

常用一维条形码

1 39 码（CODE39）

39 码可以包含数字及英文字母。除了超市、零售业的应用中使用 UPC/EAN 码外，几乎在其他应用环境中，都是使用 39 码。39 码是目前使用最广泛的条码规格，支持 39 码的软硬件设备也最齐全。

1.1 特征

- ◆ 能表示 44 个字符，A-Z、0-9、SPACE、-、.、\$、/、+、%、*
- ◆ 分散式，条码组之间使用细白条分隔
- ◆ 两种宽度
- ◆ 自我检查
- ◆ 有扩展模式 《Full ASCII Mode》
- ◆ 检查码字符可有可无，视需求而定

1.2 组成

- ◆ 各个字符有 9 条黑白相间，粗细不同的线条组成，其中 6 条为黑白细条 3 条黑白粗条
- ◆ 一串字符必须在头尾加上起始字符和结束字符 “*”

1.3 校验方法

CODE39									
字符	0	1	2	3	4	5	6	7	8
值	0	1	2	3	4	5	6	7	8
字符	9	A	B	C	D	E	F	G	H
值	9	10	11	12	13	14	15	16	17
字符	I	J	K	L	M	N	O	P	Q
值	18	19	20	21	22	23	24	25	26
字符	R	S	T	U	V	W	X	Y	Z
值	27	28	29	30	31	32	33	34	35
字符	-	.	SPACE	\$	/	+	%		
值	36	37	38	39	40	41	42		

找到输入字符串每个字符对应值，求和，除以 43，取余数。

1.4 条码说明



1.5 编码表

字母	黑條	白條	字母	黑條	白條
1	10001	0100	K	10001	0001
2	01001	0100	L	01001	0001
3	11000	0100	M	11000	0001
4	00101	0100	N	00101	0001
5	10100	0100	O	10100	0001
6	01100	0100	P	01100	0001
7	00011	0100	Q	00011	0001
8	10010	0100	R	10010	0001
9	01010	0100	S	01010	0001
0	00110	0100	T	00110	0001
A	10001	0010	U	10001	1000
B	01001	0010	V	01001	1000
C	11000	0010	W	11000	1000
D	00101	0010	X	00101	1000
E	10100	0010	Y	10100	1000
F	01100	0010	Z	01100	1000
G	00011	0010	-	00011	1000
H	10010	0010	•	10010	1000
I	01010	0010	SPACE	01010	1000
J	00110	0010	STR/STP	00110	1000
\$	00000	1110	+	00000	1011
/	00000	1101	%	00000	0111

1 = 寬黑條 / 白條

0 = 細黑條 / 白條

P.S.

在程序中可以使用“11”表示宽黑条，‘1’表示细黑条，“00”表示宽白条，“0”表示细白条。那么字符 1 就可以表示为 110100101011。使用此方法建立一个编码表，每个字符可以长度为 12 的“01”字符串来表示。

1.6 典型 CODE39 条码



1.7 CODE39 的扩展码

扩展码表同 CODE93。但是扩展方式不同，39 码使用\$,/+,%与其 26 个大写字母组合，表示 ASCII 码表中的其他字符。条空表示方式和校验方式 与标准 39 码相同。
93 码中使用的控制码与 26 个大写字母的组合。

2 93 码(CODE93)

2.1 组成

- ◆ 字母：A-Z，数字：0-9，符号：SPACE, -, ., \$, /, +, %, 控制码：\$, /, +, %,起始结束码：
□
- ◆ 每个字由 9 个模组成,包括 3 条粗细黑条及 3 条粗细白条。每一黑条或白条有可能为 1.2.3.4 模组成

2.2 特征

- ◆ 用 4 个控制码 \$, %, /, + 组合其他字母或符号，可编程 FULL ASCII 字母，读码器读到上面 4 个控制码的组合时候，送出的字尾所对应的 ASCII。
- ◆ 有 2 个检验码 C 和 K。

