



### 4.1 Structure

A GS1-128 Bar Code encodes a GTIN in a 14 digit data string and may also encode attribute data using Application Identifiers (AIs). When a GTIN-13 or GTIN-12 is encoded in a 14 digit data string, one or two filler zeros respectively must be added in front of the GTIN. With the use of AIs a GS1-128 Bar Code can encode alpha numeric attribute data and the GS1 Identification Keys for identifying Logistic Units, Locations, Assets, Documents, Services, Shipments and Consignments.

All GS1-128 Symbols follow as standard structure:

- Left Quiet Zone
- Start Character
- Function 1 Symbol Character (FNC1)
- Representation of the encoded characters with each set of information beginning with the appropriate AI
- Symbol Check Character (modulo 103)
- Stop Character
- Right Quiet Zone

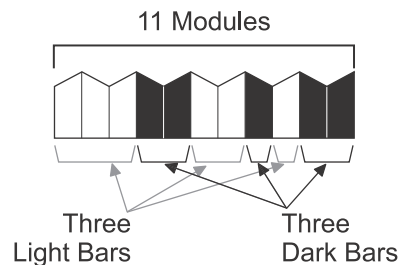




## 4.2 Encoding GS1-128 Symbols

Each symbol character except the stop character in a GS1-128 Bar Code comprises 11 modules grouped into three dark bars and three spaces. An example of a symbol character can be seen below in Figure 11. The stop pattern comprises 13 modules grouped into four dark bars and three spaces.

A list of all the combinations of dark and light modules for each human readable character is given in Table 12.



**Figure 11 - GS1-128 Symbol Character**

### 4.2.1 Character Sets

There are three ways to represent each character, called code sets A, B, and C.

#### Code Set A

Code set A includes all of the standard upper case alphanumeric characters and punctuation characters together with the symbology elements (e.g., characters with ASCII values from 00 to 95) and seven special characters.

#### Code Set B

Code set B includes all of the standard upper case alphanumeric characters and punctuation characters together with the lowercase alphabetic characters (e.g., ASCII characters 32 to 127 inclusive) and seven special characters.

#### Code Set C

Code set C is for numerical characters only and includes the set of 100 digit pairs from 00 to 99 inclusive, as well as three special characters. This allows numeric data to be encoded as two data digits per symbol character.



## 4.2.2 Special Characters

The last seven characters of code sets A and B (character values 96 to 102) and the last three characters of code set C (character values 100 to 102) are special non-data characters that, though they have particular significance to the bar code reader, have no ASCII character equivalents.

### Code Set and Shift Characters

Code set and shift characters shall be used to change from one code set to another within a symbol.

The decoder shall not transmit them.

- Code set characters: Code A, B, and C characters allow a change the symbol code set from the code set previously defined to the new code set, which is defined by the code character. This change applied to all characters following the code set character until either the end of the symbol, another code set character, or the shift character is encountered.
- The shift character has a similar function to the “shift” key on a keyboard. It allows a character set change from A to B or B to A for the **single character** that immediately follows the shift character. Subsequent characters revert to the character set defined prior to the shift character.

### Function Characters

Function Characters (FNC) provide special operations and application instructions to the bar code reading device.

- The Function 1 Symbol Character (FNC1) is always encoded after the Start Character in a GS1-128 Bar Code to differentiate GS1-128 Bar Codes from the more generalised Code-128 bar codes. In other words, any Code-128 bar code in the world that has an FNC1 immediately following the Start Character is always a GS1-128 Bar Code. FNC 1 is also used as a separator character when variable length AIs and their data fields are concatenated into a single bar code. This character is transmitted as character <GS>, ASCII value 29. (Please note that it is not permissible to encode the GS character in the bar code in place of the FNC1 as a separator character).
- The Function 2 Character (FNC2) (Message Append) is not used in the GS1 System.
- The Function 3 Character (FNC3) (Initialise) instructs the bar code reader to interpret the data from the symbol containing the FNC3 as instructions for initialisation or reprogramming of the bar code reader. The data from the symbol shall not be transmitted by the bar code reader. This character may occur anywhere in the symbol.
- The Function 4 Character (FNC4) is not used in the GS1 System.

## 4.2.3 Start and Stop Characters

- Start Characters A, B, and C define the corresponding code set to be used initially in the symbol.
- The Stop Character is common to all code sets.
- The decoder shall not transmit Start and Stop Characters.



## 4.2.4 Symbol Check Character

The Symbol Check Character (Modulo 103) is the last symbol character before the Stop Character. The Symbol Check Character must not be represented in the Human Readable Interpretation or transmitted by the decoder. Follow the steps below to calculate the Symbol Check Character. In less than 1% of cases, **FNC1** is the Symbol Check Character.

