. Modifier Characters (Continued)**

Code Type	Option Value	Option
UPC/EAN		
	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 <b>JE000</b> 12345678905	
Bookland EAN		
	0	No options specified at this time. Always transmit 0.
	Example: A Bookland EAN bar code 123456789X is transmitted as <b>JX0</b> 123456789X	

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

**JE0** (UPC chars) (terminator) **JE2** (supplemental) (terminator)

In the SE955, however, the format is changed to:

**JE0** (UPC chars) **JE2** (supplemental)

Therefore, a UPC with two supplemental characters, 01234567890510, is transmitted to the host as a 21-character string, **JE000**12345678905**JE1**10.

## Setting Code Lengths Via Serial Commands

There are two lengths (L1 and L2) for each variable length code type. See the individual code types in 8 for the L1 and L2 parameter numbers.

Depending on the selected option, the scan engine decodes:

- One discrete length bar code
- Two discrete length bar codes
- Bar codes within a range of lengths within the scan engine capability
- Any length of bar codes within the scan engine capability.

Table B-4 lists the requirements for each option.

**Table B-4. Setting Variable Code Lengths**

Code Length Option	L1 value	L2 value
One discrete length is decoded.	Discrete length to decode	0x00
Two discrete lengths is decoded.	Higher length value	Lower length value
Lengths within a range are decoded within the scan engine capability.	Lower length value	Higher length value
Any length bar code is decoded within the scan engine capability.	0x00	0x00

## Setting Prefixes and Suffixes Via Serial Commands

To append a prefix and suffixes to the decode data:

1. Set the Scan Data Transmission Format (parameter 0xE2) to the desired option.
2. Enter the required value(s) for Prefix (0x69), Suffix1 (0x68) or Suffix2 (0x6A) using the hex values for the desired ASCII value from [Table B-5](#).

**Table B-5. Character Equivalents**

Scan Value	Hex Value	Full ASCII Code 39 Encode Char.	Keystroke
1000	00h	%U	CTRL 2
1001	01h	\$A	CTRL A
1002	02h	\$B	CTRL B
1003	03h	\$C	CTRL C
1004	04h	\$D	CTRL D
1005	05h	\$E	CTRL E
1006	06h	\$F	CTRL F
1007	07h	\$G	CTRL G
1008	08h	\$H	CTRL H
1009	09h	\$I	CTRL I
1010	0Ah	\$J	CTRL J
1011	0Bh	\$K	CTRL K
1012	0Ch	\$L	CTRL L
1013	0Dh	\$M	CTRL M
1014	0Eh	\$N	CTRL N
1015	0Fh	\$O	CTRL O
1016	10h	\$P	CTRL P
1017	11h	\$Q	CTRL Q
1018	12h	\$R	CTRL R
1019	13h	\$S	CTRL S
1020	14h	\$T	CTRL T
1021	15h	\$U	CTRL U
1022	16h	\$V	CTRL V

**Table B-5. Character Equivalents (Continued)**

Scan Value	Hex Value	Full ASCII Code 39 Encode Char.	Keystroke
1023	17h	\$W	CTRL W
1024	18h	\$X	CTRL X
1025	19h	\$Y	CTRL Y
1026	1Ah	\$Z	CTRL Z
1027	1Bh	%A	CTRL [
1028	1Ch	%B	CTRL \
1029	1Dh	%C	CTRL ]
1030	1Eh	%D	CTRL 6
1031	1Fh	%E	CTRL -
1032	20h	Space	Space
1033	21h	/A	!
1034	22h	/B	'
1035	23h	/C	#
1036	24h	/D	\$
1037	25h	/E	%
1038	26h	/F	&
1039	27h	/G	'
1040	28h	/H	(
1041	29h	/I	)
1042	2Ah	/J	*
1043	2Bh	/K	+
1044	2Ch	/L	,
1045	2Dh	-	-
1046	2Eh	.	.
1047	2Fh	/	/
1048	30h	0	0
1049	31h	1	1
1050	32h	2	2

**Table B-5. Character Equivalents (Continued)**

<b>Scan Value</b>	<b>Hex Value</b>	<b>Full ASCII Code 39 Encode Char.</b>	<b>Keystroke</b>
1051	33h	3	3
1052	34h	4	4
1053	35h	5	5
1054	36h	6	6
1055	37h	7	7
1056	38h	8	8
1057	39h	9	9
1058	3Ah	/Z	:
1059	3Bh	%F	;
1060	3Ch	%G	<
1061	3Dh	%H	=
1062	3Eh	%I	>
1063	3Fh	%J	?
1064	40h	%V	@
1065	41h	A	A
1066	42h	B	B
1067	43h	C	C
1068	44h	D	D
1069	45h	E	E
1070	46h	F	F
1071	47h	G	G
1072	48h	H	H
1073	49h	I	I
1074	4Ah	J	J
1075	4Bh	K	K
1076	4Ch	L	L
1077	4Dh	M	M
1078	4Eh	N	N