2.3 校验方法

- ◆ 先查出资料所对应值，对应值的表如下

字符	0	1	2	3	4	5	6	7	8
值	0	1	2	3	4	5	6	7	8

字符	9	A	B	C	D	E	F	G	H
值	9	10	11	12	13	14	15	16	17
字符	I	J	K	L	M	N	O	P	Q
值	18	19	20	21	22	23	24	25	26
字符	R	S	T	U	V	W	X	Y	Z
值	27	28	29	30	31	32	33	34	35
字符	-	.	SPACE	\$	/	+	%		
值	36	37	38	39	40	41	42		

- ◆ 检查码 C 由资料的最右边一位用 1-20 顺序排列，若资料超过 20 位，在从 1-20 起算，顺序号作为权值，分别乘以对应值，求和，除以 47，取余数
- ◆ 检查码 K 由 C 位用 1-15 顺序排列，若资料差偶偶 15 位，再从 1-15 起算，顺序号作为权值，分别乘以对应值，求和，除以 47，取余数
- ◆ 举例：
 资料 CODE SP 9 3 “C” “K”
 资料对应值 12 24 13 14 38 9 3
 C 排列顺序 7 6 5 4 3 2 1
 K 排列顺序 8 7 6 5 4 3 2 1
 $(1 \times 3) + (2 \times 9) + (3 \times 38) + (4 \times 14) + (5 \times 13) + (6 \times 24) + (7 \times 12) = 484$
 $C = 484 \div 47 = 10 \cdots 14 \text{ (余数)}$
 则 $C = 14 = E \text{ (对应值)}$
 $(1 \times 14) + (2 \times 3) + (3 \times 9) + (4 \times 38) + (5 \times 14) + (6 \times 13) + (7 \times 24) + (8 \times 12)$
 $= 611$
 $K = 611 \div 47 = 13 \cdots 0 \text{ (余数)}$
 则 $K = 0 = 0 \text{ (对应值)}$











2.4 条码说明



2.5 编码表

Character	Value(for check digit)	Pattern	Encodation
0	0	■ ■ ■	100010100
1	1	■ ■ ■	101001000
2	2	■ ■ ■	101000100
3	3	■ ■ ■	101000010
4	4	■ ■ ■	100101000
5	5	■ ■ ■	100100100
6	6	■ ■ ■	100100010
7	7	■ ■ ■	101010000
8	8	■ ■ ■	100010010
9	9	■ ■ ■	100001010

Character	Value(for check digit)	Pattern	Encodation
A	10	■ ■ ■	110101000
B	11	■ ■ ■	110100100
C	12	■ ■ ■	110100010
D	13	■ ■ ■	110010100
E	14	■ ■ ■	110010010
F	15	■ ■ ■	110001010
G	16	■ ■ ■	101101000
H	17	■ ■ ■	101100100
I	18	■ ■ ■	101100010
J	19	■ ■ ■	100110100
K	20	■ ■ ■	100011010
L	21	■ ■ ■	101011000
M	22	■ ■ ■	101001100
N	23	■ ■ ■	101000110
O	24	■ ■ ■	100101100
P	25	■ ■ ■	100010110
Q	26	■ ■ ■	110110100
R	27	■ ■ ■	110110010
S	28	■ ■ ■	110101100
T	29	■ ■ ■	110100110
U	30	■ ■ ■	110010110
V	31	■ ■ ■	110011010
W	32	■ ■ ■	101101100
X	33	■ ■ ■	101100110
Y	34	■ ■ ■	100110110
Z	35	■ ■ ■	100111010
-	36	■ ■ ■	100101110
•	37	■ ■ ■	111010100

Space	38		111010010
\$	39		111001010
/	40		101101110
+	41		101110110
\$	42		110101110
Ⓢ	43		100100110
Ⓟ	44		111011010
Ⓛ	45		111010110
Ⓡ	46		100110010
□			101011110

