Code 128 Symbology Introduction

Version:v1.1

Crifan Li

Abstract

This article explains the details of code 128 symbology.

This article provide multiple formats:

Reading Online	HTML [1]	HTMLs [2]	PDF [3]	CHM [4]	TXT [5]	RTF [6]
Download(7Zip package)	HTML [7]	HTMLs [8]	PDF [9]	CHM [10]	TXT [11]	RTF [12]

HTML version online address:

http://www.crifan.com/files/doc/docbook/symbology_code128/release/html/symbology_code128.html

Any question or bug report are encouraged to post in the discussion group:

http://www.crifan.com/bbs/categories/symbology_code128/

Revision History		
Revision 1.0	2010-12-20	crl
1. explain the code 128 symboloy		
Revision 1.1	2012-08-09	crl
1. Release via Docbook		

[1] http://www.crifan.com/files/doc/docbook/symbology_code128/release/html/symbology_code128.html
[2] http://www.crifan.com/files/doc/docbook/symbology_code128/release/htmls/index.html
[3] http://www.crifan.com/files/doc/docbook/symbology_code128/release/pdf/symbology_code128.pdf
[4] http://www.crifan.com/files/doc/docbook/symbology_code128/release/chm/symbology_code128.chm
[5] http://www.crifan.com/files/doc/docbook/symbology_code128/release/txt/symbology_code128.txt
[6] http://www.crifan.com/files/doc/docbook/symbology_code128/release/rtf/symbology_code128.rtf
[7] http://www.crifan.com/files/doc/docbook/symbology_code128/release/html/symbology_code128.html.7z
[8] http://www.crifan.com/files/doc/docbook/symbology_code128/release/htmls/index.html.7z
[9] http://www.crifan.com/files/doc/docbook/symbology_code128/release/pdf/symbology_code128.pdf.7z
[10] http://www.crifan.com/files/doc/docbook/symbology_code128/release/chm/symbology_code128.chm.7z
[11] http://www.crifan.com/files/doc/docbook/symbology_code128/release/txt/symbology_code128.txt.7z
[12] http://www.crifan.com/files/doc/docbook/symbology_code128/release/rtf/symbology_code128.rtf.7z

Code 128 Symbology Introduction:

Crifan Li

Version:v1.1

Publication date 2012-08-09

[13] http://www.crifan.com/files/doc/docbook/soft_dev_basic/release/html/soft_dev_basic.html#cc_by_nc

Table of Contents

List of Figures

List of Tables

Glossary

Value (Value)	For the value column in Table 3.1, "Encoding Code Table", the "value" is also called code, or character, or digit. So actually: Start code = start character Check character = check digit Function code = function character .....
Variant (Variant)	The variant is just the character set, so variant A/B/C is just the Character Set A/B/C
UCC/EAN 128 (UCC/EAN 128)	This item means the symboloy for UCC 128 or EAN 128, often can be write like this: UCC-128 or EAN-128, UCC/EAN-128, UCC.EAN 128, ... It is just a variant of Code 128. More please refer to Section 2.8, "Other related info"

Chapter 1. Code 128 Definition

1.1. Symbology

Code 128 is a variable length, high density, alphanumeric symbology.

It is a very effective, high-density symbology which encoding of alphanumeric data.

Note

high-density: the high density means: when encoding the numeric data, then using Code set C, which using one code to represent two digital data. While normally using one code represent one digit/data.

1.2. Checksum Digit

Code 128 also employs a check digit for data security.

Code 128 symbology must include a checksum digit.

1.3. Character set

Code 128 has three characters Set: A, B, C.

Code 128 has 106 different bar and space patterns and each pattern can have one of three different meanings, depending on which of three different character sets (characters Set A/B/C) is employed.

Three different start characters tell the reader the character set used in the following encoding, and three shift codes permits changing character set inside the symbol.

So, sometime you can see some item like this:

- Code 128 - A, Code 128 - B, Code 128 - C
- Code128-A, Code128–B, Code128–C
- Code128A, Code128B, Code128C

Chapter 2. Code 128 Characteristics

2.1. Start Character

As you can see in Table 3.1, "Encoding Code Table"

The three start characters are:

"START A"=103, "START B"=104, "START C"=105

2.2. Code and Shift

As you can see in Table 3.1, "Encoding Code Table", three CODEs are: "Code A", "Code B", "Code C".