**Table B-5. Character Equivalents (Continued)**

<b>Scan Value</b>	<b>Hex Value</b>	<b>Full ASCII Code 39 Encode Char.</b>	<b>Keystroke</b>
1079	4Fh	O	O
1080	50h	P	P
1081	51h	Q	Q
1082	52h	R	R
1083	53h	S	S
1084	54h	T	T
1085	55h	U	U
1086	56h	V	V
1087	57h	W	W
1088	58h	X	X
1089	59h	Y	Y
1090	5Ah	Z	Z
1091	5Bh	%K	[
1092	5Ch	%L	\
1093	5Dh	%M	]
1094	5Eh	%N	^
1095	5Fh	%O	_
1096	60h	%W	'
1097	61h	+A	a
1098	62h	+B	b
1099	63h	+C	c
1100	64h	+D	d
1101	65h	+E	e
1102	66h	+F	f
1103	67h	+G	g
1104	68h	+H	h
1105	69h	+I	i
1106	6Ah	+J	j

**Table B-5. Character Equivalents (Continued)**

<b>Scan Value</b>	<b>Hex Value</b>	<b>Full ASCII Code 39 Encode Char.</b>	<b>Keystroke</b>
1107	6Bh	+K	k
1108	6Ch	+L	l
1109	6Dh	+M	m
1110	6Eh	+N	n
1111	6Fh	+O	o
1112	70h	+P	p
1113	71h	+Q	q
1114	72h	+R	r
1115	73h	+S	s
1116	74h	+T	t
1117	75h	+U	u
1118	76h	+V	v
1119	77h	+W	w
1120	78h	+X	x
1121	79h	+Y	y
1122	7Ah	+Z	z
1123	7Bh	%P	{
1124	7Ch	%Q	
1125	7Dh	%R	}
1126	7Eh	%S	~
1127	7Fh		Undefined

Values from 1128 through 1255 (hex values 80h through FFh for SSI) may also be set.





## ***Glossary***

<b>Aperture</b>	The opening in an optical system defined by a lens or baffle that establishes the field of view.
<b>Bar</b>	The dark element in a printed bar code symbol.
<b>Bar Code Density</b>	The thickness of the narrowest element in the bar code (i.e., 5 mil, 10 mil, etc.).
<b>Bar Height</b>	The dimension of a bar measured perpendicular to the bar width.
<b>Bar Width</b>	Thickness of a bar.
<b>CDRH</b>	Center for Devices and Radiological Health. A federal agency responsible for regulating laser product safety. This agency specifies various laser operation classes based on power output during operation.
<b>Character</b>	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.
<b>Character Set</b>	Those characters available for encodation in a particular bar code symbology.
<b>Code 39</b>	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.
<b>Dead Zone</b>	An area within a scanner's field of view, in which specular reflection may prevent a successful decode.
<b>Decode</b>	To recognize a bar code symbology (e.g., UPC/EAN) and then analyze the content of the specific bar code scanned.
<b>Depth of Field</b>	The range between minimum and maximum distances at which a scanner can read a symbol with a certain minimum element width.