ASCII	CODE 93	ASCII	CODE 93	ASCII	CODE 93	ASCII	CODE 93
NUL	⓪ U	SP	Space	@	⓪ V	~	⓪ W
SOH	Ⓢ A	!	Ⓢ A	A	A	a	Ⓢ A
STX	Ⓢ B	"	Ⓢ B	B	B	b	Ⓢ B
ETX	Ⓢ C	#	Ⓢ C	C	C	c	Ⓢ C
EOT	Ⓢ D	\$	\$	D	D	d	Ⓢ D
ENQ	Ⓢ E	%	%	E	E	e	Ⓢ E
ACK	Ⓢ F	&	Ⓢ F	F	F	f	Ⓢ F
BEL	Ⓢ G	'	Ⓢ G	G	G	g	Ⓢ G
BS	Ⓢ H	(Ⓢ H	H	H	h	Ⓢ H
HT	Ⓢ I)	Ⓢ I	I	I	i	Ⓢ I
LF	Ⓢ J	•	Ⓢ J	J	J	j	Ⓢ J
VT	Ⓢ K	+	+	K	K	k	Ⓢ K
FF	Ⓢ L	'	Ⓢ L	L	L	l	Ⓢ L
CR	Ⓢ M	—	—	M	M	m	Ⓢ M
SO	Ⓢ N	•	•	N	N	n	Ⓢ N
SI	Ⓢ O	/	/	O	O	o	Ⓢ O
DLE	Ⓢ P	0	0	P	P	p	Ⓢ P
DC1	Ⓢ Q	1	1	Q	Q	q	Ⓢ Q
DC2	Ⓢ R	2	2	R	R	r	Ⓢ R
DC3	Ⓢ S	3	3	S	S	s	Ⓢ S
DC4	Ⓢ T	4	4	T	T	t	Ⓢ T
NAK	Ⓢ U	5	5	U	U	u	Ⓢ U
SYN	Ⓢ V	6	6	V	V	v	Ⓢ V
ETB	Ⓢ W	7	7	W	W	w	Ⓢ W
CAN	Ⓢ X	8	8	X	X	x	Ⓢ X
EM	Ⓢ Y	9	9	Y	Y	y	Ⓢ Y
SUB	⓪ Z	:	Ⓢ Z	Z	Z	z	Ⓢ Z
ESC	⓪ A	;	⓪ F	[⓪ K	{	⓪ P
FS	⓪ B	<	⓪ G	\	⓪ L	;	⓪ Q
GS	⓪ C	=	⓪ H]	⓪ M	}	⓪ R
RS	⓪ D	>	⓪ I	^	⓪ N	~	⓪ S
US	⓪ E	?	⓪ J	-	⓪ O	DEL	⓪ T

P.S.程序编码中，结束符号模块比起始符号多一个“1”；

使用控制符组合字母所表示的字符，编码时需要分解成控制符和大写字母两个模块。比如字符 NUL 可分解成控制符%和 U，条空为“111011010 和 110010110

2.6 典型 CODE93 条码



3 128 码(CODE128)

3.1 特征

- ◆ 能表示 106 种字元
- ◆ 3 个字符集
- ◆ 连续式
- ◆ 四种宽度
- ◆ 三种字符集可串联使用
- ◆ 使用检查码

3.2 组成

- ◆ 提供 128 个 ASCII 的所有字符
- ◆ 有 A、B、C 三种不同模组
- ◆ C 组数字专用模组，可从 00-99 编码，缩短编码长度
- ◆ A、B、C 三种模组依资料结构，可混合编码
- ◆ 每个字由 3 条粗细黑条及 3 条粗细白条组成，但每个模组都是 12 各个单元宽度
- ◆ 三组模组起始码不同，但结束码相同。

3.3 校验方法

- ◆ 首先为字符串标志条码（码字）
- ◆ 从起始符开始，自左向右为每个条码字符编号
- ◆ 为每一个字符分配权数，起始符和第一个字符的权数位 1，以后依次增加
- ◆ 每个字符的权数与码值相乘
- ◆ 所有乘积求和
- ◆ 第 5 步的结果除以 103 求余数
- ◆ 码值等于余数的字符即为校验符；

[illegible]