Within a symbol, one can shift between code sets by using the special character CODE and SHIFT.

The CODE character shifts the code for all subsequent characters to the specified code set.

The SHIFT character just changes the next character and only changes between Code Set A and Code Set B or the reverse.

So, within the data in Code 128, changes to other character Set:

For character set A: change to character set B is "Code B=100", to character set C is "Code C=99"

For character set B: change to character set A is "Code A=101", to character set C is "Code A=99"

For character set C: change to character set A is "Code B=101", to character set C is "Code B=100"

The example of changing to Character Set C in Character Set A using 99 = "Code C", can be found in later Section 2.5, "Calculate the Check/Checksum Digit".

2.3. Character set usage domain

Generally:

Character Set A: encodes all upper case ('A=33' to 'Z=58') and ASCII control characters ('NUL=64' to 'US=95');

Character Set B: encodes all upper ('A=33' to 'Z=58') and lower ('a=65' to 'z=90') case characters;

Character Set C: encodes numeric digit pairs 00 through 99 ('00=00' to '99=99').

Character Set A and B: used to encode alpha-numeric information

Character Set C : offers double density when being used to encode numeric data, which means use a single value/code, decoded as two digital value/data, that is one value = two data.

This third character set - Character Set C, effectively doubles the code density when printing numeric data.

2.4. Function Code

In addition to ASCII characters, Code 128 also allows encoding of four special function codes (FNC1 - FNC4), which are listed in Table 3.1, "Encoding Code Table".

The meaning of function code FNC1 and FNC4 were originally left open for application specific purposes.

2.4.1. FNC1

Recently an agreement was made by the Automatic Identification Manufacturers Assoc. (AIM) and the European Article Numbering Assoc. (EAN) to reserve FNC1 for use in EAN applications.

FNC1 at the beginning of a bar code indicates that it begins with a 2- 3- or 4-digit application identifier assigned by the Uniform Code Council, which explains the following digits. For example, application identifier 421 indicates that an ISO 3166-1 numeric country code and ship-to postal code follows.

For example, the U.S. ZIP code for the White House would generally be printed as "(421) 840 20500", but would actually be coded as "[Start C] [FNC1] 42 18 40 20 50 [Code A] 0 [Check symbol 80] [Stop]"

The FNC1 for the UCC/EAN-128

The function 1 (FNC 1), which follows the start character, enables scanners and processing software to auto-discriminate between UCC/EAN-128 and other barcode symbologies, and subsequently only process relevant data.

2.4.2. FNC2

FNC2 is used to instruct a bar code reader to concatenate the message in a bar code symbol with the message in the next symbol.

2.4.3. FNC3

FNC3 is used to instruct a bar code reader to perform a reset. When FNC3 is encoded anywhere in a symbol, any data also contained in the symbol is discarded.

2.4.4. FNC4

FNC4 remains available for use in closed system applications

2.5. Calculate the Check/Checksum Digit

Before a Code 128 symbol may be encoded, the software must compute the correct checksum digit which will be included in the bar code. The checksum digit is based on Modulus 103 Checksum based on the weighted sum of the values of each of the digits in the message that is being encoded, including the start character.

The steps for calculating the check digit are as follows:

1. Take the value of the start character ("START A"=103, or "START B"=104, or "START C"=105) and make that the starting value of the running checksum.
2. Starting with the first data character following the start character, take the value of the character (between 0 and 102, inclusive) multiply it by its character position (1) and add that to the running checksum.
3. Take each additional character in the message, take its value, and multiply it by its character position, and add the total to the running checksum
4. use the resulting to MOD 13, that is divide the resulting running checksum by 103, then the remainder is the checksum digit, which is added to the end of the message

5. The stop character is appended after the checksum digit.

This is easier to understand with an example.

Let's calculate the checksum digit for the sample bar code above, "HI345678".

The checksum digit is included in all Code 128 bar codes, but it isn't printed as part of the text below the bar code symbol (as is the case with UPC and EAN symbols).