<b>Digitized Bar Pattern (DBP)</b>	A digital representation of a decoded bar code.
<b>IEC</b>	International Electrotechnical Commission. This international agency regulates laser safety by specifying various laser operation classes based on power output during operation.
<b>LASER - Light Amplification by Stimulated Emission of Radiation</b>	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.
<b>MIL</b>	1 mil = 1 thousandth of an inch.
<b>Nominal</b>	The intended value for a specified parameter. Tolerances are specified as positive and negative deviations from this value.
<b>PMMA</b>	Polymethyl Methacrylic, or Acrylic.
<b>Reflectance</b>	Amount of light returned from an illuminated surface.
<b>Remote Scanner Management</b>	Symbol's Remote Scanner Management technology enables a host to manage a Symbol scanner or scan engine remotely.
<b>Resolution</b>	The narrowest element dimension which can be distinguished by a particular reading device or printed with a particular device or method.
<b>RoHS</b>	Restriction of Hazardous Substances - EU Directive 2002/95/EC. Restricts the use of certain hazardous substances in electrical or electronic equipment sold or used in Europe after July 1, 2006. These substances are lead, mercury, cadmium, hexavalent chromium, polybrominated biphenyls, and polybrominated diphenyl ethers.
<b>SOS</b>	Start of Scan. This signal marks the beginning of each scan and provides a window for the Digitized Bar Pattern.
<b>Space</b>	The lighter element of a bar code formed by the background between bars.
<b>Specular Reflection</b>	The mirror-like reflection of light from a surface, which can "blind" a scanner.
<b>Symbol</b>	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.
<b>Tolerance</b>	Allowable deviation from the nominal.
<b>UPC</b>	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.
<b>Visible Laser Diode (VLD)</b>	A solid state device which produces visible laser light. Laser light emitted from the diode used in this scanner has a wavelength of nominally 650 nanometers.
<b>WEEE</b>	Waste Electrical and Electronic Equipment - EU Directive 2002/96/EC. Electrical or electronic equipment which is waste within the meaning of Article 1(a) of Directive 75/442/EEC, including all components, subassemblies and consumables which are part of the product at the time of discarding. The directive regulates the separate collection, specified treatment, recovery and recycling targets, and funding for handling waste electrical and electronic equipment in Europe effective August 13, 2005.

# Index

## A

AC electrical characteristics	7-1
accessories	
12-pin flex strip	2-14
ambient light immunity	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
anti-reflection coaters	2-7
application notes	7-1

## B

bar codes	8-7
beep after good decode	8-14
beeper tone	8-9
beeper volume	8-8
bi-directional redundancy	8-17
cancel	8-74
codabar	8-52
CLSI editing	8-54
enable/disable	8-52
length	8-53
NOTIS editing	8-54
code 128	8-32
lengths	8-33
UCC/EAN-128	8-32
code 39	8-34
check digit verification	8-37
code 39 full ASCII	8-38
lengths	8-36
transmit check digit	8-37

trioptic code 39	8-34
code 93	
lengths	8-40
discrete 2 of 5	
lengths	8-50
event reporting	8-71–8-72
boot up event	8-72
decode event	8-71
parameter event	8-72
interleaved 2 of 5	8-44
check digit verification	8-47
convert 1 2 of 5 to EAN-13	8-48
lengths	8-45
transmit check digit	8-48
ISBT 128	
enable/disable	8-33
laser on time	8-10
linear code type security	8-16–8-17
MSI plessey	8-55
check digit algorithm	8-58
check digits	8-57
lengths	8-56
transmit check digit	8-57
numeric bar codes	8-73–8-74
power mode	8-12
prefix/suffix values	8-62
RSS	8-59
RSS-14	8-59

RSS-Limited	8-59, 8-60
scan angle	8-11
scan data transmission format	8-63
serial parameters	8-65
baud rate	8-65
data packet format	8-69
host serial response time-out	8-69
intercharacter delay	8-70
parity	8-67
software handshaking	8-68
stop bit select	8-70
set defaults	8-7
symbolologies	
code 11	8-41
code 11 lengths	8-41
UPC-A preamble	8-25
time-out between same symbol	8-14
transmit code ID character	8-61
transmit no read message	8-15
trigger modes	8-13
UPC/EAN	8-18
bookland EAN	8-20
convert UPC-E to UPC-A	8-28
convert UPC-E1 to UPC-A	8-28
decode supplementals	8-21
EAN zero extend	8-29
EAN-13	8-20
EAN-8	8-19
EAN-8 to EAN-13 type	8-29
security level	8-30
supplemental redundancy	8-22
supplementals	8-22
UCC coupon extended code	8-31
UPC-A	8-18
UPC-A check digit	8-23
UPC-A preamble	8-25
UPC-E	8-18
UPC-E check digit	8-23
UPC-E preamble	8-26
UPC-E1	8-19
UPC-E1 check digit	8-24
UPC-E1 Preamble	8-27
beeper	
definitions	1-7
block diagram	1-3
bullets	xv

## C

code 11 bar codes	
code 11	8-41
lengths	8-41
commands	
engine control	10-4
remote monitoring	9-2
commercially available coatings	2-7
anti-reflection	2-7
polysiloxane	2-7
communication	10-1

communications summary	10-30
ACK/NAK	10-30
errors	10-30
number of data bits	10-30
retries	10-30
RTS/CTS lines	10-30
serial response time-out	10-30
configurations	xv
considerations	
abrasion resistance	2-7
anti-reflection coatings	2-7
cell cast acrylic	2-6
cell cast ADC	2-6
chemically tempered float glass	2-6
collection beam geometry	2-5
collection clear aperture	2-6
color	2-7
environment	2-1
ESD	2-1
exit window materials	2-6
grounding	2-1
housing design	2-4
laser clear aperture	2-5
optical	2-4
surface quality	2-7
Continuous On Current	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
continuous power mode	1-4
conventions	
notational	xv

