

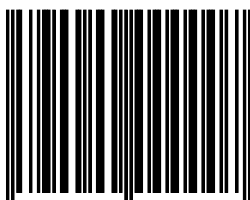
P/N: MUL-53247-02

# ***Advanced Programming Guide***

For Omni-Directional  
Laser Scanners

 **ZEBEX**

Enter/Exit programming



(This barcode is also found at back cover page.)

Framed values are default values.





*This guide is designed for advanced settings of Hand  
Free laser scanner.*

***IMPORTANT NOTICE***

ZEBEX INDUSTRIES INC. MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANT ABILITY AND FITNESS FOR A PARTICULAR PROPOSE. ZEBEX INDUSTRIES INC. SHALL NOT BE LIABLE FOR ERRORS CONTAINED HEREIN OF FOR INCIDENTAL CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURISHING, PERFORMANCE, OR USE OF THESE MATERIALS.

Copyright 2004 by  
**ZEBEX INDUSTRIES INC.**

All other registered trademarks are registered companies' property. ZEBEX is a registered trademark of ZEBEX INDUSTRIES INC.

All rights reserved, reproduction of this document or any portion of its contents is not allowed without the specific written consent of **ZEBEX INDUSTRIES INC.**

Printed In May, 2004  
<http://www.zebex.com.tw/>

## **TABLE OF CONTENTS**

Introduction .....	1
Changing the scanner setting.....	2
Default parameters.....	2
Default values of operating parameters.....	3
Default values of keyboard emulation parameters setting.....	4
Default values of RS-232C serial communication parameters .....	4
Default values of USB emulation parameters....	4
Default values of wand emulation parameters ..	5
Default values of decoding parameters .....	6
Program procedure using barcode menus .....	7
System setting.....	8
Return to factory default.....	9
Return to PC/AT default .....	10
Return to RS-232 default .....	11
Return to USB default .....	11
Return to Wand emulation default.....	12
Return as customer default .....	12
Sleep timeout selection .....	14
Same code delay time.....	22
Beeper sound selection.....	29
Barcode identifier code setting.....	40
Message delay .....	50
Character delay.....	53
Interface communication setting.....	57
RS-232C interface configuration .....	58



Baud rate setting .....	59
Data bit setting .....	64
Stop bit setting .....	65
Parity bit setting .....	66
Handshaking protocol .....	69
ACK/NAK response time setting .....	72
Message terminator for RS-232C .....	78
Keyboard wedge interface configuration ....	82
Message terminator for keyboard wedge .....	83
Keyboard language selection.....	85
Capital Lock setting.....	91
Function key emulation .....	92
USB interface configuration .....	93
Message terminator for USB.....	97
Wand emulation configuration.....	99
Emulation speed selection .....	100
Emulation data output selection .....	102
Emulation data idle state.....	104
Data editing .....	105
Header and trailer .....	106
Truncate header / trailer character .....	108
Add code length .....	110
Symbology configuration .....	111
Reading code selection.....	112
UPC / EAN parameters setting.....	123
Format.....	123
Force UPC-E to UPC-A format.....	127

Force UPC-A to EAN-13 format .....	128
Force EAN-8 to EAN-13 format.....	129
Transmit UPC-A check digit .....	131
Transmit UPC-E leading character.....	132
Transmit UPC-E check digit .....	133
Transmit EAN-8 check digit.....	134
Transmit EAN-13 check digit.....	135
Transmit UPC-A leading character.....	136
Addendum.....	137
Addendum format .....	139
Data Redundant check.....	149
Addendum seek time .....	161
UPC/EAN addendum left/right margin	164
Code 39 parameters setting .....	169
Character set .....	169
Start/stop character transmission.....	170
Check digit .....	171
Code39 reading length setting .....	173
Concatenation.....	175
Code 32 "A" character transmit .....	176
Data redundant check.....	177
Codabar parameters setting.....	180
Format.....	180
Codabar reading length setting .....	183
Codabar concatenation .....	185
Check digit .....	186
Data redundant check.....	188
Code 128 parameters setting .....	190

Check digit .....	190
FNC2 concatenation .....	192
Code 128 reading length setting .....	193
EAN-128 FNC1 character .....	195
Data redundant check .....	196
ITF 2 of 5 parameters setting .....	199
Check digit .....	199
ITF 2 of 5 reading length setting.....	201
Data redundant check .....	204
Chinese postcode parameters setting.....	207
Check digit .....	207
Chinese postcode reading length setting ...	209
Data redundant check .....	212
MSI/PLESSY code parameters setting.....	214
Double check digit.....	214
Single check digit .....	216
MSI/PLESSY code reading length setting..	218
Data redundant check .....	220
Code 93 parameters setting .....	223
Check digit .....	223
Code 93 code reading length setting.....	225
Data redundant check .....	227
Full ASCII code table .....	230
Appendixes .....	295
Appendix A .....	296
Appendix B.....	298

## 1. INTRODUCTION

The series of scanners can be configured by scanning a series of programming barcode labels. This allows decoding options and interface protocols to be tailored to a specific application. The configuration is stored in non-volatile memory and will not be lost by removing power from the scanner

The scanner is a multi-interface communication scanner. If you had ordered only the one type of interface, default will change the initial interface configuration to the interface requested, i.e. RS-232C, keyboard wedge, or USB. But if you had ordered as multi-interface, the initial interface configuration will be set as keyboard wedge (PC/AT); if needed to change to other interface, you need to set up from a programming guide.

***Under the programming mode, the laser scanner will acknowledge a good and valid reading with one short beep. It will give serial beeps for either an invalid or bad reading.***

## **2. CHANGE THE SCANNER SETTING**

In order to change the scanner setting, please follow the sequence below

1. Scan Enter/Exit programming mode barcode (2 beeps Low—high)
2. Scan barcode for the desired feature (1beep)
3. Scan Enter/Exit programming mode barcode to save the configuration.(2 beeps ,long---short)

After reading a valid barcode in programming mode the scanner will gave a high beep.

## **3. DEFAULT PARAMETERS**

This table gives the default settings of all the programmable parameters. The default settings will be restored whenever the "Reset" programming label is scanned and the laser scanner is in programming.

## **DEFAULT VALUES OF OPERATING PARAMETERS**

<b><i>Function</i></b>	<b><i>Default</i></b>
<b><i>Sleep mode</i></b>	
Motor sleep mode	After 30 minutes
Laser sleep mode	After 10 minutes
<b>Scanner timing</b>	
Same code delay	50msec
<b>Beeper Tone</b>	
Frequency	medium
Duration	50msec
<b>Code Identifiers</b>	
Code ID	off
Code 39	M
ITF 2 of 5	I
Chinese post code	H
UPC-A	A
UPC-E	E
EAN-13	F
EAN-8	FF
Codabar	N
Code 128	K
Code 93	L
MSI/Plessy	P

**DEFAULT VALUES OF KEYBOARD EMULATION  
PARAMETERS SETTING**

<b><i>Function</i></b>	<b>Default Values</b>
Keyboard type selection	IBM PC/AT USA
Message terminator	Enter/ carriage return

**DEFAULT VALUES OF RS-232C SERIAL  
COMMUNICATION PARAMETERS**

<b>Function</b>	<b>Default Values</b>
Handshaking protocol	None
ACK/NAK response time setting	300 msec
Baud rate	9600
Data bit	8
Stop bit	1
Parity	None
Message terminator selection	CR/LF

**DEFAULT VALUES OF USB EMULATION  
PARAMETERS**

<b>Function</b>	<b>Default Values</b>
Keyboard Type	US Keyboard
Message Terminator	Enter

## **DEFAULT VALUES OF WAND EMULATION PARAMETERS**

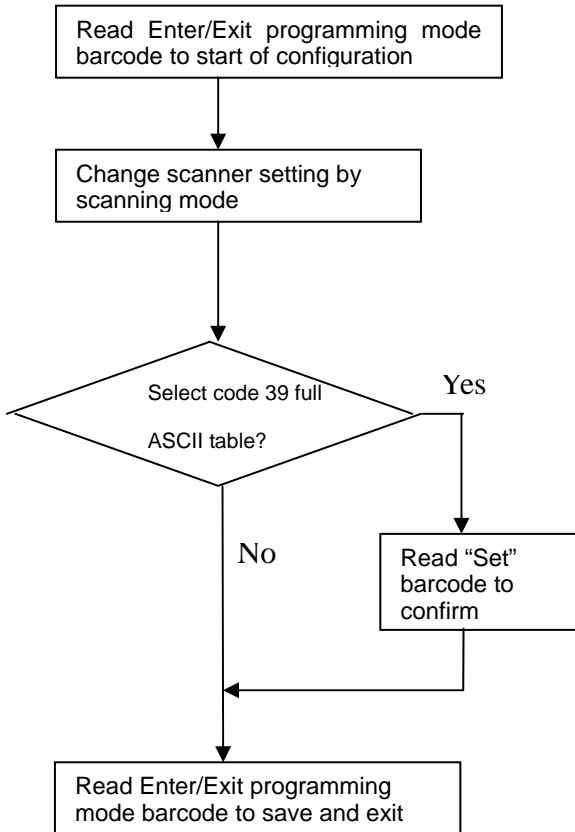
<b>Function</b>	<b>Default Values</b>
Wand emulation speed	Normal
Wand emulation output	Black = High
Data output format	Transmit as scan



## DEFAULT VALUES OF DECODING PARAMETERS

Function	Code	Default Value
Reading codes selection	Code 39	Enable
	ITF 2 of 5	Disable
	Chinese Post Code	Disable
	UPC/EAN/JAN	Enable
	Codabar	Disable
	MSI/PLESSY	Disable
	Code 128	Disable
	Code 93	Disable
	EAN-128	Disable
	Italian Pharmacy	Disable
	ISSN/ ISBN	Disable
Code 39	Codes	Standard
	Start/stop characters	Not transmitting
	Check digit	Disabled
	Concatenation	Off
	Length	3~32
Interleaved 2 of 5	Length	6-32
	Check digit	Disable
Chinese Post Code	Length	10~32
	Check digit	Disable
UPC/EAN /JAN	Format	All
	Addendum	Disable
	UPC-E=UPC-A	Disabled
	UPC-A leading digit	Transmit
	UPC-A check digit	Transmit
	UPC-E leading digit	Transmit
	UPC-E check digit	Transmit
Codabar	Type	Standard
	Start/stop characters	A,B,C,D
	Length	6~32 digits
Code 128	FNC 2 append	Disable
	Check digit	Disable
Code 93	Length	3~32
	Check digit	Not transmit
MSI	Length	6~32
	Check digit	Transmit
Italian Pharmacy	Transmit "A" Character	Not transmitting

# PROGRAM PROCEDURE USING BARCODE MENUS

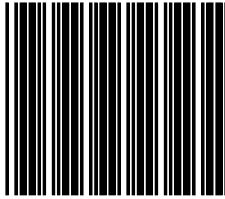


# ***System setting***

*The series scanner is a multi-interface communication scanner. If you had ordered only the one type of interface, ZEBEX will change the initial interface configuration to the interface requested, i.e. RS-232C, keyboard wedge, wand emulation or USB.*

*But if you had ordered as multi-interface, the initial interface configuration will be set as keyboard wedge (PC/AT); if needed to change to other interface, you need to set up from this section.*

Reset (Return to factory default)

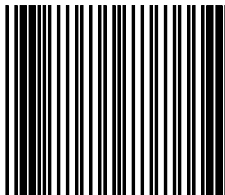


The reading of the “RESET” label turns all the parameters back to default values.

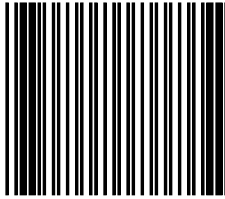
**The scanner remains in the last interface mode when the scanner is reset.**

The reading of the “Show Version” label will show firmware version.

Display Firmware Version

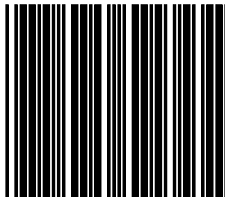


Abort(Exit programming mode)



The reading of the “ABORT” label discards all the parameters reading prior to the “Enter/Exit of Programming”.

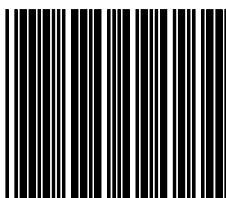
Return to PC/AT default



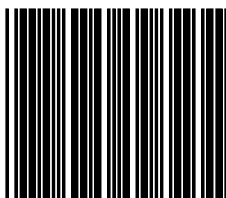
Keyboard wedge interface for IBM PC AT /PS2 and compatibles.

The RS-232C interface scanner is used when connecting to the serial port of a PC or terminal.

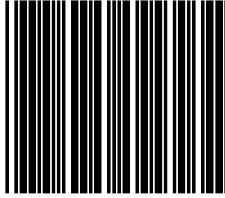
Return to RS-232 default



Return to USB default



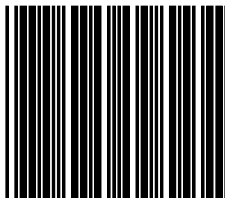
Return to wand emulation default



The Wand emulation is achieved by decoding a scanned barcode and then encoding it again, so that the output is readily decoded by an external decoder designed for processing of wand data.

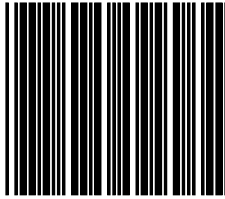
The reading of the label will recover all parameter back to customer default.

Return as customer default



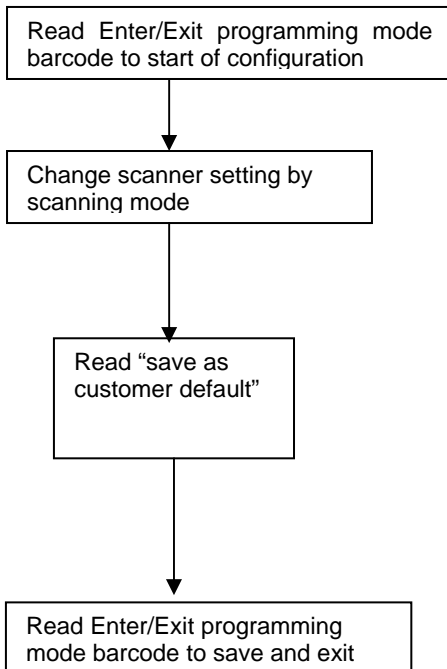


## Save as customer default



The setting allows the multi-desired setting barcode to be saved as a customer default.

### **How to save as customer default**



## *Sleep Timeouts Selection*

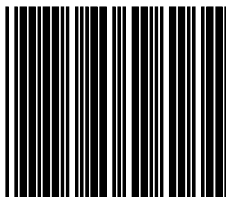
The series scanner with 2 steps (laser and motor) enter to sleep mode.

The timeout programming barcode allow you to set the time for switching the motor or laser to off if the scanner isn't used. This feature is included to reduce power consumption and lengthen scanner life.

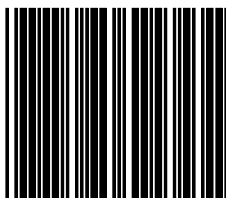
### **NOTE**

**Laser safety requires that the laser switch off before the motor. If you set the motor timeout shorter than the laser timeout, the laser will switch off when the motor times out.**

Motor Sleep mode off

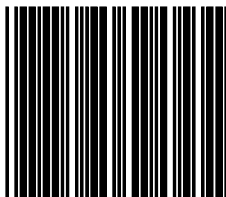


Motor Sleep time 5 min.

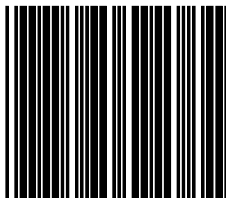




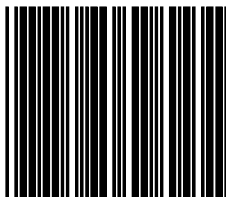
Motor Sleep time 10 min.



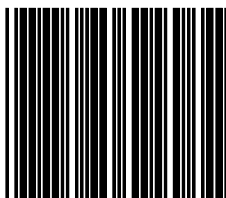
Motor Sleep time 20 min.



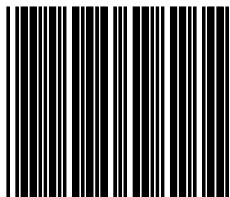
Motor Sleep time 30 min.