Table 2.1. Calculate checksum digit of "HI345678"

Barcode	START-A	H	I	CODE C	34	56	78
Character Value	103	40	41	99	34	56	78
Character Position	-	1	2	3	4	5	6
Calculation	103	40 * 1	41 * 2	99 * 3	34 * 4	56 * 5	78 * 6
Weighted Sum	103	40	82	297	136	280	468

Summing up the running checksum for each digit, we get:

103 + 40 + 82 + 297 + 136 + 280 + 468 = 1406

Then,

1046 MOD 13 = the remainder of 1406 / 103 = 67

Thus the checksum digit is the character which has a value of 67.

Note

1. Note that the checksum starts with the first Start Character, with a weight of 1, and that the first data character also has a weight of 1.
2. In my understanding, the above "CODE-C=99" is the generated barcode, but not displayed/printed, just like the check digit.

2.6. Structure / Composition of Code 128

Table 2.2. Structure of Code 128

one of three start codes "START "=103, or "START B"=104, or "START C"=105	the data itself	a check character	a stop character	Termination bar of 11

We will now code the above example, HI345678, in Code 128. As we calculated in the Checksum Digit Calculation section, the checksum digit is 67. So we must also code the checksum digit at the end of the message.

We encode each digit using the encoding table above:

1. The START-A character: 11010000100
2. The digit "H" encoded as: 11000101000
3. The digit "I" encoded as: 11000100010
4. The "CODE-C" character: 10111011110

5. The digits "34" encoded as: 10001011000
6. The digits "56" encoded as: 11100010110
7. The digits "78" encoded as: 11000010100
8. The checksum digit of 67 encoded as: 10000101100
9. The STOP character: 11000111010
10. The termination bar: 11

This is shown in the following graphical representation where the bar code has been sectioned-off into areas that reflect each of the 10 components just mentioned.

Figure 2.1. Structure of Code Example for HI345678

Note

Also others describe the structure Code 128 like this:

Table 2.3. Structure of Code 128 with quiet zone

a leading quiet zone	one of three start codes "START "=103, or "START B"=104, or "START C"=105	the data itself	a check character	a stop character	a trailing quiet zone

Then this picture illuminates it well:

Figure 2.2. Structure of Code 128 with quiet zone

2.7. Compare to Other symbology

For Code 39, Code 93, Code 128, EAN /UCC 128:

There are all variable length alphanumeric symbology.

And their "Data Capacity" - the practical upper limit, dependent on the scanner and is typically between 20 and 40 characters.

Code 128 is more efficient at encoding data than Code 39 or Code 93.

Code 128 is the best choice for most general bar code applications.

Code 39 and Code 128 are both very widely used while Code 93 is rarely used.

2.8. Other related info

GS1-128, formerly called UCC/EAN-128 (UCC-128 or EAN-128) only can encode numeric data using the Code 128 Character Set C.

UCC/EAN-128 starts with character C, follow FNC1 character(102) and other data.

Chapter 3. Append

3.1.

This table indicates how to encode each digit of a Code 128 bar code.

Note that it is easiest to think of each character as a value between 0 and 105, inclusive, rather than thinking of them as characters.

The character that a value represents depends on what mode (or character set) you're in-so rather than thinking of a character as "A" or "B", etc. it is more appropriate to think of it as 33, 34, etc.