**Step 1:** Retrieve the symbol character value from Table 12 on page 22.

**Step 2:** Each symbol character position is given a weight. The Start Character is weighted 1. Then, beginning on the left with the first symbol character following the Start Character, the FNC1, the weights are 1, 2, 3, and 4 to...n for all subsequent symbol characters up to, but not including, the Symbol Check Character itself; n denotes the number of symbol characters representing data or special information in the symbol, exclusive of the Start and Stop Characters and Symbol Check Character.

**Note:** Both the Start Character and the FNC1 following the Start Character are weighted by 1

**Step 3:** Multiply each symbol character value by its weight.

**Step 4:** Add the totals of the calculations from step 3.

**Step 5:** Divide the result of step 4 by 103.

**Step 6:** The remainder derived from the calculation in step 5 is the symbol character value of the Symbol Check Character. Find the value of the remainder in Table 12 on page 22. You can then see the associated representation in the appropriate character set, or as bars and spaces.

Below is an example of a Symbol Check Character calculation for the Batch/Lot Number 2503X45 encoded using AI (10).

Characters	Start C	FNC1	10	25	03	Code B	X
Step 1: Character values	105	102	10	25	3	100	56
Step 2: Weights	1	1	2	3	4	5	6
Step 3: Multiply value by weight	105	102	20	75	12	500	336
Step 4: Sum of step 3	1150						
Step 5: Divide by 103	1150 / 103 = 11						
Step 6: Remainder = Symbol Check Character value	17						

**TABLE 11.** Symbol Check Character example calculation



## 4.2.5 Encoding GS1-128 Bar Codes Efficiently

Always try to minimise the GS1-128 Bar Code length. Make sure you follow the guidance provided in this section.

**Note:** The term **data** refers to the Application Identifier followed by a string of human readable characters.

A Start character precedes every group of data. Control characters are listed under character set A, positions 64 to 95, of Table 12 on page 22.

The conditions for using the different Start characters are:

- when the data begins with four or more digits, use Start C
- when the data begins with less than four digits, and a control character occurs in the data before any lower case character, use Start A
- otherwise, use Start B.

If you use Start C and the data begins with an odd number of digits, place a Code A or Code B before the last digit. Use the above guidelines to choose between Code A and Code B.

When four or more digits occur together in character sets A or B:

- if there is an even number of digits in the group, insert Code C before the first digit
- if there is an odd number of digits in the group, insert Code C immediately after the first digit.

When in character **set B** and a control character appears in the data:

- followed by a lower case character before another control character appears, insert the Shift character before the control character
- otherwise, insert Code A before the control character.

When in character **set A** and lower case character appears in the data:

- followed by a control character before another lower case character, insert the Shift character before the lower case character
- otherwise, insert Code B before the lower case character.

When in character **set C**, and a non-numeric character occurs in the data:

- insert Code A or B before the non-numeric character using the above guidelines to choose between Code A and B.





## 4.2.6 Data Character Encodation

The symbol character bar (dark bar) and space (light bar) patterns shown in Table 12 represent the data characters listed under the columns for code set A, B, or C. The choice of code set depends on the Start Character, the use of code A, code B, or code C characters, or the shift character. If the symbol begins with Start Character A, then code set A is defined initially. Code set B and code set C are similarly defined by beginning the symbol with Start Character B or C, respectively. The code set can be redefined within the symbol by using code A, code B, and code C characters or the shift character (see [“Special Characters” on page 19](#) for the use of special characters).

The same data may be represented by different Code 128 Symbols through the use of different combinations of Start Character, code set, and shift characters. The individual applications do not specify code sets A, B, or C. Section 5.4.8.10 contains rules to minimise the length of the symbol for any given data.

Each symbol character is assigned a numeric value listed in Table 12. This value is used in calculating the Symbol Check Character value. It may also be used to provide a conversion to and from ASCII values.