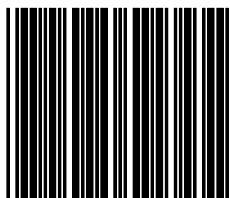
Motor Sleep time 60 min.



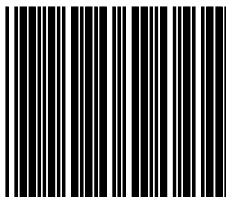
Laser sleep mode off



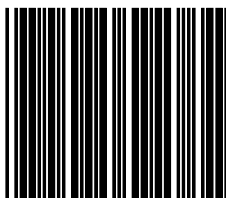
Laser Sleep time 5 min.



Laser Sleep time 10 min.

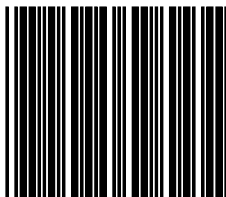


Laser Sleep time 15 min.

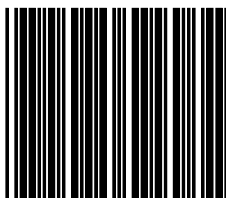




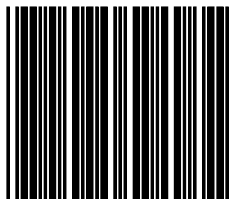
Laser sleep time 20 min.



Laser Sleep time 25 min.



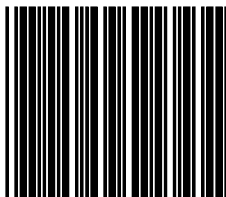
Laser Sleep time 30 min.



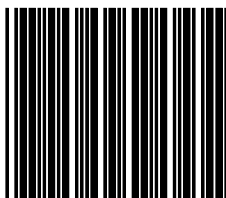
## *Same Code Delay Time*

This parameter sets the minimum time between decodes of the same symbol.  
A typical setting is 200msec

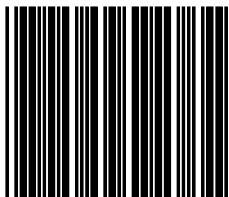
Same code delay time 50 msec.



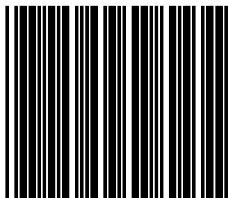
Same code delay time 100 msec.



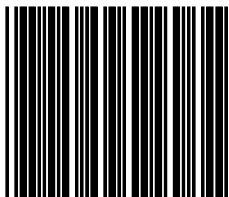
Same code delay time 200 msec.



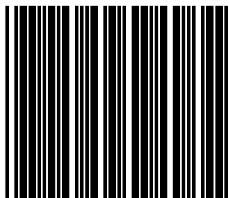
Same code delay time 300 msec.



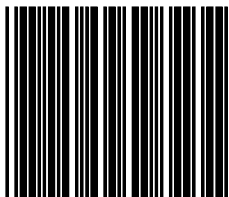
Same code delay time 400 msec.



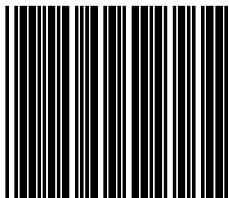
Same code delay time 500 msec.



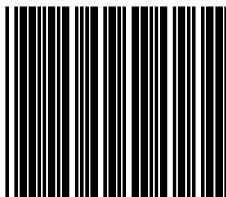
Same code delay time 600 msec.



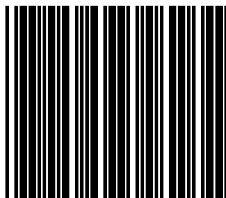
Same code delay time 700 msec.



Same code delay time 800 msec.

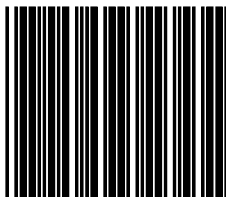


Same code delay time 900 msec.





Same code delay time 1000 msec.



Same code delay time infinite



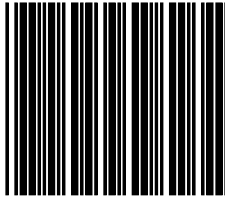


## *Beeper Sound Selection*

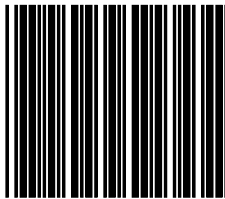
The beeper has below programmable feature

- The “good read” beep sound can be adjusted using  
Tone frequency ,Volume and duration time
  - power on beep
  - enter to sleep mode is silent or sound

Led/Beep after transmission

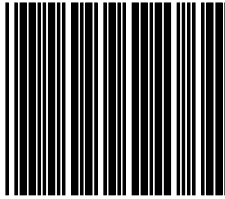


Led/Beep before transmission

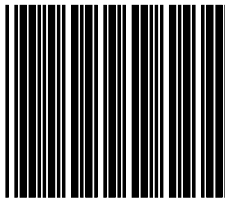




Power-up tone enable

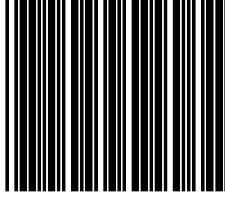


Power-up tone disable

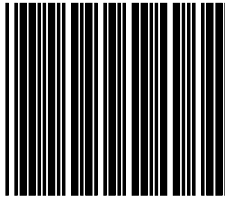




Sleep sound



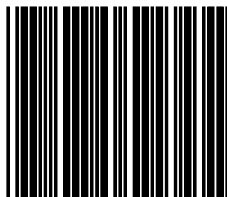
Sleep silent



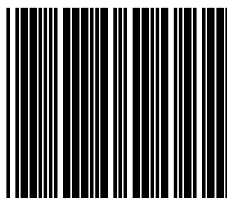




Medium beeper tone

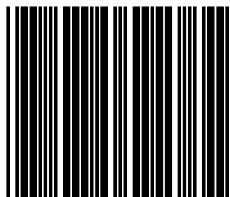


Low beeper tone

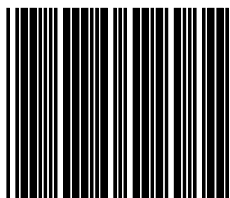




High beeper tone

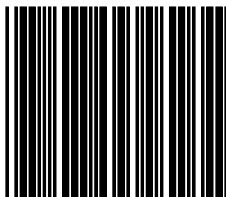


Speaker disable

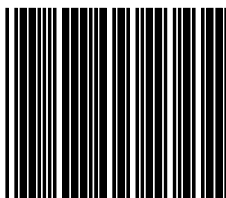




Beeper sound duration (100msec.)

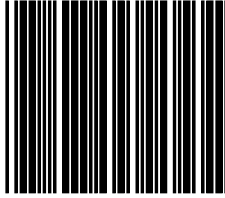


Beeper sound duration (50msec.)

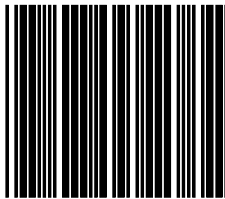




Beeper sound duration(20msec.)



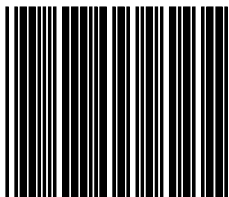
Beeper sound duration(5msec.)



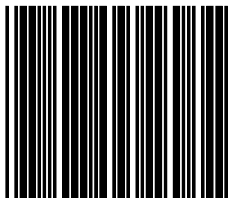




Beeper sound duration 200msec



beeper sound duration 500msec



Loud beeper volume



Medium beeper volume



low beeper volume



## Barcode Identifier

### Code Setting

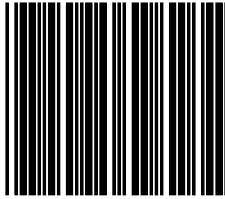
The series scanner can transmit max.2 digit barcode identifier code for different type of barcode, Use enable or disable identifier setting barcode to choose transmit or not transmit barcode identifier code.

Each of the series type scanners also can set max.2digits as barcode identifier code according to different barcode. The procedure is as follows:

- 1.) Scan "Enter/Exit programming mode" label
- 2.) Scan "Barcode identifier setting code" label
- 3.) Scan the new code mark from ASCII table (max. two digits). For example, if one "AB" want for code mark then scan "A" and "B"
- 4.) Scan "save setting to confirm" label
- 5.) Scan " Enter/Exit programming mode" label

## Barcode Identifier Code Selection

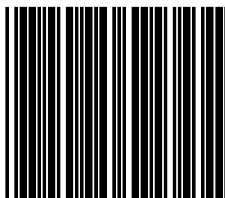
Disable identifier code



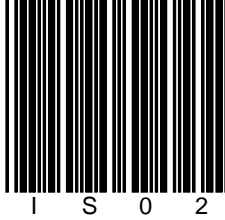
scan “enable identifier code” label to transmit the label ID as following table

Code 39	M
ITF 2 of 5	I
Chinese post code	H
UPC-A	A
UPC-E	E
EAN-13	F
EAN-8	FF
Codabar	N
Code 128	K
Code 93	L
MSI/Plessy	P

Enable identifier code

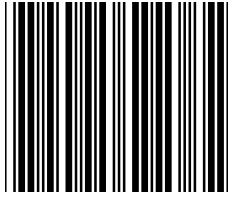


**SET MESSAGE FORMAT WITH CODE IDENTIFIER**  
**AS ALPHA-30 FORMAT**



<b>Code</b>	<b>Code identifier</b>
UPC-A	A
UPC-E	E
EAN-8	FF
EAN-13	F
CODE 39	*
CODBAR	%
ITF 2 OF 5	i
CODE 93	&
CODE 128	#
MSI/PLESSY	@
EAN-128	P

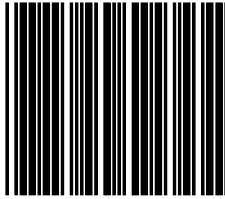
## Enable Identifier code table as AIM standard



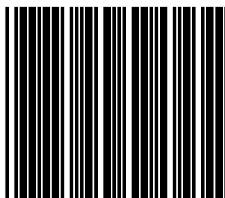


## **Barcode Identifier Code setting**

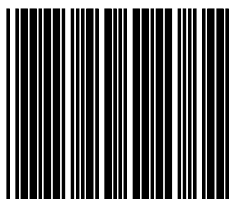
Code 39 identifier code setting



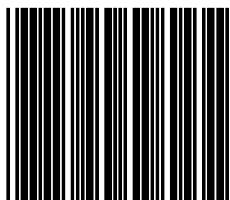
ITF 2 of 5 identifier code setting



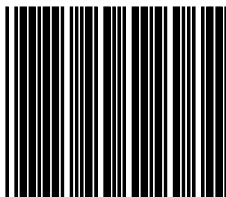
## Chinese Post code identifier code setting



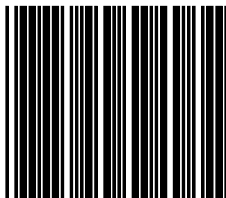
## UPC-E identifier code setting



## UPC-A identifier code setting

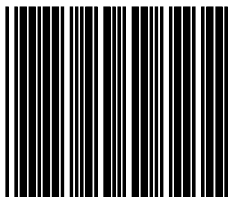


## EAN-13 identifier code setting

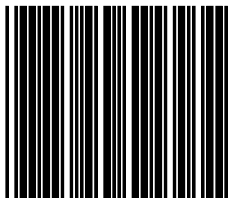




## EAN-8 identifier code setting

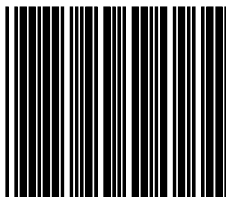


## Codabar identifier code setting

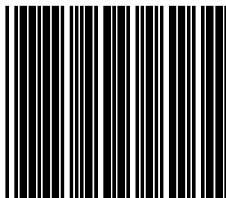




Code 128 identifier code setting



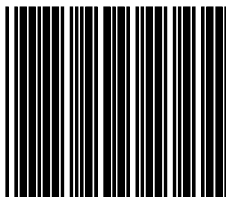
Code 93 identifier code setting



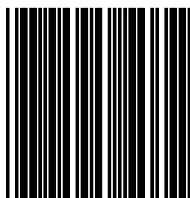




MSI identifier code setting



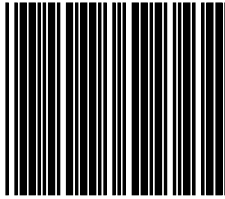
Save setting to confirm



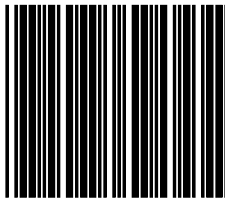
## *Message delay*

The series scanner allow you to add a delay between two consecutive messages. This delay will be added before each date transmission.

Enter message delay 0 ms

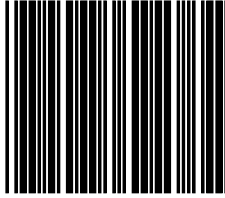


Inter message delay 100 ms

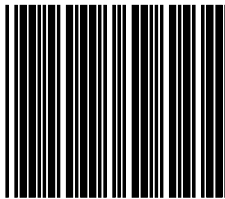




Inter message delay 500 ms



Inter message delay 1000 ms

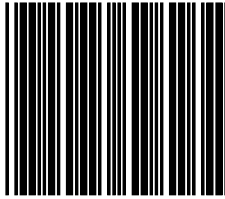




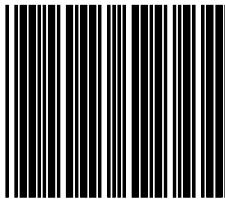
## *Character delay*

This option governs delay time between two consecutive characters; the delay time can be altered by scanning the following labels.

Character delay 5 ms



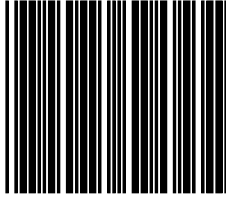
Inter character delay 0 ms



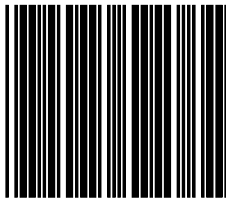




Inter character delay 10 ms

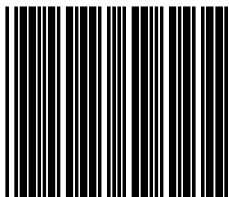


Inter character delay 20 ms





Inter character delay 50 ms



# ***Interface Communication Setting***

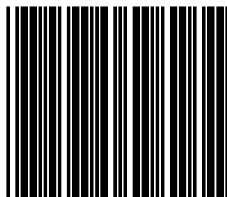
The series scanner must be configured to operate with your POS terminal. The following interfaces are supported:

- \*Keyboard wedge
- \*RS-232 C interface
- \*USB interface
- \*Wand emulation

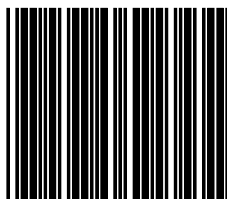
*RS-232C interface*  
*configuration*

## Baud Rate setting

Baud Rate 38400



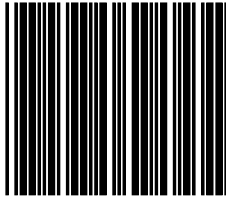
Baud Rate 19200



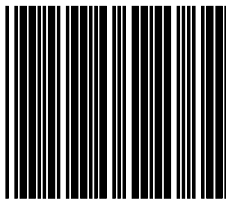




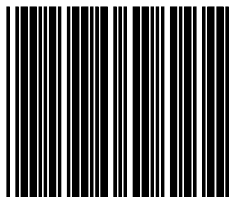
Baud Rate 9600



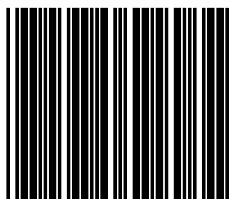
Baud Rate 4800



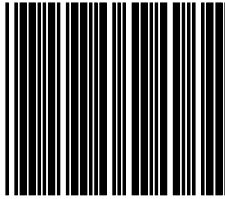
Baud Rate 2400



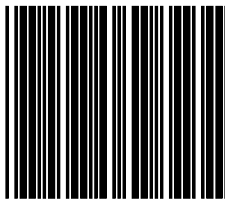
Baud Rate 1200



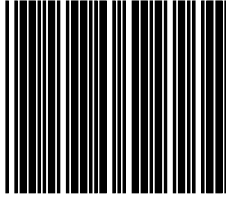
Baud Rate 600



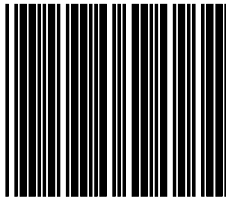
Baud Rate 300



Baud Rate 57600

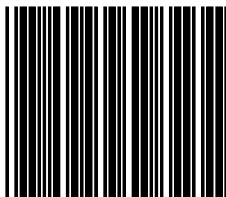


Baud Rate 115200

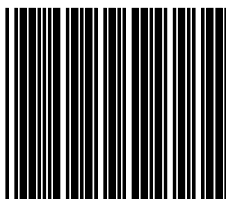


## Date Bit Setting

7 data bit

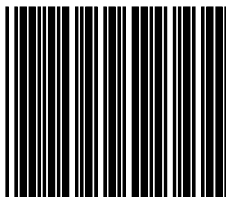


8 data bit

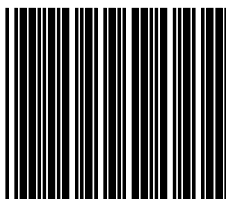


## Stop Bit Setting

1 Stop bit

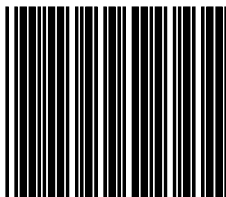


2 Stop bit

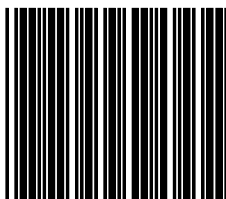


## Parity Bit Setting

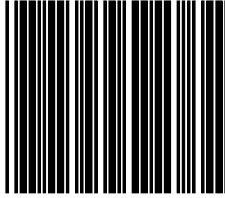
Even Parity



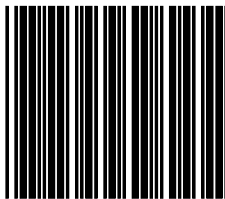
Odd Parity



## Mark Parity

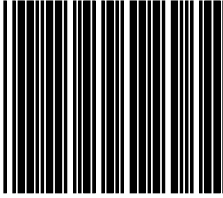


## Space Parity





None Parity



## Handshaking Protocol

The RS-232C type scanner supports four handshaking protocols, With these options of communication protocol.you can tailor the scanner to meet the requirement of most system. These handshaking protocol are:

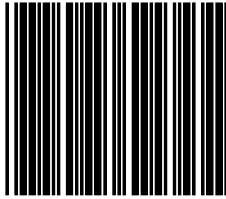
**\*None:** The scanner will transmit any read data unconditionally. The scanner will not check the receiving device nor the transmitted message.

**\*RTS/CTS:** Under this handshaking protocol, the scanner use the RTS pin to instruct the connected device to transmit data and test the CTS pin for readiness of the connected device to receive data.

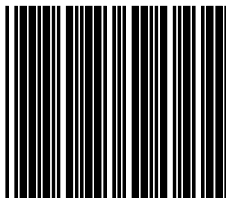
**\*ACK/NAK:** While selecting this option, the scanner waits for an a ACK or NAK signal from the host computer after each data transmission. Normally, the scanner will temporarily store the scanned data in the memory buffer before receiving the ACK or NAK signal. If the ACK signal is received, it will clear the transmitted data and continue to send the next data .In case of the NAK signal is received. it will repeat to transmit the same data until receiving the ACK signal.

**\*Xon/Xoff:** During the data communication, if a scanner receives an Xoff (ASCII 013H),it will stop the transmission at once. The scanner waits for a Xon (ASCII 01H) to start the transmission again.

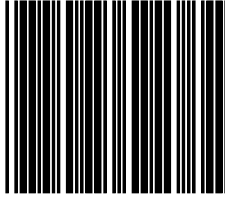
None handshaking



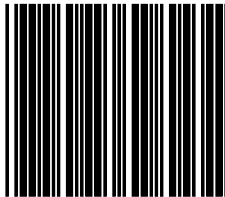
ACK/NAK



Xon/Xoff

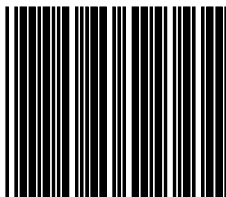


RTS/CTS

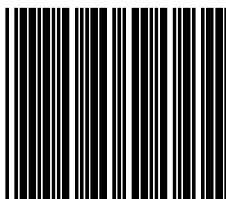


## ACK/NAK Response Time Setting

ACK/NAK response time 300ms

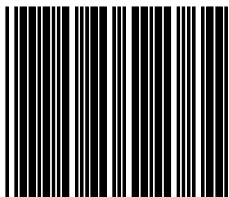


ACK/NAK response time 2s

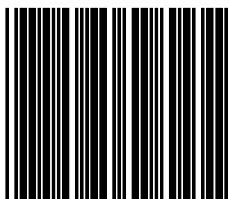




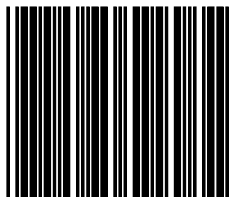
ACK/NAK response time 500ms



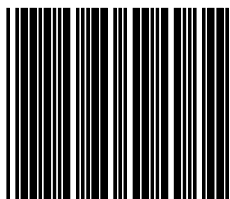
ACK/NAK response time 3s



ACK/NAK response time 1s

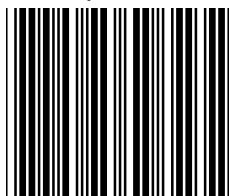


ACK/NAK response time 5s

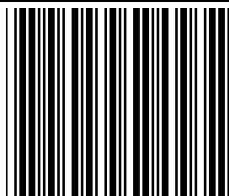




ACK/NAK response time Infinity



Disable ACK/NAK timeout beeper



Enable ACK/NAK timeout beeper



Enable Beeper on<BEL> character



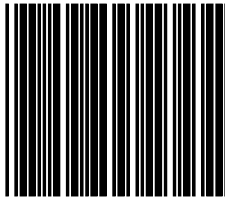
Ignore Beep on <BEL>character



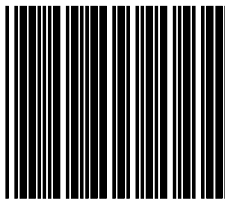
## Message terminator for RS-232C

The series RS-232C type scanner can be programmed to append a terminator to every message sent via the serial port. Different terminator will be appended at the end of message sent from the serial port.

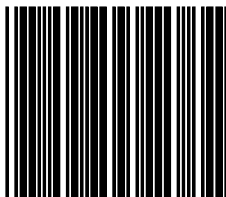
Rs-232 message terminator - none



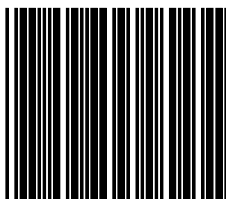
RS232-message terminator - CR/LF



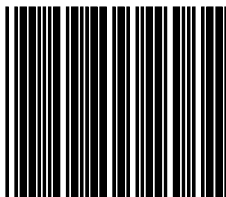
RS-232 message terminator - CR



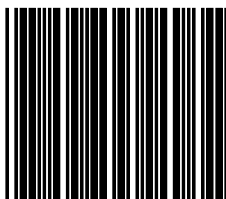
RS-232 message terminator - LF



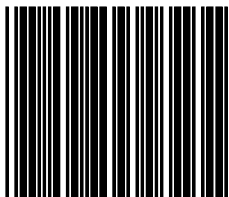
RS-232 message terminator - H tab



RS-232 message terminator - STX/ETX



RS-232 message terminator - EOT

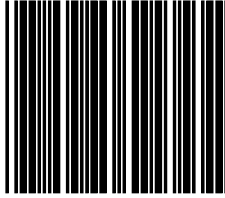


*Keyboard wedge*  
*interface configuration*

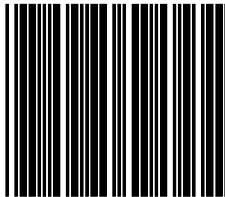


## Message terminator for keyboard Wedge

Keyboard terminator---none

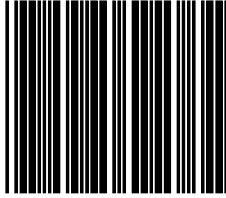


Keyboard terminator---Enter

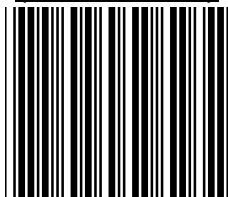




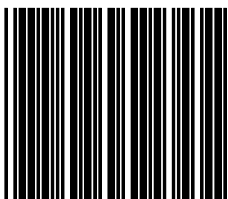
Keyboard terminator---H-TAB



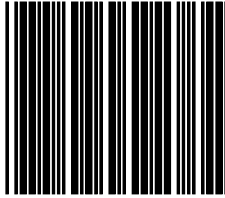
Keyboard Language selection  
Enable International keyboard type  
(Alt method)



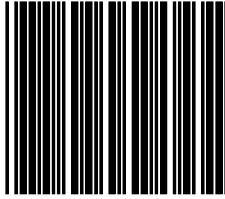
Keyboard language support---USA



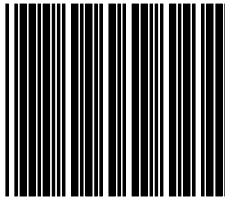
Keyboard language support---Germany



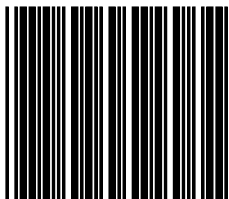
Keyboard language support---UK



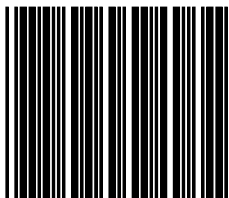
keyboard language support---French



Keyboard language support---Spanish



Keyboard language support--- Italian



Keyboard language support--- Swiss

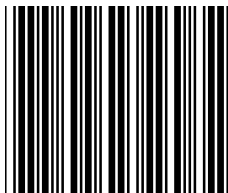


Keyboard language support---Swedish





Keyboard language support---Japanese



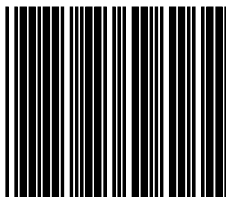
Keyboard language support---Belgium



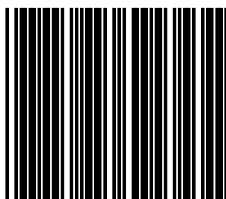
## **Capital Lock**

Select the suitable code to match your keyboard caps lock status

Capital lock on



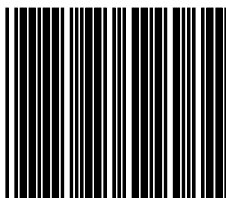
Capital lock off



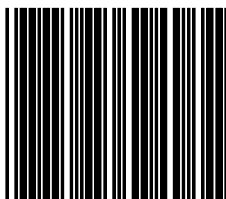
## **Function Key Emulation**

The series allow you to emulate Function keys, Arrow keys, and many other “extended” keys. An IBM compatible keyboard does not translate to ASCII characters. It can be concatenated with input data as header and/or trailer. (see Appendix B)

Function key emulation enable



Function key emulation disable



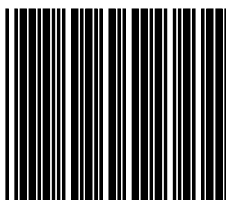
## *USB interface configuration*

The USB mode is effectively a USB HID(Human interface devices) keyboard device emulator that works with hosts, such as USB-compatible operating system and USB ports. USB compatible operating systems are Windows 98, Windows NT 5.0 and later, no additional software is needed since the USB driver support its built-in operating system

## **Keyboard Type**

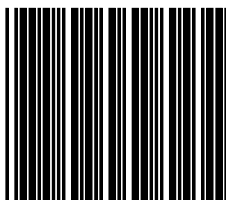
Keyboard type---USA

(Scan method)

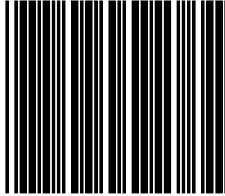


Scan the bar code corresponding to your keyboard type.

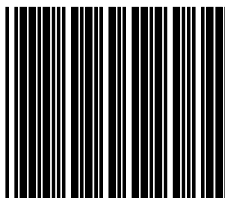
keyboard language support---French



Keyboard language support---Germany



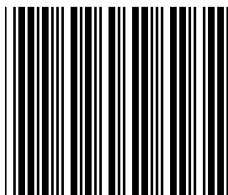
Keyboard language support---Spanish



Keyboard language support---Japanese

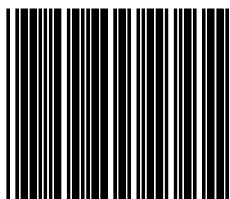


Keyboard type.  
(ALT code method)



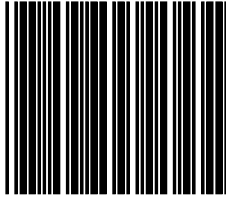
## Message terminator for USB

None

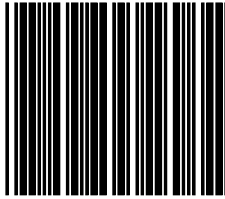


Enter





H tab

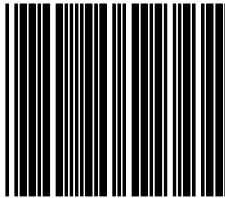


## *Wand emulation configuration*

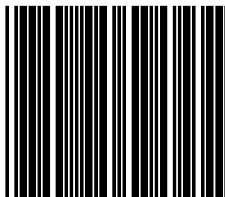
## Emulation speed selection

The data output speed can be set to benefit the external decoder

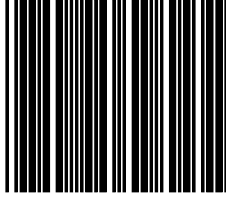
Wand emulation speed=Low  
(1ms narrow element width)



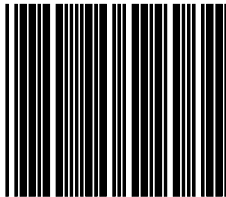
Wand emulation speed =medium  
(600 us narrow element width)



Wand emulation speed= high  
(300 us narrow element width)



Wand emulation speed ---- higher  
(100us narrow element width)

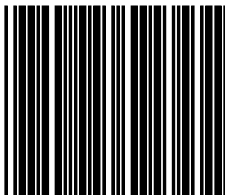




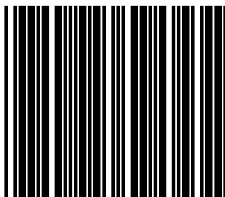
## Emulation data output selection

The decoded data output logic level can be set to befit the external decoder

Wand emulation data output black = high



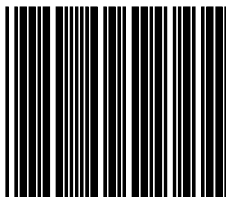
Wand emulation data output black = low



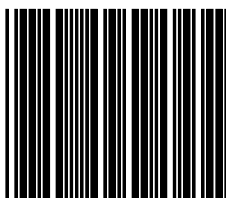




## Wand Data Transmitted as Scanning



Enable Wand output data format as Code 39

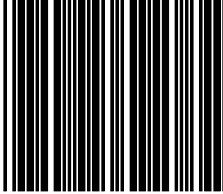




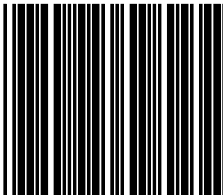
## Emulation data idle state selection

The level refer to wand emulation data signal when not in use

Idle=high



Idle=low



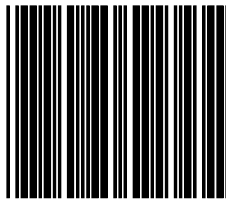
# ***Data Editing***

## HEADER AND TRAILER

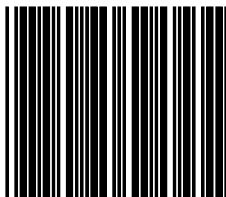
The **Header and Trailer** allows you to append a header and/or a trailer to every message transmitted via the serial ports ,USB or the keyboard port. There is no restriction in selecting header or trailer characters as far as the sum of the lengths of header and trailer is not greater than 10 digits.