Table 3.1. Encoding Code Table

VALUE	WHICH REPRESENTS IN CHARACTER SET			ENCODING	VALUE	WHICH REPRESENTS IN CHARACTER SET			ENCODING
	A	B	C			A	B	C	
00	SP	SP	00	11011001100	53	U	U	53	11011101110
01	!	!	01	11001101100	54	V	V	54	11101011000
02	"	"	02	11001100110	55	W	W	55	11101000110
03	#	#	03	10010011000	56	X	X	56	11100010110
04	$	$	04	10010001100	57	Y	Y	57	11101101000
05	%	%	05	10001001100	58	Z	Z	58	11101100010
06	&	&	06	10011001000	59	[	[	59	11100011010
07	'	'	07	10011000100	60	\	\	60	11101111010
08	(	(	08	10001100100	61	]	]	61	11001000010
09	)	)	09	11001001000	62	^	^	62	11110001010
10	*	*	10	11001000100	63	_	_	63	10100110000
11	+	+	11	11000100100	64	NUL	`	64	10100001100
12	,	,	12	10110011100	65	SOH	a	65	10010110000
13	-	-	13	10011011100	66	STX	b	66	10010000110
14	.	.	14	10011001110	67	ETX	c	67	10000101100
15	/	/	15	10111001100	68	EOT	d	68	10000100110
16	0	0	16	10011101100	69	ENQ	e	69	10110010000
17	1	1	17	10011100110	70	ACK	f	70	10110000100
18	2	2	18	11001110010	71	BEL	g	71	10011010000
19	3	3	19	11001011100	72	BS	h	72	10011000010
20	4	4	20	11001001110	73	HT	i	73	10000110100
21	5	5	21	11011100100	74	LF	j	74	10000110010
22	6	6	22	11001110100	75	VT	k	75	11000010010
23	7	7	23	11101101110	76	FF	l	76	11001010000
24	8	8	24	11101001100	77	CR	m	77	11110111010
25	9	9	25	11100101100	78	SO	n	78	11000010100
26	:	:	26	11100100110	79	SI	o	79	10001111010

VALUE	WHICH REPRESENTS IN CHARACTER SET			ENCODING	VALUE	WHICH REPRESENTS IN CHARACTER SET			ENCODING
	A	B	C			A	B	C	
27	;	;	27	11101100100	80	DLE	p	80	10100111100
28	<	<	28	11100110100	81	DC1	q	81	10010111100
29	=	=	29	11100110010	82	DC2	r	82	10010011110
30	>	>	30	11011011000	83	DC3	s	83	10111100100
31	?	?	31	11011000110	84	DC4	t	84	10011110100
32	@	@	32	11000110110	85	NAK	u	85	10011110010
33	A	A	33	10100011000	86	SYN	v	86	11110100100
34	B	B	34	10001011000	87	ETB	w	87	11110010100
35	C	C	35	10001000110	88	CAN	x	88	11110010010
36	D	D	36	10110001000	89	EM	y	89	11011011110
37	E	E	37	10001101000	90	SUB	z	90	11011110110
38	F	F	38	10001100010	91	ESC	{	91	11110110110
39	G	G	39	11010001000	92	FS	|	92	10101111000
40	H	H	40	11000101000	93	GS	}	93	10100011110
41	I	I	41	11000100010	94	RS	~	94	10001011110
42	J	J	42	10110111000	95	US	DEL	95	10111101000
43	K	K	43	10110001110	96	FNC3	FNC3	96	10111100010
44	L	L	44	10001101110	97	FNC2	FNC2	97	11110101000
45	M	M	45	10111011000	98	SHIFT	SHIFT	98	11110100010
46	N	N	46	10111000110	99	Code C	Code C	99	10111011110
47	O	O	47	10001110110	100	Code B	FNC4	Code B	10111101110
48	P	P	48	11101110110	101	FNC4	Code A	Code A	11101011110
49	Q	Q	49	11010001110	102	FNC1	FNC1	FNC1	11110101110
50	R	R	50	11000101110	103	START A	START A	START A	11010000100
51	S	S	51	11011101000	104	START B	START B	START B	11010010000
52	T	T	52	11011100010	105	START C	START C	START C	11010011100
						STOP	STOP	STOP	11000111010

Note

There are three type special values:

- Green Color is Shift Code of Character Set
- Blue Color is Start character
- Red Color is Function Code

Bibliography

[1] CODE 128 SYMBOLOGY[1]

[2] Bar Code Symbologies[2]

[3] Bar Coding Basics[3]

[4] All About Code 128 Barcode[4]

[5] Wiki: Code 128[5]

[6] Code 128 Specification[6]

[7] GS1-128 Barcodes[7]

[8] 关于区分CODE128和EAN128码[8]

[1] http://www.barcodeisland.com/code128.phtml
[2] http://www.taltech.com/resources/intro_to_bc/bcsymbol.htm
[3] http://www.taltech.com/resources/intro_to_bc/bcbascs.htm
[4] http://www.adams1.com/128code.html
[5] http://en.wikipedia.org/wiki/Code_128
[6] http://www.barcodeman.com/info/c128.php3
[7] http://www.gs1-128.info/
[8] http://space.cnblogs.com/group/topic/3503/