## D

decode distances	
SE950-E100R	4-10
SE950-E105R	5-10
SE950-I100R	4-7
SE955-I105R	5-7
decode zone	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
decode zones	
SE955-E100R 35 degree	4-8
SE955-E100R 47 degree	4-9
SE955-E105R 35 degree	5-8
SE955-E105R 47 degree	5-9
SE955-I100R 35 degree	4-5
SE955-I100R 47 degree	4-6
SE955-I105R 35 degree	5-5
SE955-I105R 47 degree	5-6
default table	8-2
depth of chassis	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
depth of chassis (total)	
SE955-I100R/E100R	4-3
developer kit	2-13, 2-16, A-2
discovery	9-1

**E**

electrical interface	1-6
emissions	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
engine control commands	10-4
error indications	
format	1-7
input	1-7
RS232	1-7
transmission	1-7
ESD protection	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
exit window manufacturers	2-7

**F**

flex strips	
12-pin	2-14

**H**

height	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
humidity	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3

**I**

information, service	xv
input voltage	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
installation	2-1, 3-1, 4-1, 5-1
installing the SE824	2-3

**L**

laser class	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
laser power	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
laser safety	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
location and positioning	2-9
low power mode	1-4

**M**

mounting	2-2
----------	-----

**N**

notational conventions	xv
------------------------	----

**O**

operating temperature	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
operational parameters	8-1
optical resolution	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
overview	1-1

**P**

parameter configuration	9-1
parameters	
set defaults	8-7
parameters, operational	8-1
pitch	
SE955-I105R/E105R	5-2
pitch angle	
SE955-I100R/E100R	4-2
power management	
continuous power	1-4
low power	1-4
power requirements	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
print contrast	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2

**R**

regulatory	
laser class	4-3, 5-3
requirements	6-1
RoHS	4-3, 5-3
remote monitoring commands	9-2
Remote Scanner Management	xiii
remote scanner management	1-3, 3-2, 3-4, 3-7, 9-1
RF immunity	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
RoHS	4-3, 5-3
roll	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
RSM	1-xiii, 1-3, 3-2, 3-4, 3-7, 9-1

**S**

scan angle	
SE950-I000WR/E000WR	4-2
SE955-I105R/E105R	5-2
scan repetition rate	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
scanning	
errors	1-7

scanning current	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
serial interface specification (SIF)	A-1
common attributes	A-2
decoder	A-3
receiving data	A-4
transmitting data	A-3
host	A-5
receiving data	A-5
transmitting data	A-5
terms and definitions	A-1
character	A-2
data	A-2
inactive	A-1
the systems	A-1
tolerances	A-2
service information	xv
shock	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
simple serial interface (SSI)	9-1, 10-1
skew tolerance	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
SLEEP command	1-5
SSI	
commands	10-2
field descriptions	10-3
message formats	10-4
aim off	10-4
aim on	10-5
beep	10-6
cmd ack	10-7
cmd nak	10-8
decode data	10-9
led off	10-12
led on	10-13
param defaults	10-14
param request	10-15
param send	10-17
reply revision	10-18
scan disable	10-20
scan enable	10-21
sleep	10-22
start decode	10-23
stop decode	10-24
wakeup	10-25
SSI transactions	10-27
general data transactions	10-27
ACK/NAK handshaking	10-27
transfer of decode data	10-28
standby current	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
storage temperature	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
supported code types	10-10

surge current	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
symbol global customer interaction center	xv

## T

technical specifications	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
theory of operation	1-1
timing characteristics	7-1
timing waveforms	7-3
transaction examples	A-6

## U

UPC/EAN bar codes	
UPC-A preamble	8-25

## V

Vcc noise level	
SE955-I100R/E100R	4-2
SE955-I105R/E105R	5-2
vibration	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-2

## W

wavefront distortion	2-4
weight	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3
width	
SE955-I100R/E100R	4-3
SE955-I105R/E105R	5-3

# *Tell Us What You Think...*

We'd like to know what you think about this Manual. Please take a moment to fill out this questionnaire and fax this form to: (631) 738-3318, or mail to:

Symbol Technologies, Inc.  
One Symbol Plaza M/S B-10  
Holtsville, NY 11742-1300  
Attention: Advanced Data Capture  
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.

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.







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



**72E-72322-03**  
**Revision A - October 2006**