1. Select either header or trailer you are going to program by scanning the corresponding label
2. Scan the character(s) you want from the enclosed ASCII table to set as header or trailer (be sure to enable full ASCII code 39 option before you start).
3. Read the “save setting to confirm” label to confirm your choice into memory.

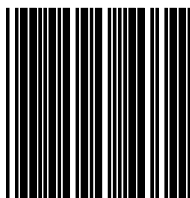
Header (Preamble)



Trailer (Postamble)



Save setting to confirm

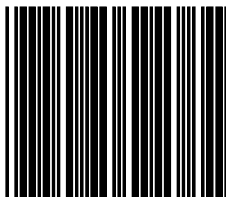


## **Truncate Header/trailer character**

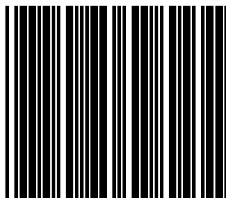
This setting allows you to truncate a number of header or trailer for symbology. When you do, the specific character you select is deleted from the symbology you want.

- 1.) Scan "Enter/Exit programming mode" label.
- 2.) Select "truncate header or truncate trailer" label.
- 3.) Scan two barcode value from the full ASCII code table(0~9). For example, if 2 number of header that you want to clear, then scan "0" and "2.
- 4.) Scan "save setting to confirm" label
- 5.) Scan "Enter/Exit programming mode" label to end of configuration.

Truncate header character



Truncate trailer character

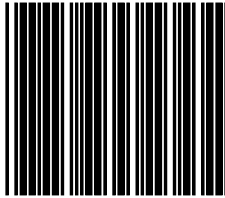




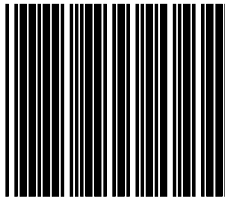
## Add Code Length

This option allows you to add the reading barcode numeric characters as header

Add code length as header enable(all barcode)



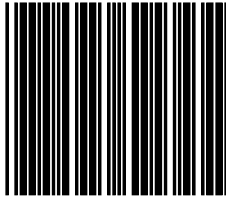
Add code length as header disable (all barcode)



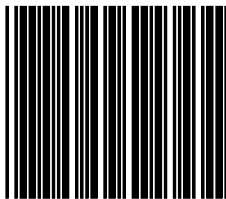
# ***Symbology Configuration***

The series scanner can be programmed to recognize one or more barcode symbologies automatically. If the scanner is configured to support multiple barcode symbologies, the scanner will discriminate different symbologies automatically, however, to improve scanning performance, you should enable only those symbologies that will be in active use.

Code 39 enable

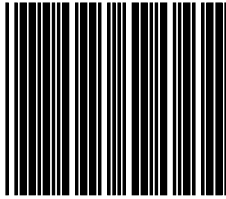


Code 39 disable

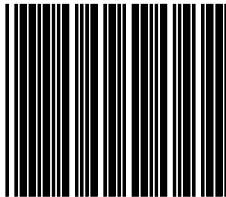




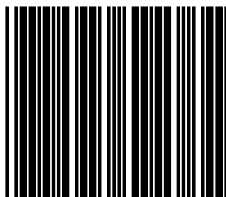
Codabar enable



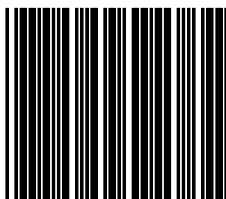
Codabar disable



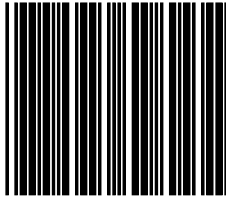
UPC/EAN/JAN enable



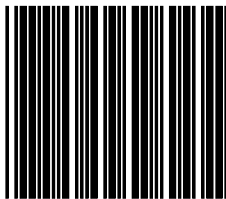
UPC/EAN/JAN disable  
(only can't transmitted but can decode)



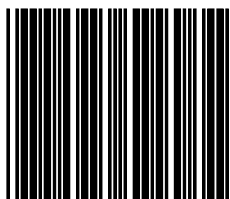
ITF 2 of 5 enable



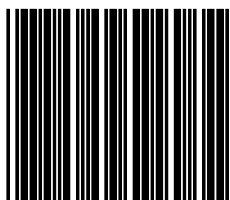
ITF 2 of 5 disable



Chinese postcode enable



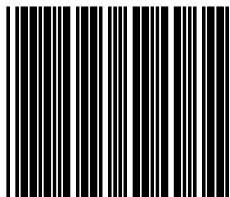
Chinese postcode disable



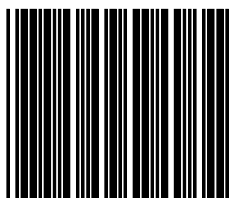




Code 128 enable

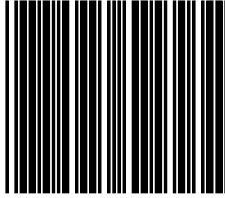


Code 128 disable

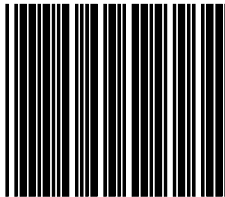




Code 93 enable

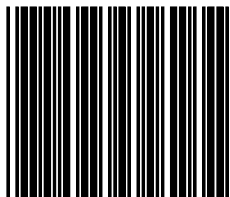


Code 93 disable

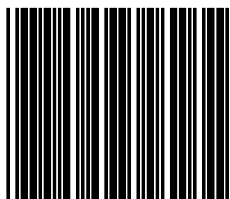




EAN-128 enable

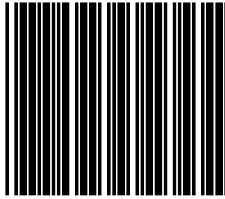


EAN-128 disable

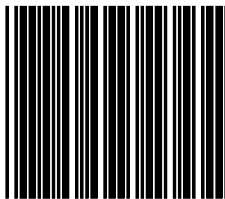




EAN convert to ISSN/ISBN

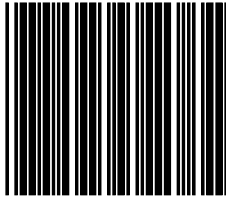


EAN convert to ISSN/ISBN disable

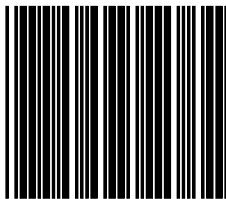




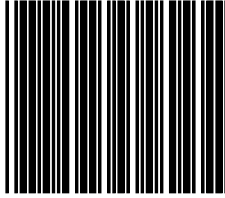
Code 32 enable



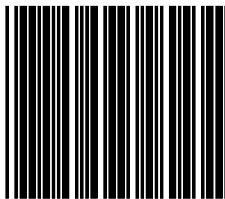
Code 32 disable



MSI enable



MSI disable





## **UPC/EAN Parameters setting**

The series scanner can be programmed to recognize some or all derivatives of UPC, EAN.

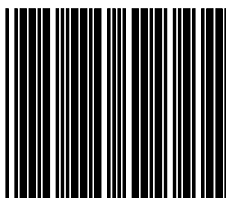
These derivatives are UPC-A, UPC-E, EAN-8, and EAN-13. Either 2 of 5 addendum digits are supported addendum digits are those additional digits after normal stop character.

The programming menu for UPC/EAN/JAN also provides several options to govern the transmission of scanned data.

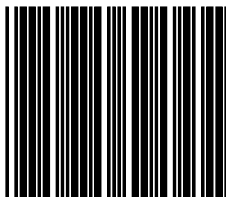
- \*UPC/EAN expansion
- \*Check digit transmission
- \*Data redundant check
- \*Addendum seek timeout
- \*Addendum left/right margin adjust

### **Format**

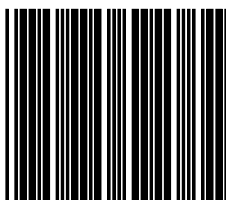
UPC/EAN/JAN all enable



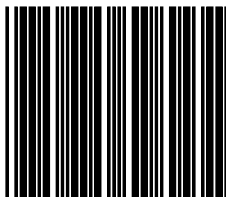
EAN-8 or EAN-13 enable



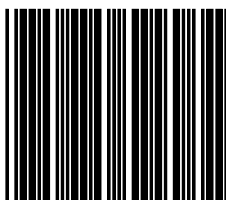
UPC-A and EAN-13 Enable



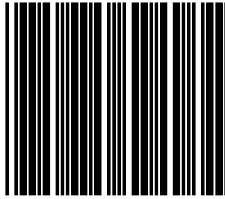
UPC-A and UPC-E Enable



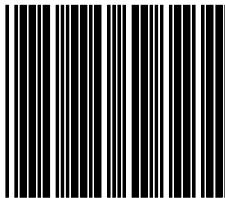
UPC-A enable



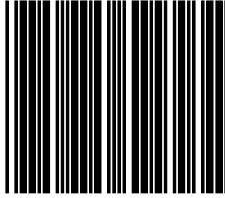
UPC-E enable



EAN-13 enable

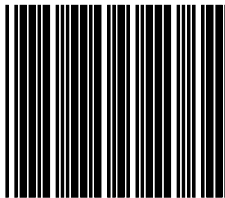


EAN-8 enable



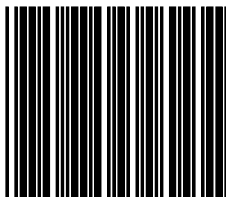
**Force UPC-E to UPC-A format**

Force UPC-E to UPC-A format enable



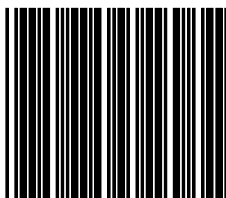


Force UPC-E to UPC-A format disable

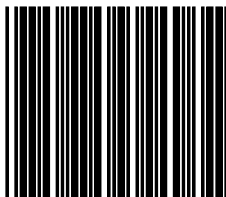


**Force UPC-A to EAN-13 format**

Force UPC-A to EAN-13 format enable

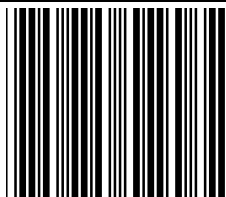


Force UPC-A to EAN-13 format disable

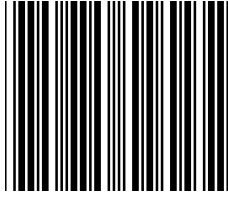


**Force EAN-8 to EAN-13 format**

Force EAN-8 to EAN-13 format disable

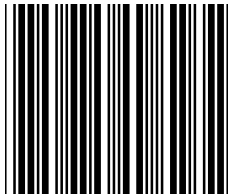


Force EAN-8 to EAN-13 format enable

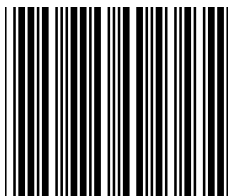


**EAN-13 country code with first “0” can transmitted**

EAN-13 first “0” can transmitted

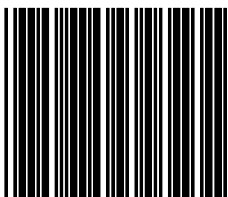


EAN-13 first "0" can't transmitted

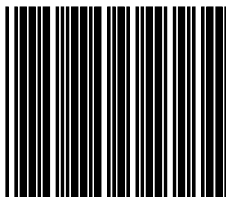


**Transmit UPC-A check digit**

Transmit UPC-A check digit enable

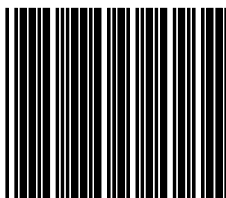


Transmit UPC-A check digit disable

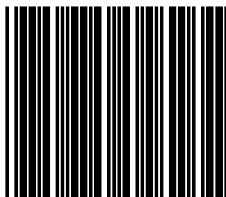


**Transmit UPC-E leading character**

Transmit UPC-E leading character enable

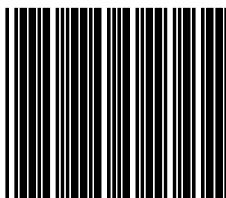


Transmit UPC-E leading character disable

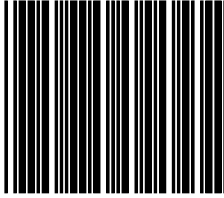


**Transmit UPC-E check digit**

Transmit UPC-E check digit enable

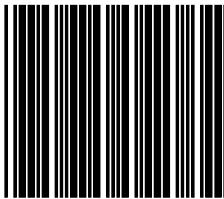


Transmit UPC-E check digit disable

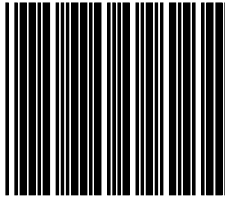


**Transmit EAN-8 check digit**

Transmit EAN-8 check digit enable

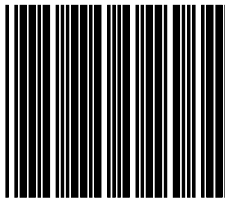


Transmit EAN-8 check digit disable



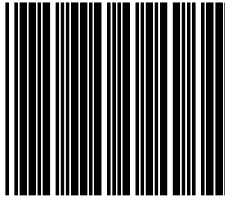
**Transmit EAN-13 check digit**

Transmit EAN-13 check digit enable



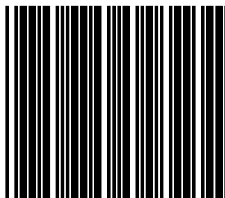


Transmit EAN-13 check digit disable

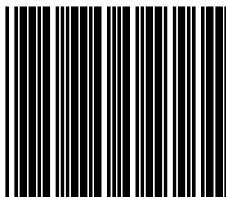


**Transmit UPC-A leading character**

Transmit UPC-A leading character enable

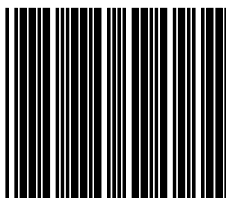


Transmit UPC-a leading character disable

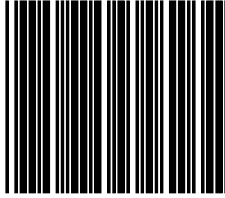


### **Addendum**

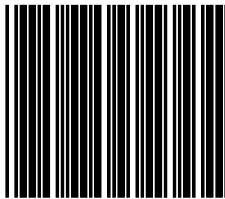
UPC/EAN add on off



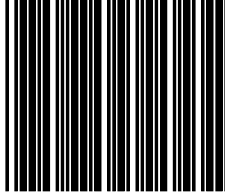
Add on 5 only



Add on 2 only

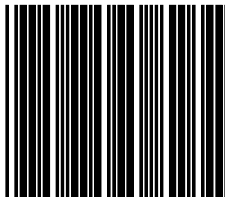


Add on 2 or 5

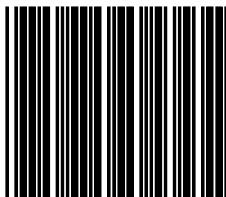


**Add on format**

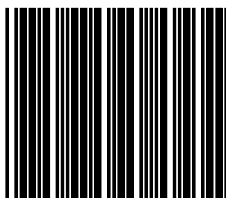
Add on format with separator



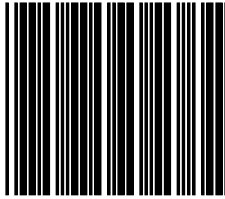
Add on format without separator



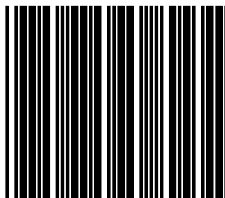
EAN/UPC +Add on (none mandatory)



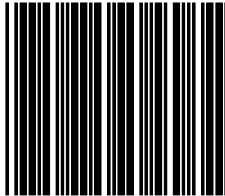
EAN/UPC + Add on(mandatory)



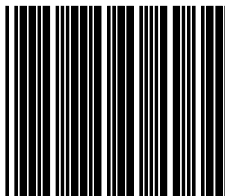
EAN/UPC + add on mandatory for 378/379 French  
Supplement requirement  
Not sent for other



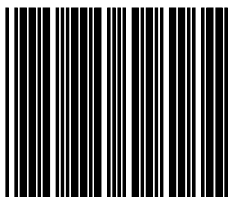
EAN/UPC +add on mandatory for 978/977  
book land Supplement requirements  
Not sent for other