Value	Code set A	Code set B	Code Set C	Element Widths (Modules) B S B S B S
0	SP	SP	00	2 1 2 2 2 2
1	!	!	01	2 2 2 1 2 2
2	"	"	02	2 2 2 2 2 1
3	#	#	03	1 2 1 2 2 3
4	\$	\$	04	1 2 1 3 2 2
5	%	%	05	1 3 1 2 2 2
6	and	and	06	1 2 2 2 1 3
7	'	'	07	1 2 2 3 1 2
8	(	(	08	1 3 2 2 1 2
9	)	)	09	2 2 1 2 1 3
10	*	*	10	2 2 1 3 1 2
11	+	+	11	2 3 1 2 1 2
12	,	,	12	1 1 2 2 3 2
13	-	-	13	1 2 2 1 3 2
14	.	.	14	1 2 2 2 3 1
15	/	/	15	1 1 3 2 2 2
16	0	0	16	1 2 3 1 2 2
17	1	1	17	1 2 3 2 2 1
18	2	2	18	2 2 3 2 1 1
19	3	3	19	2 2 1 1 3 2
20	4	4	20	2 2 1 2 3 1

**TABLE 12.** Data character representation in GS1-128 Bar Codes





Value	Code set A	Code set B	Code Set C	Element Widths (Modules) B S B S B S
21	5	5	21	2 1 3 2 1 2
22	6	6	22	2 2 3 1 1 2
23	7	7	23	3 1 2 1 3 1
24	8	8	24	3 1 1 2 2 2
25	9	9	25	3 2 1 1 2 2
26	:	:	26	3 2 1 2 2 1
27	;	;	27	3 1 2 2 1 2
28	<	<	28	3 2 2 1 1 2
29	=	=	29	3 2 2 2 1 1
30	>	>	30	2 1 2 1 2 3
31	?	?	31	2 1 2 3 2 1
32	@	@	32	2 3 2 1 2 1
33	A	A	33	1 1 1 3 2 3
34	B	B	34	1 3 1 1 2 3
35	C	C	35	1 3 1 3 2 1
36	D	D	36	1 1 2 3 1 3
37	E	E	37	1 3 2 1 1 3
38	F	F	38	1 3 2 3 1 1
39	G	G	39	2 1 1 3 1 3
40	H	H	40	2 3 1 1 1 3
41	I	I	41	2 3 1 3 1 1
42	J	J	42	1 1 2 1 3 3
43	K	K	43	1 1 2 3 3 1
44	L	L	44	1 3 2 1 3 1
45	M	M	45	1 1 3 1 2 3
46	N	N	46	1 1 3 3 2 1
47	O	O	47	1 3 3 1 2 1
48	P	P	48	3 1 3 1 2 1
49	Q	Q	49	2 1 1 3 3 1
50	R	R	50	2 3 1 1 3 1
51	S	S	51	2 1 3 1 1 3
52	T	T	52	2 1 3 3 1 1
53	U	U	53	2 1 3 1 3 1
54	V	V	54	3 1 1 1 2 3
55	W	W	55	3 1 1 3 2 1
56	X	X	56	3 3 1 1 2 1

**TABLE 12.** Data character representation in GS1-128 Bar Codes




Value	Code set A	Code set B	Code Set C	Element Widths (Modules) B S B S B S
57	Y	Y	57	3 1 2 1 1 3
58	Z	Z	58	3 1 2 3 1 1
59	[	[	59	3 3 2 1 1 1
60	\	\	60	3 1 4 1 1 1
61	]	]	61	2 2 1 4 1 1
62	^	^	62	4 3 1 1 1 1
63	_	_	63	1 1 1 2 2 4
64	NUL	'	64	1 1 1 4 2 2
65	SOH	a	65	1 2 1 1 2 4
66	STX	b	66	1 2 1 4 2 1
67	ETX	c	67	1 4 1 1 2 2
68	EOT	d	68	1 4 1 2 2 1
69	ENQ	e	69	1 1 2 2 1 4
70	ACK	f	70	1 1 2 4 1 2
71	BEL	g	71	1 2 2 1 1 4
72	BS	h	72	1 2 2 4 1 1
73	HT	i	73	1 4 2 1 1 2
74	LF	j	74	1 4 2 2 1 1
75	VT	k	75	2 4 1 2 1 1
76	FF	l	76	2 2 1 1 1 4
77	CR	m	77	4 1 3 1 1 1
78	SO	n	78	2 4 1 1 1 2
79	SI	o	79	1 3 4 1 1 1
80	DLE	p	80	1 1 1 2 4 2
81	DC1	q	81	1 2 1 1 4 2
82	DC2	r	82	1 2 1 2 4 1
83	DC3	s	83	1 1 4 2 1 2
84	DC4	t	84	1 2 4 1 1 2
85	NAK	u	85	1 2 4 2 1 1
86	SYN	v	86	4 1 1 2 1 2
87	ETB	w	87	4 2 1 1 1 2
88	CAN	x	88	4 2 1 2 1 1
89	EM	y	89	2 1 2 1 4 1
90	SUB	z	90	2 1 4 1 2 1
91	ESC	{	91	4 1 2 1 2 1
92	FS		92	1 1 1 1 4 3