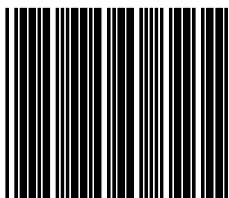
EAN/UPC + addon mandatory for 434/439  
German Supplement requirement  
Not sent for other



EAN/UPC + add on mandatory for 419/414  
Euro amounts Supplement requirement  
not sent for other



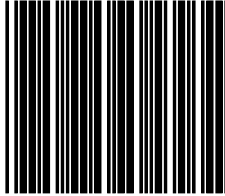
EAN/UPC + add on mandatory for 378/379  
French Supplement requirement  
optionally for other



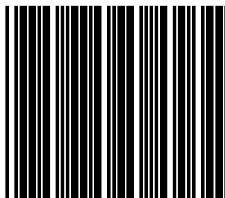




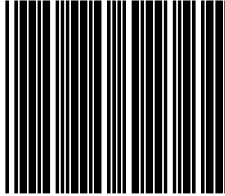
EAN/UPC + add on mandatory for 978/977  
Book land Supplement requirement  
optionally for other



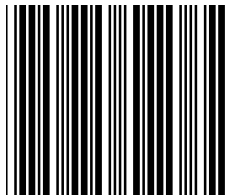
EAN/UPC + add on mandatory for 434/439  
German Supplement requirement  
optionally for other



EAN/UPC + add on mandatory for 419/414  
Euro amounts Supplement requirement  
optionally for other



EAN/UPC + add on mandatory for 491  
Japanese(bookland) Supplement requirement  
optionally for other

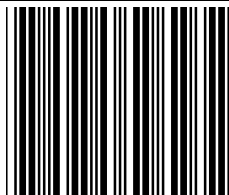


EAN/UPC + add on mandatory for 491  
Japanese(bookland) Supplement requirement

Not sent for other



Double code not allowed



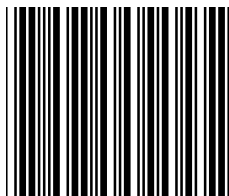
Double code mandatory for 978/192



Double code format without separator



double code format with separator



double code format with free (one character)

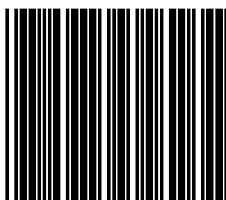


## **Data Redundant Check**

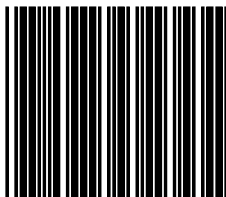
The option allows you to set decoder data redundant check. Before it is accept as a good read. A higher data redundant check read setting offers more assurance that a bar code has been read correctly, while a lower setting allows faster scanning performance.

### **UPC-A Data Redundant Check**

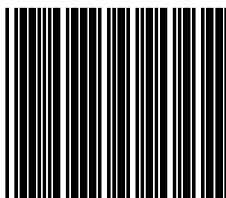
UPC-A data redundant check = 0



UPC-A data redundant check = 1

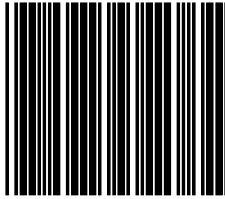


UPC-A data redundant check = 2



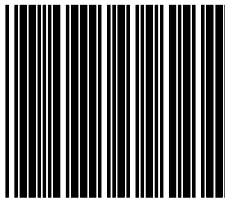


UPC-A data redundant check = 3

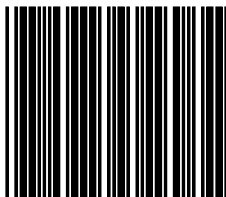


### **UPC-E Data Redundant Check**

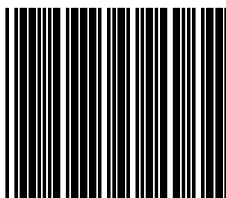
UPC-E data redundant check = 0



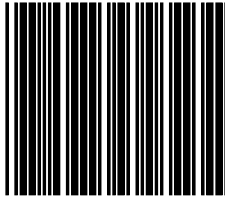
UPC-E data redundant check = 1



UPC-E data redundant check = 2

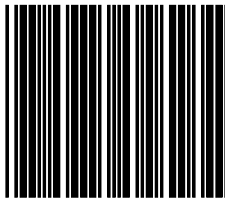


UPC-E data redundant check = 3

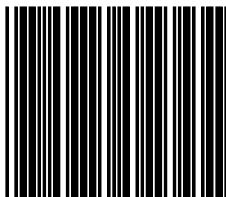


**EAN-13 Data Redundant Check**

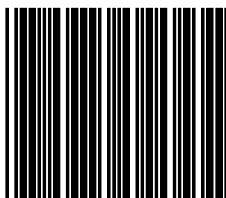
EAN-13 data redundant check = 0



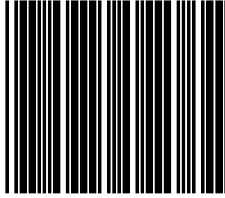
EAN- 13 data redundant check = 1



EAN-13 data redundant check = 2

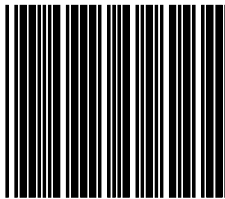


EAN-13 data redundant check = 3

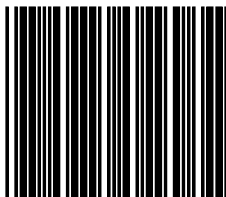


### **EAN-8 Data Redundant Check**

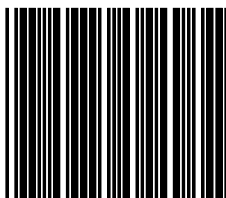
EAN-8 data redundant check = 0



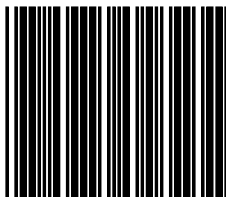
EAN-8 data redundant check = 1



EAN-8 data redundant check =2

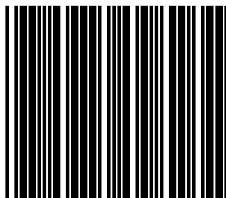


EAN-8 data redundant check = 3

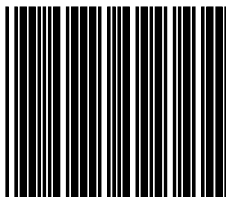


**2 digit addendum data redundant check**

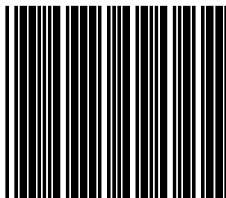
2 digit addendum data redundant check = 0



2 digit addendum data redundant check = 1

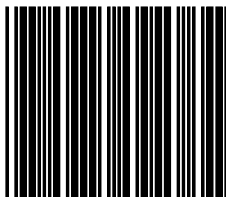


2 digit addendum data redundant check = 2



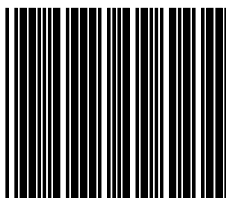


2 digit addendum data redundant check = 3

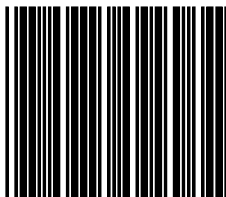


**5 digit addendum data redundant check**

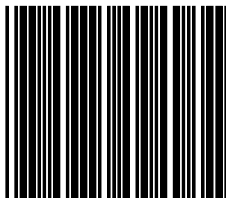
5 digit addendum data redundant check = 0



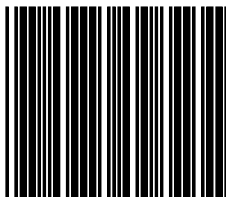
5 digit addendum data redundant check = 1



5 digit addendum data redundant check =2

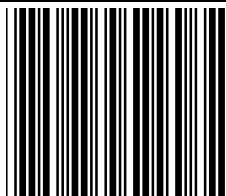


5 digit addendum data redundant check = 3

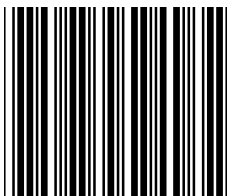


**Addendum seek timeout**

Addendum seek timeout =6



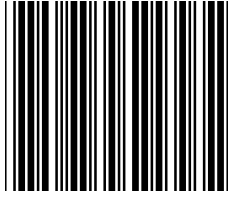
Addendum seek timeout=7



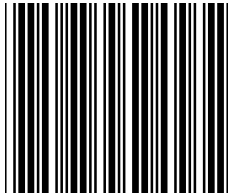
Addendum seek timeout=8



Addendum seek timeout=9

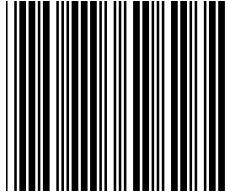


Addendum seek timeout=10

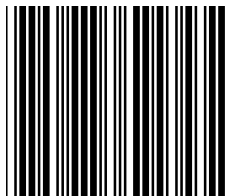


## UPC/EAN left/Right margin parameter setting

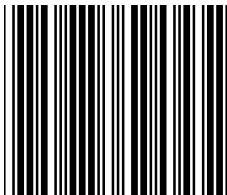
UPC/EAN addendum Left /Right margin required  
=15/15



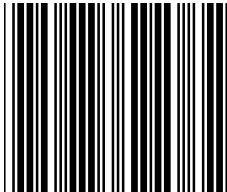
UPC/EAN addendum Left /Right margin  
required=5/10



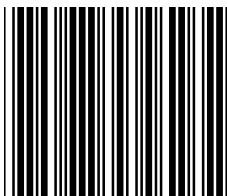
UPC/EAN addendum Left /Right margin required=1/5



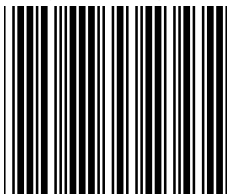
UPC/EAN addendum Left /Right margin required=0/0



UPC/EAN left /right margin required=15/15



UPC/EAN left /right margin required=10/10





UPC/EAN left /right margin required=7/7



UPC/EAN left /right margin required=5/5



UPC/EAN left /right margin required=2/2

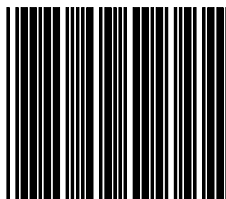


## **Code 39 Parameters setting**

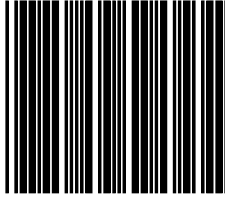
The scanner can program to support the standard code 39 or Full ASCII code 39. In addition, it is user's option to transmit or not to transmit the start and stop characters. You can also enable or disable the check digit feature. If the check digit feature is enabled, you have the further option to decide whether the check digit is transmitted or not.

### **Character Set**

Standard code 39

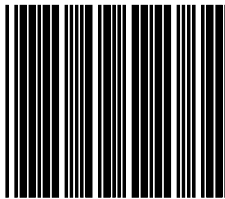


FULL ASCII code 39

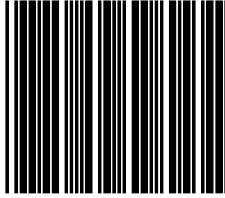


**Start/Stop Character Transmission**

Code 39 start/stop character transmission

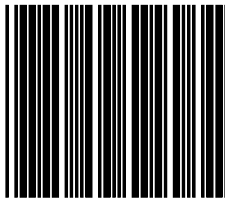


Code 39 start/stop character without transmission

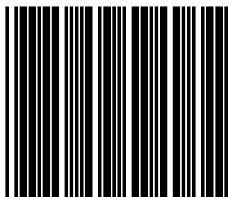


### Check Digit

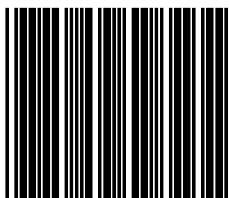
Code 39 check digit calculate and transmit



Code 39 check digit calculate but without transmit



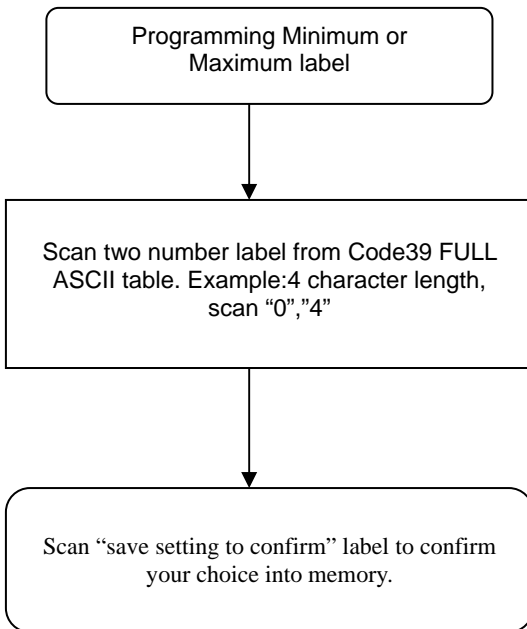
No check character



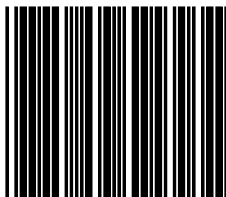
## **Code 39 reading length setting**

The default code 39 length is 3 ~32 character.  
It can be set at minimum 1 digit and maximum  
62 digits

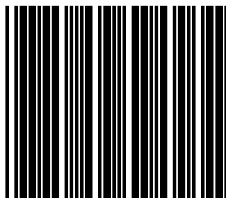
### **CODE LENGTH SETTING FLOW**



Code 39 maximum length setting

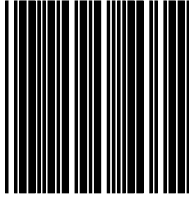


Code 39 minimum length setting



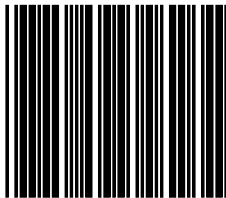


Save setting to confirm

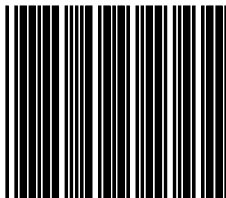


## Concatenation

Code 39 concatenation enable

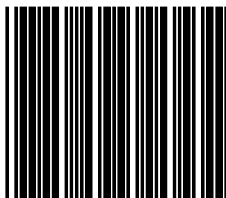


Code 39 concatenation disable

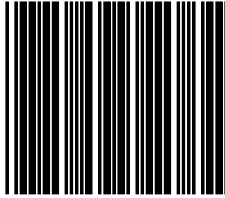


### Code 32”A “ Character Transmit

Code 32 (Italian pharmacy) transmit “A” character



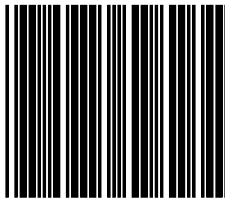
Code 32(Italian pharmacy)without transmit "A"  
character



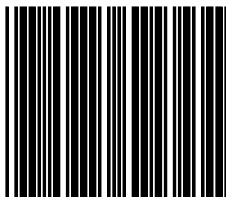
## **Data Redundant Check**

The option allows you to set decoder data redundant check. Before it is accept as a good read. A higher data redundant check read setting offers more assurance that a bar code has been read correctly, while a lower setting allows faster scanning performance.

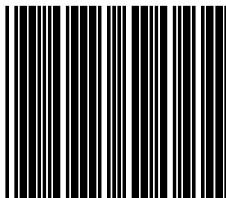
Code 39 data redundant check = 0



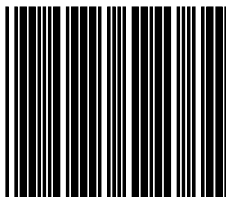
Code 39 data redundant check = 1



Code 39 data redundant check = 2



Code 39 data redundant check = 3



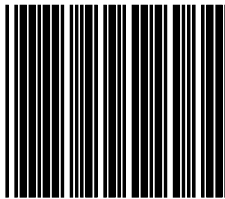
## CODABAR Parameters Setting

For this option, the Codabar symbology are contained

- Check Character Verification or Transmission
- CODABAR concatenation
- Data redundant check
- Start/Stop Characters
- Min./Max. length setting

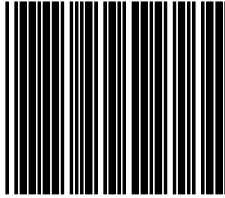
### Format

Codabar start/stop character transmission ----none





Codabar start/stop character transmission ----  
a/t,b/n,c/\*,d/e

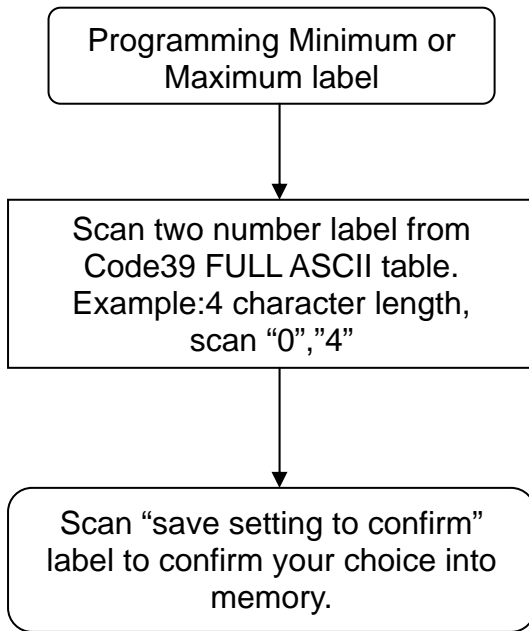




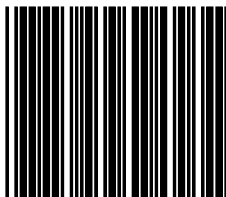
## CODABAR reading length setting

The default CODABAR length is 6 ~32 character. It can be set at minimum 1 digit and maximum 62 digits

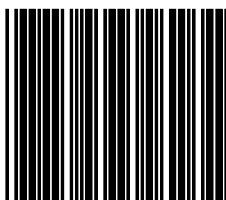
### CODE LENGTH SETTING FLOW



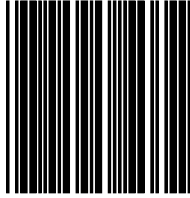
Codabar maximum length setting



Codabar minimum length setting

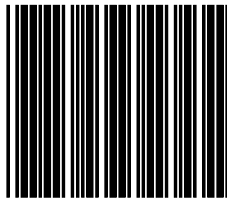


Save setting to confirm

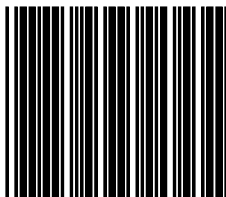


## **Concatenation**

Codabar concatenation disable

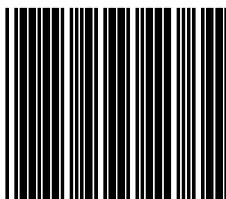


Codabar concatenation enable



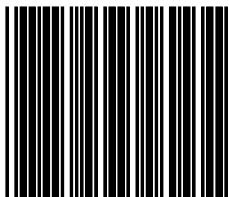
**Check digit**

No check character

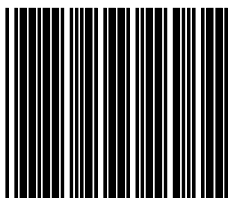




Check digits calculate but not transmit



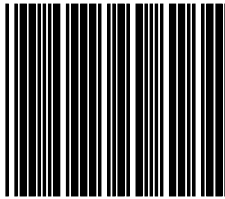
Check digit calculate and transmit



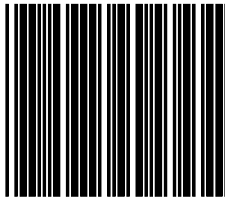
## **Data Redundant Check**

The option allows you to set decoder data redundant check. Before it is accept as a good read. A higher data redundant check read setting offers more assurance that a bar code has been read correctly, while a lower setting allows faster scanning performance.

Codabar data redundant check = 0



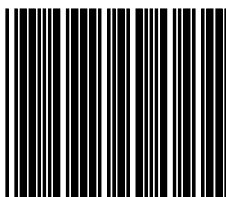
Codabar data redundant check = 1



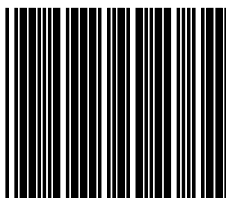




Codabar data redundant check = 2



Codabar data redundant check = 3



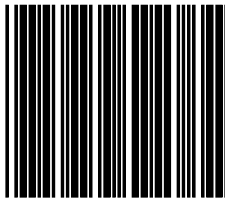
## Code 128 Parameters Setting

For this option the code 128 symbology are contained

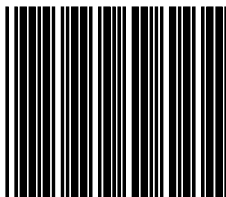
- Check Character Verification or Transmission
- FNC2 concatenation
- Data redundant check
- FNC1 transmission for EAN-128
- Min./Max. length setting

### Check Digit

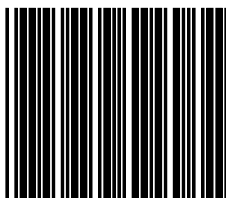
No check character



Calculate but not transmit



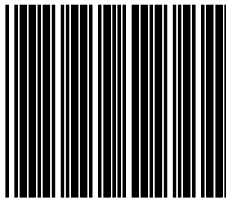
Calculate and transmit



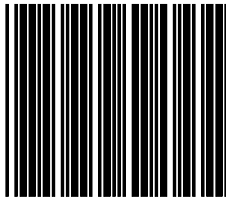
## Code 128 FNC2 concatenation

This function permits the temporary storage of a code in the decoder, if this code starts with FNC 2 character. The message buffered will be concatenated and transmitted with the next code having no FNC 2 character

Code 128 FNC2 concatenation enable



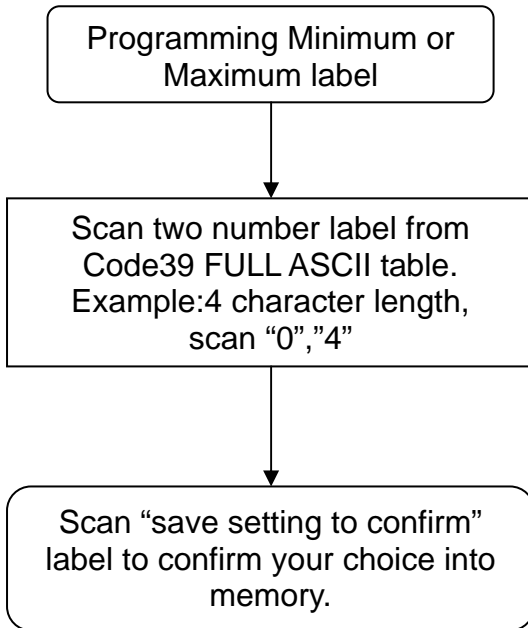
Code 128 FNC2 concatenation disable



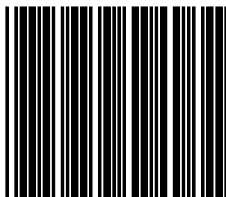
## Code 128 reading length setting

The default code 128 length is 3 ~62 character. It can be set at minimum 1 digit and maximum 62 digits

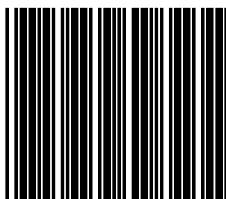
### CODE LENGTH SETTING FLOW



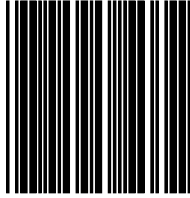
Code 128 maximum length setting



Code 128 minimum length setting

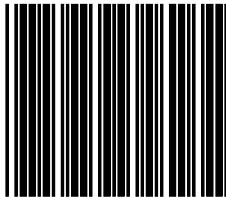


Save setting to confirm



**EAN-128 FNC1 Character**

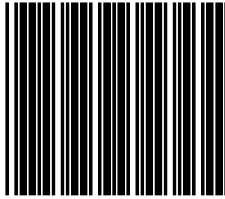
EAN-128 FNC1 Character transmitted







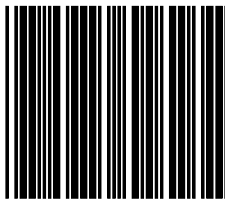
EAN-128 FNC1 not character transmitted



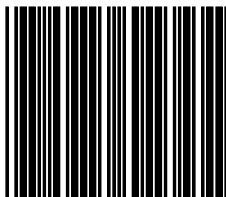
### **Data Redundant Check**

The option allows you to set decoder data redundant check. Before it is accept as a good read. A higher data redundant check read setting offers more assurance that a bar code has been read correctly, while a lower setting allows faster scanning performance

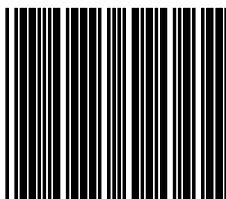
Code 128 data redundant check = 0



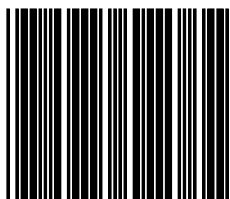
Code 128 data redundant check = 1



Code 128 data redundant check = 2



Code 128 data redundant check = 3



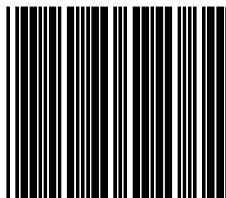
## **ITF 2 of 5 Parameters Setting**

For this option the ITF 2 of 5 symbology are contained

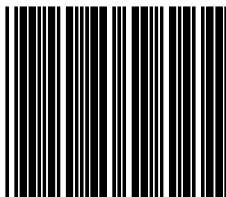
- Check Character Verification or Transmission
- Data redundant check
- Two fixed length setting
- Min./Max. length setting

### **Check Digit**

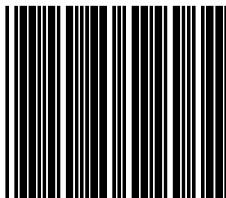
ITF 2 of 5 no check character



ITF 2 of 5 check digit calculate and transmit



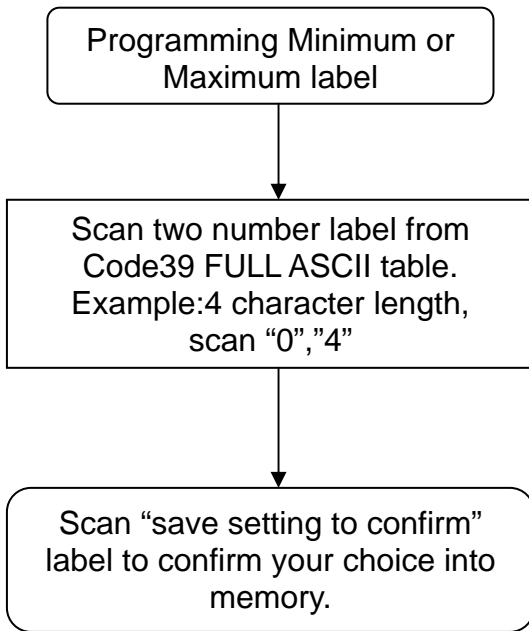
ITF 2 of 5 check digit calculate but without transmit



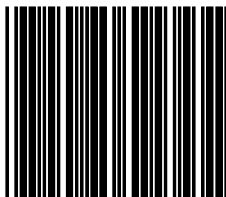
## ITF 2 of 5 reading length setting

The default ITF 2 of 5 length is 6 ~32 character. It can be set at minimum 2 digit and maximum 62 digits

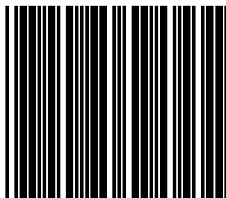
### CODE LENGTH SETTING FLOW



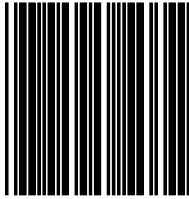
ITF 2 of 5 code maximum length setting



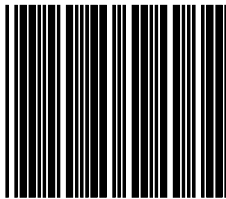
ITF 2 of 5 code minimum length setting



Save setting to confirm

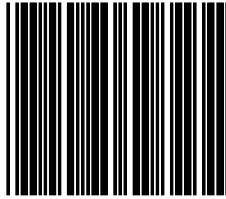


ITF 2 of 5 one fixed length setting





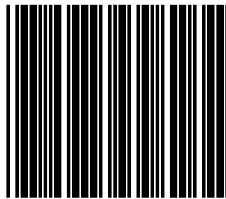
ITF 2 of 5 two fixed length setting



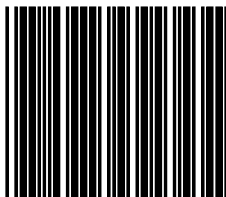
### **Data Redundant Check**

The option allows you to set decoder data redundant check. Before it is accept as a good read. A higher data redundant check read setting offers more assurance that a bar code has been read correctly, while a lower setting allows faster scanning performance

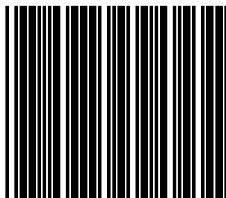
ITF 25 data redundant check =0



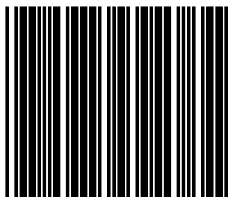
ITF 25 data redundant check = 1



ITF 25 data redundant check = 2



ITF 25 data redundant check = 3



## **Chinese Post Code Parameters Setting**

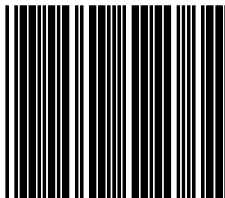
For this option the Chinese post code symbology are contained

- Check Character Verification or Transmission
- Data redundant check
- Min./Max. length setting

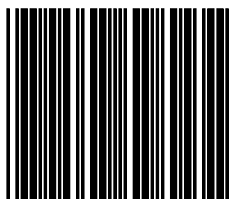
•

### **Check Digit**

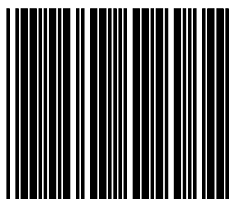
Chinese postcode no check digit



Chinese post code check digit calculate and transmit



Chinese postcode check digit calculate but without  
transmit

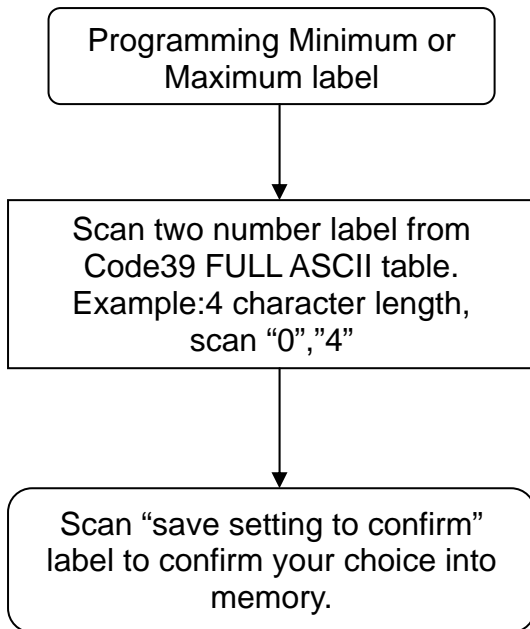


## Chinese postcode reading length setting

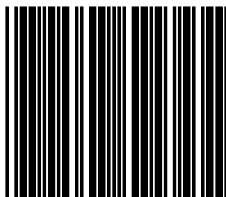
The default Chinese post code length is 10 ~32 character. It can be set at minimum 1 digit and maximum 62 digits.