**TABLE 12.** Data character representation in GS1-128 Bar Codes




Value	Code set A	Code set B	Code Set C	Element Widths (Modules) B S B S B S
93	GS	}	93	1 1 1 3 4 1
94	RS	~	94	1 3 1 1 4 1
95	US	DEL	95	1 1 4 1 1 3
96	FNC 3	FNC 3	96	1 1 4 3 1 1
97	FNC 2	FNC 2	97	4 1 1 1 1 3
98	SHIFT	SHIFT	98	4 1 1 3 1 1
99	CODE C	CODE C	99	1 1 3 1 4 1
100	CODE B	FNC 4	CODE B	1 1 4 1 3 1
101	FNC 4	CODE A	CODE A	3 1 1 1 4 1
102	FNC 1	FNC 1	FNC 1	4 1 1 1 3 1
103	START A			2 1 1 4 1 2
104	START B			2 1 1 2 1 4
105	START C			2 1 1 2 3 2

**TABLE 12.** Data character representation in GS1-128 Bar Codes

	Bars (B) and spaces (S) B S B S B S B
STOP	2 3 3 1 1 1 2

**TABLE 13.** Stop character representation in a GS1-128 Bar Code



## 4.3 GS1-128 System Considerations and Processing Software

Wherever possible, AIs and fixed field lengths have been chosen to result in an even number of numeric characters. This reduces the length of the bar code, because GS1-128 symbols can pack a pair of digits into one symbol character.

Although GS1-128 Bar Codes can encode any length alphanumeric field, data fields take up less space when they are all numeric and have an even number of digits. For example, when assigning batch numbers, select even length numeric batch numbers to produce the shortest bar code.

**Note:** AIs with three digit identifiers, for example AI (400) - Purchase Order Number, require an odd number of data digits to make an even length field.

The symbology identifier prefix **JC1** identifies the symbology being read by the scanner. GS1-128 Bar Codes must be processed to break them into fields through their respective AIs.





## GS1-128 Field Length Indicators

Als identify field lengths as either:

- variable
- pre-defined

When several Als and their data fields are concatenated into one bar code, each variable length field must be followed by the FNC 1 character unless it is the last field encoded in the symbol.

Provisions have been made for pre-defined fields which will be allocated in the future. See Table 14. Always include this table in processing software. The table is permanent - it will not change. This means that decoding software can be programmed independently of any future additional release of Als.

Table 14 specifies the length of the data string including the two digit indicator. The picture of the data string is not pre-defined.

Element Strings with Pre-defined Length using Application Identifiers (Als)			
First two digits of the Application Identifier	Number of Characters (AI and Data Field)	First two digits of the Application Identifier	Number of Characters (AI and Data Field)
00	20	17	8
01	16	(18)	8
02	16	(19)	8
(03)	16	20	4
(04)	18	31	10
11	8	32	10
12	8	33	10
13	8	34	10
(14)	8	(35)	10
15	8	(36)	10
(16)	8	41	16

**TABLE 14.** Pre-Defined Length Application Identifiers

**Note:** This table is limited to the listed numbers and will remain unchanged. The numbers in parenthesis are not yet assigned.



## User Application AIs

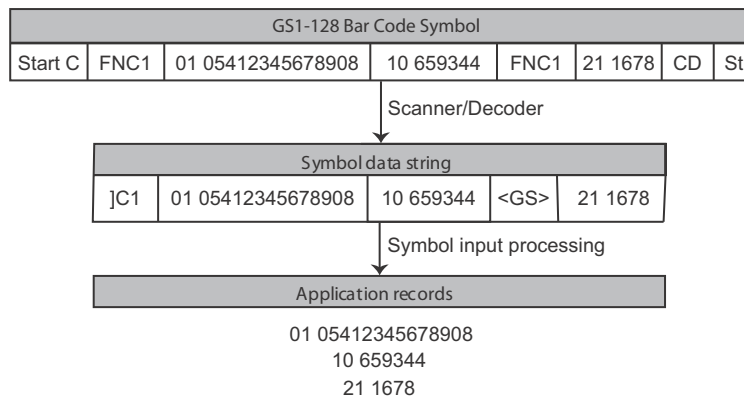
In addition to the fixed length field indicators described in “GS1-128 Field Length Indicators” on page 27 and shown in Table 14, you need to define a table of AIs which will be used by your applications.