**In order to avoid missing characters when scanning is incomplete, we recommend to use a short-range length or fixed length to read.**

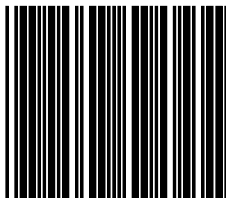
### CODE LENGTH SETTING FLOW



Chinese post code maximum length setting



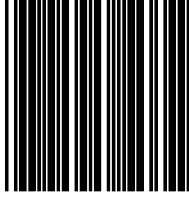
Chinese post code minimum length setting







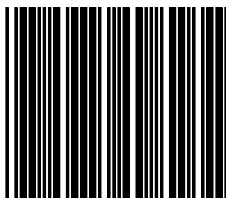
Save setting to confirm



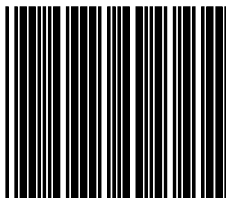
## Data Redundant Check

The option allows you to set decoder data redundant check. Before it is accept as a good read. A higher data redundant check read setting offers more assurance that a bar code has been read correctly, while a lower setting allows faster scanning performance

Chinese post code data redundant check = 0

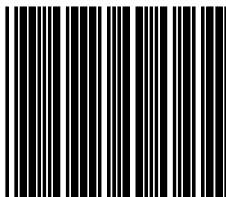


Chinese post code data redundant check = 1

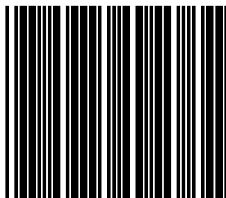




Chinese post code data redundant check = 2



Chinese post code data redundant check = 3



## **MSI/PLESSY code Parameters Setting**

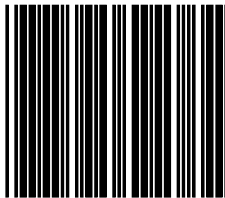
For this option the MSI/PLESSY symbology are contained

- Check Character Verification or Transmission
- Data redundant check
- Min./Max. length setting

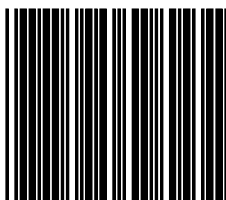
### **Double Check Digit**

MSI/PLESSY double check digit calculate but not

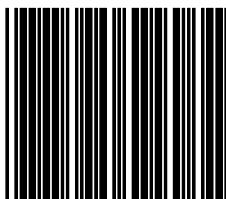
transmit



MSI/PLESSY double check digit without calculate  
and transmit

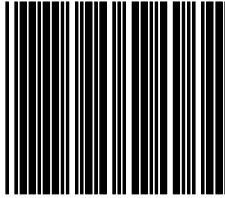


MSI/PLESSY double check digit calculate but only  
first digit transmit



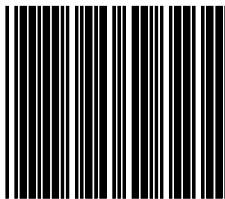


MSI/PLESSY double check digit calculate and both  
transmit



### **Single Check Digit**

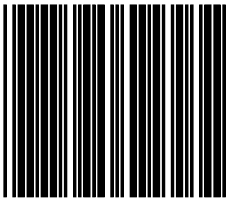
MSI/PLESSY single check digit calculate but without  
transmit







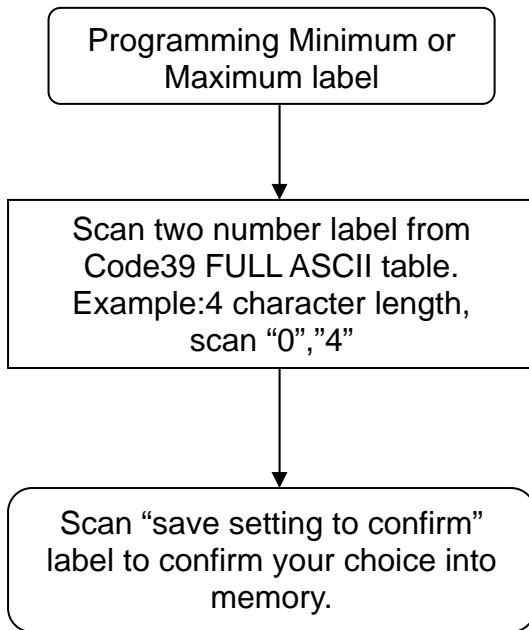
MSI/Plessey single check digit calculate and transmit



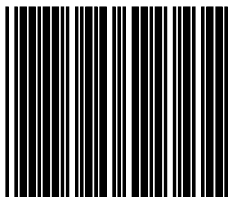
## MSI/PLESSY code reading length setting

The default MSI/PLESSY code length is 6~32 character. It can be set at minimum 1 digit and maximum 62 digits

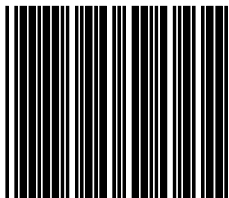
### CODE LENGTH SETTING FLOW



MSI/PLESSY maximum length setting

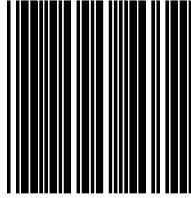


MSI/PLESSY minimum length setting





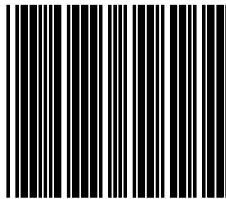
Save setting to confirm



### **Data Redundant Check**

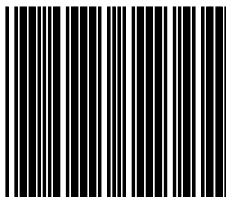
The option allows you to set decoder data redundant check. Before it is accept as a good read. A higher data redundant check read setting offers more assurance that a bar code has been read correctly, while a lower setting allows faster scanning performance

MSI data redundant check = 0

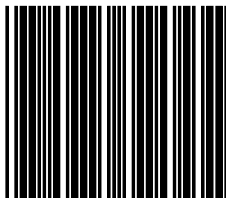




MSI data redundant check = 1



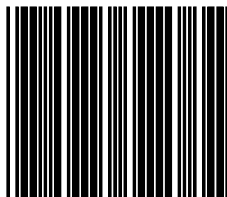
MSI data redundant check = 2







MSI data redundant check = 3



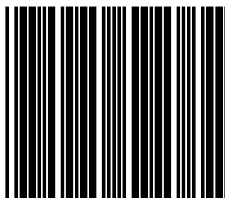
## **CODE 93 Parameters Setting**

For this option the CODE 93 symbology are contained

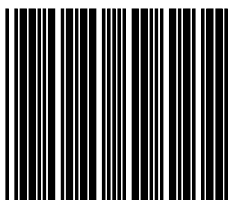
- Check Character Verification or Transmission
- Data redundant check
- Min./Max. length setting

### **Check Digit**

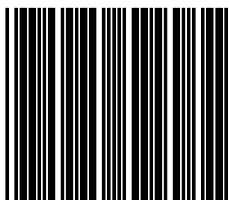
Code 93 check digit calculate but without transmit



Code 93 check digit not calculate and without  
transmit



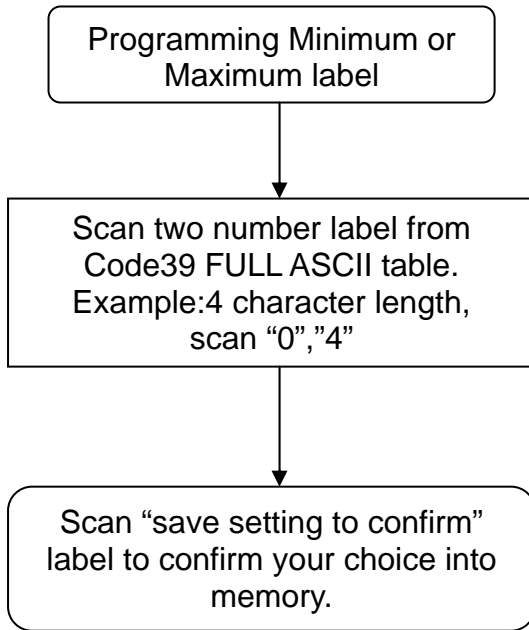
Code 93 check digit calculate and transmit



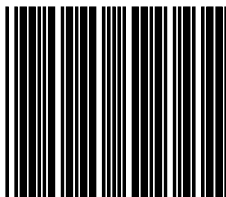
## CODE93 code reading length setting

The default CODE93 code length is 3 ~32 character. It can be set at minimum 1 digit and maximum 62 digits.

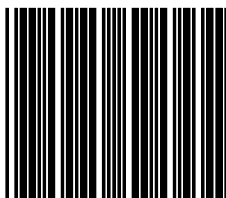
### CODE LENGTH SETTING FLOW



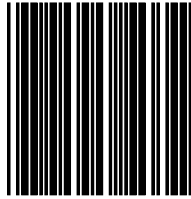
Code 93 maximum length setting



Code 93 minimum length setting



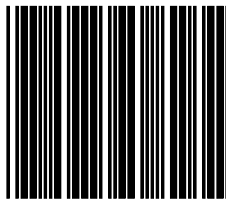
Save setting to confirm



## **Data Redundant Check**

The option allows you to set decoder data redundant check before it is accepted as a good read. A higher data redundant check read setting offers more assurance that a barcode has been read correctly, while a lower setting allows faster scanning performance.

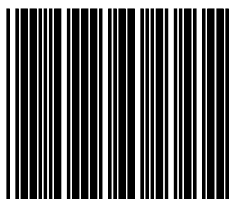
Code 93 data redundant check = 0



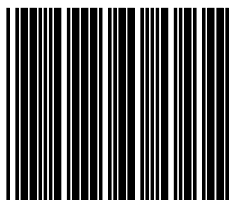




Code 93 data redundant check = 1

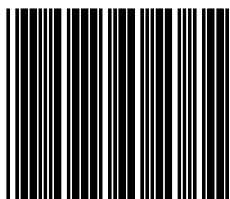


Code 93 data redundant check = 2



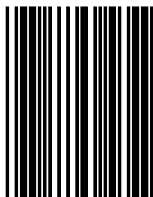


Code 93 data redundant check = 3

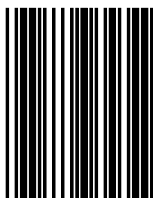


# Full ASCII Code Table

Full ASCII --- NUL

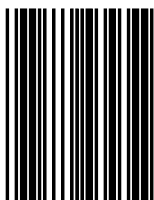


Full ASCII ---- SOH  
(Function Key---Ins)

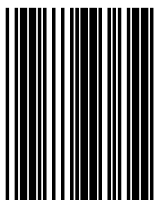




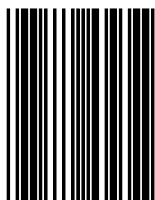
Full ASCII ----STX  
(Function Key---Del)



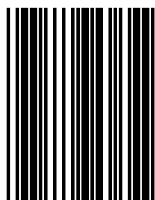
Full ASCII ---- ETX  
(Function Key---Home)



Full ASCII ---- EOT  
(Function Key---End)

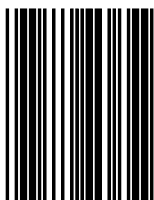


Full ASCII ---- ENQ  
(Function Key---Up arrow)

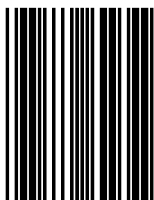




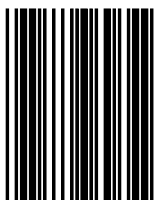
Full ASCII ---- ACK  
(Function Key---Down arrow)



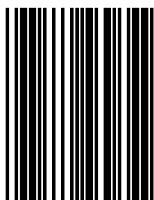
Full ASCII ---- BEL  
(Function Key---Left arrow)



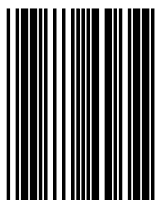
Full ASCII ---- BS  
(Function Key---Backspace)



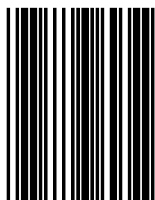
Full ASCII ---- HT  
(Function Key---Tab)



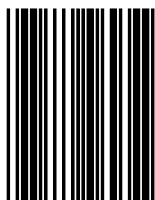
Full ASCII ---- LF  
(Function Key---Enter(num))



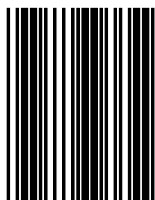
Full ASCII ---- VT  
(Function Key---Right arrow)



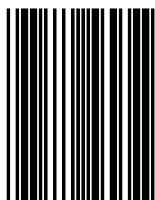
Full ASCII ---- FF  
(Function Key---PgUp)



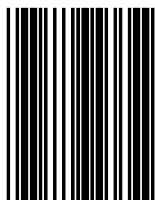
Full ASCII ---- CR  
(Function Key---Enter(alphabet))



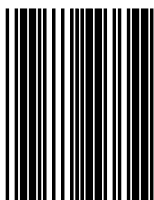
Full ASCII ---- SO  
(Function Key---PgDn)



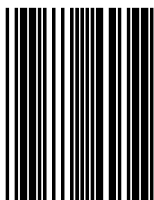
Full ASCII ---- SI  
(Function Key---Shift)



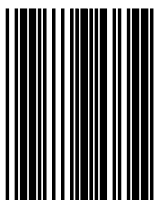
Full ASCII ---- DLE  
(Function Key---5(num))



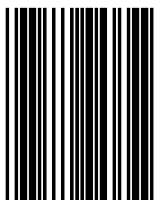
Full ASCII ---- DC1  
(Function Key---F1)



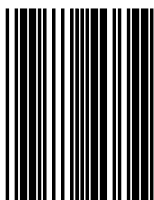
Full ASCII ---- DC2  
(Function Key---F2)



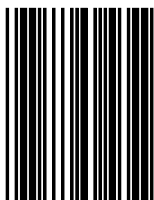
Full ASCII ---- DC3  
(Function Key---F3)



Full ASCII ---- DC4  
(Function Key---F4)

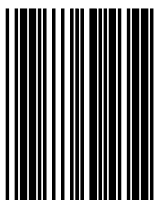


Full ASCII ---- NAK  
(Function Key---F5)

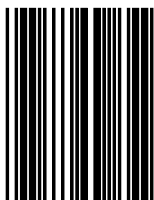




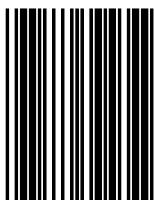
Full ASCII ---- SYN  
(Function Key---F6)



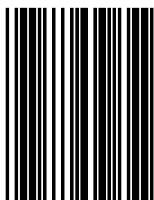
Full ASCII ---- ETB  
(Function Key---F7)



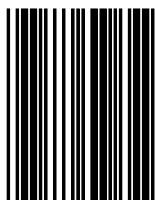
Full ASCII ---- CAN  
(Function Key---F8)



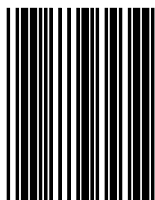
Full ASCII ---- EN  
(Function Key---F9)



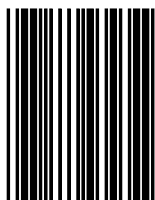
Full ASCII ---- SUB  
(Function Key---F10)



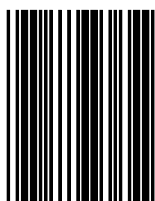
Full ASCII ---- ESC  
(Function Key---F11)



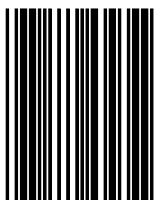
Full ASCII ---- FS  
(Function Key---F12)



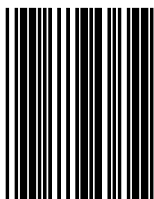
Full ASCII ---- GS  
(Function Key---ESC)



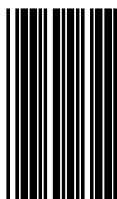
Full ASCII ---- RS  
(Function Key---Ctl(L))



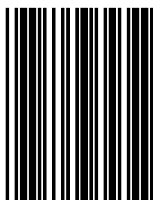
Full ASCII ---- US  
(Function Key---Alt(L))



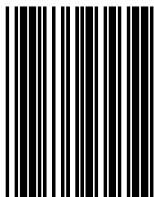
Full ASCII ---- SP



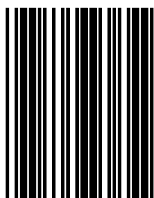
Full ASCII ---- !



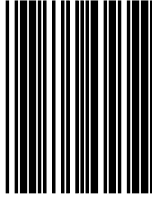
Full ASCII ---- “



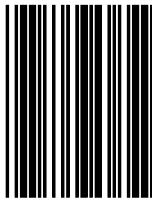
Full ASCII ---- #



Full ASCII ---- \$

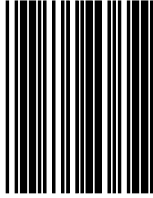


Full ASCII ---- %

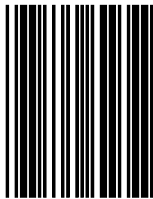




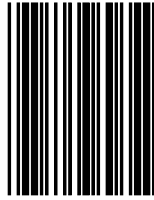
Full ASCII ---- &



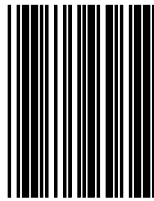
Full ASCII ---- ‘



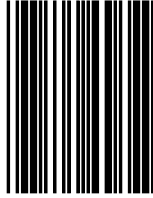
Full ASCII ---- (



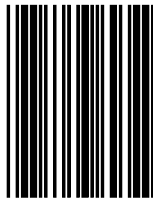
Full ASCII ---- )



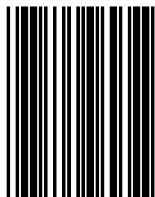
Full ASCII ---- \*



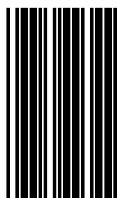
Full ASCII ---- +



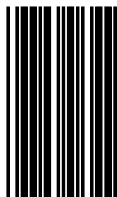
Full ASCII ---- ,



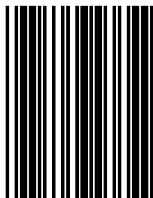
Full ASCII ---- -



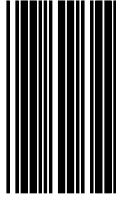
Full ASCII ---- .



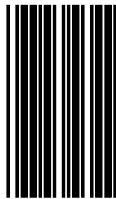
Full ASCII ---- /



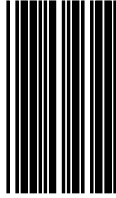
Full ASCII ---- 0



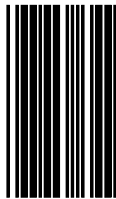
Full ASCII ---- 1



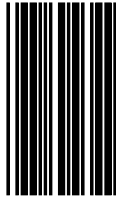
Full ASCII ---- 2



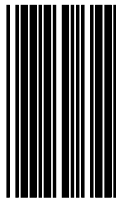
Full ASCII ---- 3



Full ASCII ---- 4

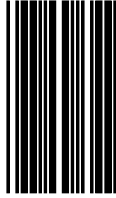


Full ASCII ---- 5

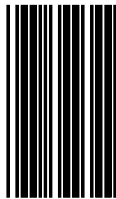




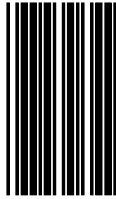
Full ASCII ---- 6



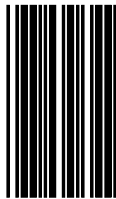
Full ASCII ---- 7



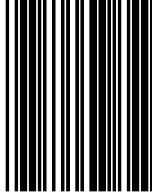
Full ASCII ---- 8



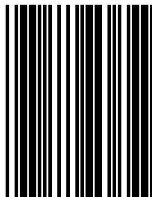
Full ASCII ----9



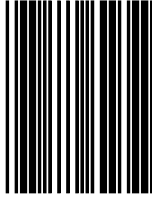
Full ASCII ---- :



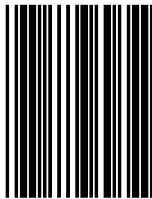
Full ASCII ---- ;



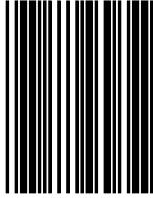
Full ASCII ---- <



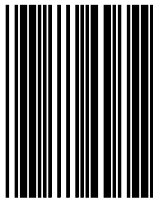
Full ASCII ---- =



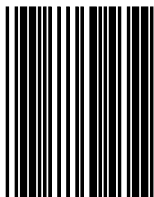
Full ASCII ---- >



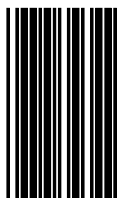
Full ASCII ---- ?



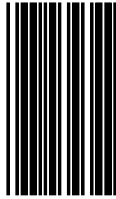
Full ASCII ---- @



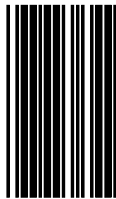
Full ASCII ---- A



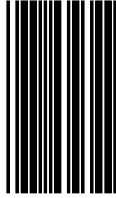
Full ASCII ---- B



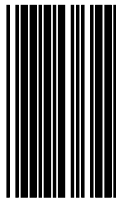
Full ASCII ---- C



Full ASCII ---- D

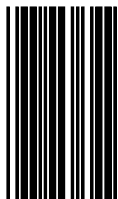


Full ASCII ---- E

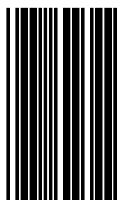




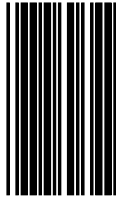
Full ASCII ---- F



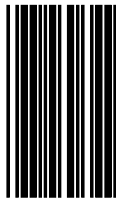
Full ASCII ---- G



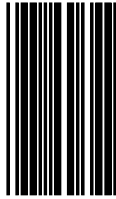
Full ASCII ---- H



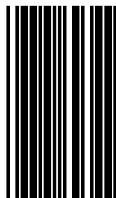
Full ASCII ---- I



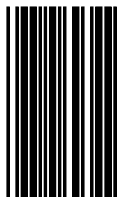
Full ASCII ---- J



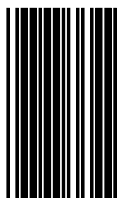
Full ASCII ---- K



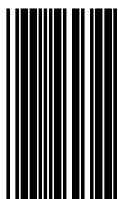
Full ASCII ---- L



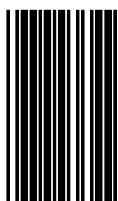
Full ASCII ---- M



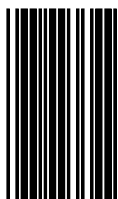
Full ASCII ---- N



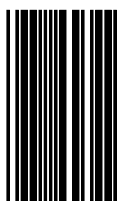
Full ASCII ---- O



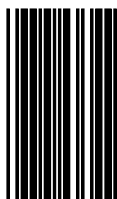
Full ASCII ---- P



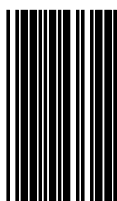
Full ASCII ---- Q



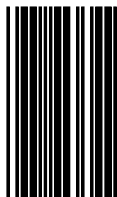
Full ASCII ---- R



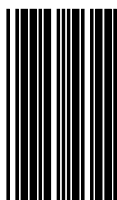
Full ASCII ---- S



Full ASCII ---- T

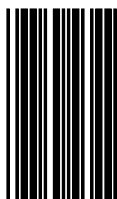


Full ASCII ---- U

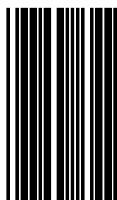




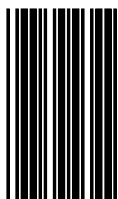
Full ASCII ---- V



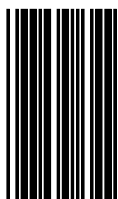
Full ASCII ---- W



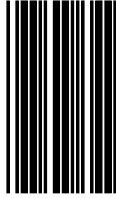
Full ASCII ---- X



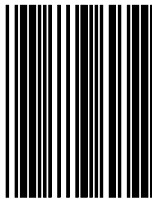
Full ASCII ---- Y



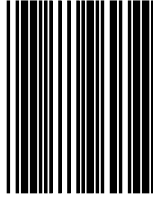
Full ASCII ---- Z



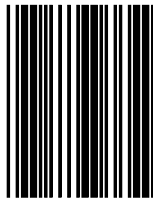
Full ASCII ---- [



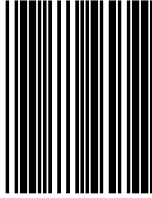
Full ASCII ---- \



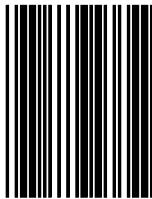
Full ASCII ---- ]



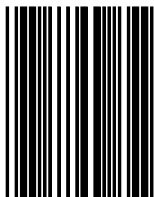
Full ASCII ---- ^



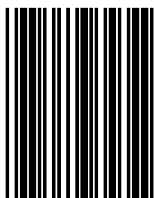
Full ASCII ---- \_



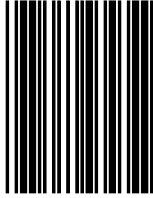
Full ASCII ---- `



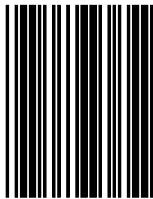
Full ASCII ---- a



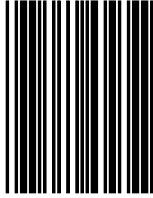
Full ASCII ---- b



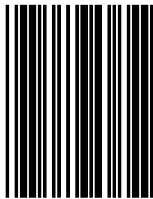
Full ASCII ---- c



Full ASCII ---- d

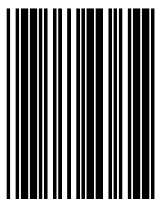


Full ASCII ---- e

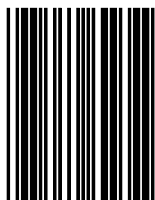




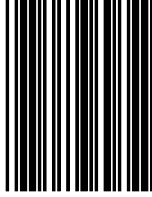
Full ASCII ---- f



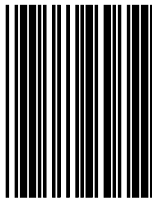
Full ASCII ---- g



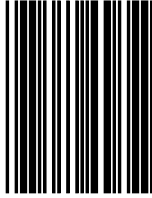
Full ASCII ---- h



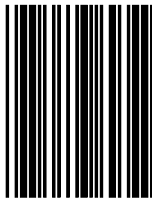
Full ASCII ---- i



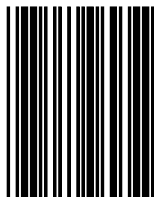
Full ASCII ---- j



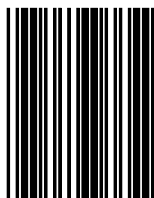
Full ASCII ---- k



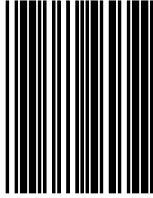
Full ASCII ---- I



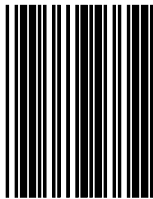
Full ASCII ---- m



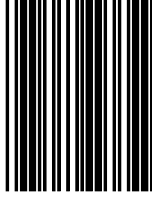
Full ASCII ---- n



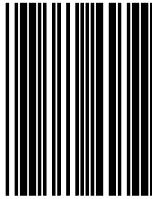
Full ASCII ---- o



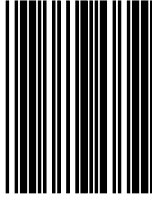
Full ASCII ---- p



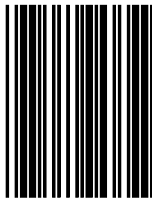
Full ASCII ---- q



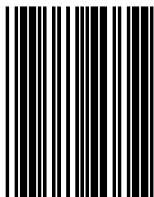
Full ASCII ---- r



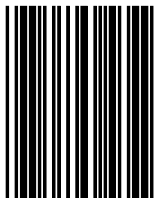
Full ASCII ---- s



Full ASCII ---- t

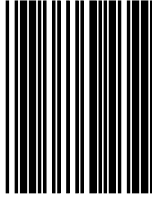


Full ASCII ---- u

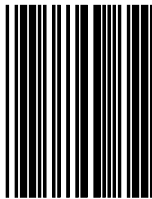




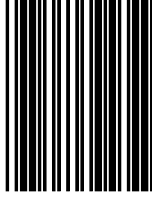
Full ASCII ---- v



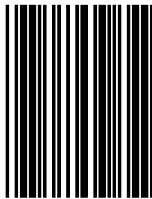
Full ASCII ---- w



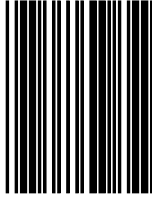
Full ASCII ---- x



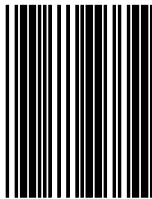
Full ASCII ---- y



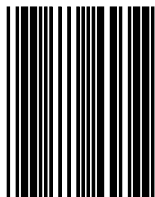
Full ASCII ---- z



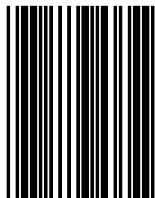
Full ASCII ---- {



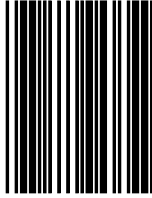
Full ASCII ---- |



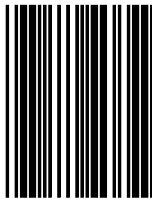
Full ASCII ---- }



Full ASCII ---- ~



Full ASCII ---- DEL



# APPENDIXES

## APPENDIX A

### CODE 39 FULL ASCII CODE TABLE

ASCII	CODE 39	VALEUR HEXA.	ASCII	CODE 39	VALEUR HEXA.
NUL	%U	00	%	/E	25
SOH	\$A	01	&	/F	26
STX	\$B	02	'	/G	27
ETX	\$C	03	(	/H	28
EOT	\$D	04	)	/I	29
ENQ	\$E	05	*	/J	2A
ACK	\$F	06	+	/K	2B
BEL	\$G	07	,	/L	2C
BS	\$H	08	-	-	2D
HT	\$I	09	.	.	2E
LF	\$J	0A	/	/	2F
VT	\$K	0B	0	0	30
FF	\$L	0C	1	1	31
CR	\$M	0D	2	2	32
SO	\$N	0E	3	3	33
SI	\$O	0F	4	4	34
DLE	\$P	10	5	5	35
DC1	\$Q	11	6	6	36
DC2	\$R	12	7	7	37
DC3	\$S	13	8	8	38
DC4	\$T	14	9	9	39
NAK	\$U	15	:	/Z	3A
SYN	\$V	16	;	%F	3B
ETB	\$W	17	<	%G	3C
CAN	\$X	18	=	%H	3D
EM	\$Y	19	>	%I	3E
SUB	\$Z	1A	?	%J	3F
ESC	%A	1B	@	%V	40
FS	%B	1C	A	A	41
GS	%C	1D	B	B	42
RS	%D	1E	C	C	43
US	%E	1F	D	D	44
SP	SP	20	E	E	45
!	/A	21	F	F	46
"	/B	22	G	G	47
#	/C	23	H	H	48
\$	/D	24	I	I	49

## APPENDIX A

### CODE 39 FULL ASCII CODE TABLE

ASCII	CODE 39	VALEUR HEXA.	ASCII	CODE 39	VALEUR HEXA.
J	J	4A	e	+E	65
K	K	4B	f	+F	66
L	L	4C	g	+G	67
M	M	4D	h	+H	68
N	N	4E	i	+I	69
O	O	4F	j	+J	6A
P	P	50	k	+K	6B
Q	Q	51	l	+L	6C
R	R	52	m	+M	6D
S	S	53	n	+N	6E
T	T	54	o	+O	6F
U	U	55	p	+P	70
V	V	56	q	+Q	71
W	W	57	r	+R	72
X	X	58	s	+S	73
Y	Y	59	t	+T	74
Z	Z	5A	u	+U	75
[	%K	5B	v	+V	76
\	%L	5C	w	+W	77
]	%M	5D	x	+X	78
^	%N	5E	y	+Y	79
_	%O	5F	z	+Z	7A
`	%W	60	{	%P	7B
a	+A	61		%Q	7C
b	+B	62	}	%R	7D
c	+C	63	~	%S	7E
d	+D	64	DEL	%T	7F



## APPENDIX B

### FUNCTION KEY EMULATION

FUNCTION KEY	ASCII	CODE 39	FUNCTION KEY	ASCII	CODE 39
Ins	\$A	01	F1	\$Q	11
Del	\$B	02	F2	\$R	12
Home	\$C	03	F3	\$S	13
End	\$D	04	F4	\$T	14
Up	\$E	05	F5	\$U	15
Down	\$F	06	F6	\$V	16
Left	\$G	07	F7	\$W	17
Backspace	\$H	08	F8	\$X	18
TAB	\$I	09	F9	\$Y	19
Enter(num)	\$J	0A	F10	\$Z	1A
Right	\$K	0B	F11	%A	1B
PgUp	\$L	0C	F12	%B	1C
Enter	\$M	0D	ESC	%C	1D
PgDn	\$N	0E	Ctl(L)	%D	1E
shift	\$O	0F	Alt(L)	%E	1F
5 (num)	\$P	10			

Enter/Exit programming