## Scanner/Decoder

The scanner/decoder:

- reads the bar code, identifies Code-128 and decodes the bar code
- verifies the bar code’s integrity by checking the value of its Check Digit (by the modulo 103 algorithm)
- creates the symbology identification, that is, start character + FNC 1 = ]C1
- translates the FNC 1 separator character into <GS> (ASCII 29)
- formats the symbol data string
- transmits the symbol data string to the symbol input processing software of the computer.

The GS1-128 Bar Code Symbol in this example contains a GTIN, Batch Number, and Serial Number.



Note: Separators and spaces are shown for easy reading.

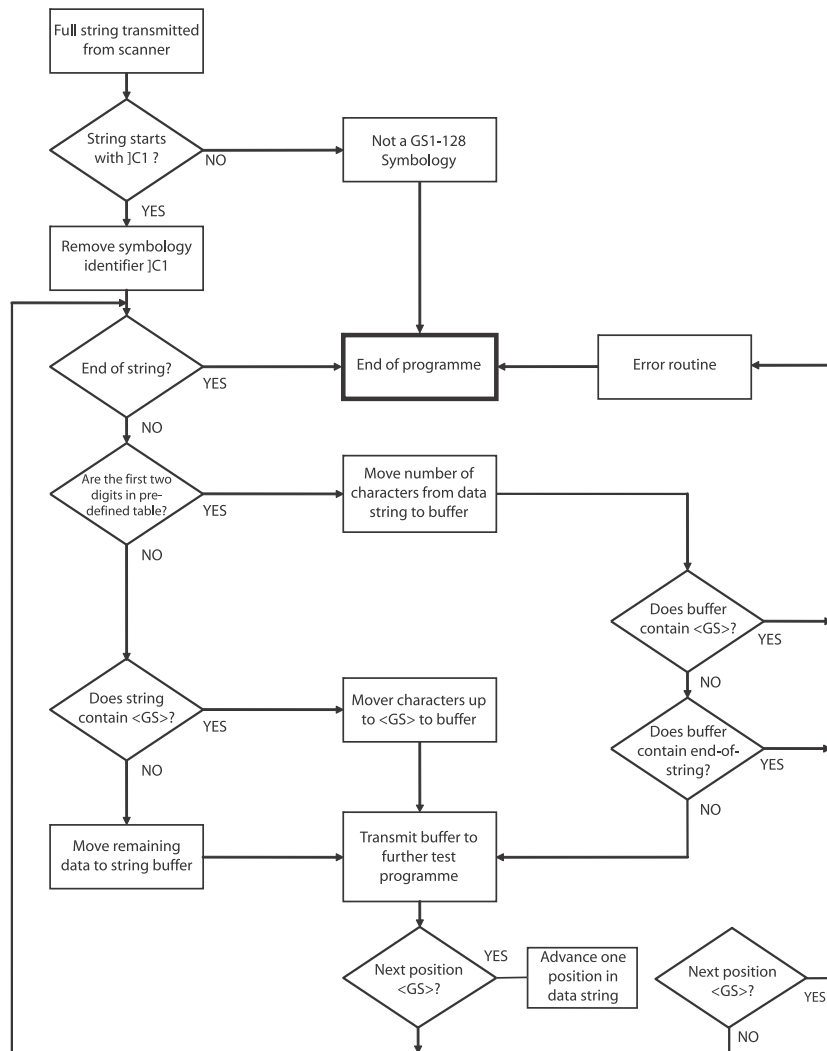
**Figure 12 - Bar Code Input Processing Chart**



## Symbol Input Processing

Symbol input processing software:

- verifies that symbology is GS1-128 by checking the symbology identifier ]C1
- separates the AIs in the symbol data string using pre-defined length indicator table and <GS> separator
- transmits each AI and its data field to the application processing software



**Figure 13 - Decoding Software Flowchart - Basic Required Logic**

It may not be the optimum software implementation. It assumes the scanner provides ]C1 to identify GS1-128, and <GS> (ASCII 29) for any other FNC 1 characters following the initial FNC 1 in the start character.



## Transmitted Data

All data characters are included in the data transmission.

The shift characters, code characters, symbol Check Digit, and stop pattern are not transmitted.

The double character start pattern (Start A, B, or C) plus FNC 1 are transmitted as ]C1. This special string of characters identifies the GS1-128 Bar Code from other symbologies.

The FNC1, used as a separator when multiple AIs and their data fields are concatenated, is transmitted as character <GS> (ASCII 29). It is not permissible to substitute a GS character set for the FNC1 within the bar code itself.

Never use the character string ]C1 and the character <GS> in the user data portion of any Application Identifier.

The AI is not part of the data field. When the data is used in other applications, such as EDI, the AI must be dropped.