3.4 编码表

CODE A	CODE B	CODE C	VALUE		CODE A	CODE B	CODE C	VALUE	
Space	Space	00	0		=	=	29	29	
!	!	01	1		>	>	30	30	
`	`	02	2		?	?	31	31	
\$	\$	03	3		@	@	32	32	
#	#	04	4		A	A	33	33	
%	%	05	5		B	B	34	34	
&	&	06	6		C	C	35	35	
,	,	07	7		D	D	36	36	
((08	8		E	E	37	37	
))	09	9		F	F	38	38	
#	#	10	10		G	G	39	39	
+	+	11	11		H	H	40	40	
`	`	12	12		I	I	41	41	
-	-	13	13		J	J	42	42	
.	.	14	14		K	K	43	43	
/	/	15	15		L	L	44	44	
0	0	16	16		M	M	45	45	
1	1	17	17		N	N	46	46	
2	2	18	18		O	O	47	47	
3	3	19	19		P	P	48	48	
4	4	20	20		Q	Q	49	49	
5	5	21	21		R	R	50	50	
6	6	22	22		S	S	51	51	
7	7	23	23		T	T	52	52	
8	8	24	24		U	U	53	53	
9	9	25	25		V	V	54	54	
:	:	26	26		W	W	55	55	
:	:	27	27		X	X	56	56	
<	<	28	28		Y	Y	57	57	

CODE A	CODE B	CODE C	VALUE		CODE A	CODE B	CODE C	VALUE	
Z	Z	58	58	■■■■■ ■■■ ■■	DC3	S	83	83	■ ■■■■■ ■■
[[59	59	■■■■■ ■■■ ■■	DC4	T	84	84	■ ■■■■■ ■■
\	\	60	60	■■■■■ ■■■■■ ■■	NAK	U	85	85	■ ■■■■■ ■■
]]	61	61	■■■ ■■ ■■ ■■	SYN	V	86	86	■■■■■ ■■ ■■
		62	62	■■■■■ ■■ ■■	ETB	W	87	87	■■■■■ ■■ ■■
—	—	63	63	■ ■■ ■■■	CAN	X	88	88	■■■■■ ■■ ■■
NUL	.	64	64	■ ■■ ■■■	EM	Y	89	89	■■■ ■■ ■■■■■
SOH	*	65	65	■ ■■ ■■■	SUB	Z	90	90	■■■ ■■■■■ ■■
STX	b	66	66	■ ■■ ■■■	ESC	{	91	91	■■■■■ ■■ ■■
ETX	C	67	67	■ ■■ ■■■	FS		92	92	■ ■■ ■■■■■
EOT	D	68	68	■ ■■ ■■■	GS	}	93	93	■ ■■ ■■■■■
ENO	E	69	69	■ ■■ ■■	RS	~	94	94	■ ■■ ■■■■■
ACK	F	70	70	■ ■■ ■■	US	DET	95	95	■ ■■■■■ ■■
BEL	G	71	71	■ ■■ ■■	FNC3	FNC3	96	96	■ ■■■■■ ■■
BS	H	72	72	■ ■■ ■■	FNC2	FNC2	97	97	■■■■■ ■■ ■■
HT	I	73	73	■ ■■ ■■	Shift	Shift	98	98	■■■■■ ■■ ■■
LF	J	74	74	■ ■■ ■■	Code C	Code C	99	99	■ ■■■■■ ■■■■■
VT	K	75	75	■■■ ■■ ■■	Code B	Code 4	Code B	100	■ ■■■■■ ■■■■■
FF	L	76	76	■■■ ■■ ■■	FNC4	CODE A	Code	101	■■■■■ ■■ ■■■■■
CR	M	77	77	■■■■■ ■■■■■ ■■	FNC1	FNC1	FNC1	102	■■■■■ ■■ ■■■■■
SO	N	78	78	■■■ ■■ ■■	START	(Code A)		103	■■■ ■■ ■■
SI	O	79	79	■ ■■■■■ ■■	START	(Code B)		104	■■■ ■■ ■■
DLE	P	80	80	■ ■■ ■■■■■	START	(Code C)		105	■■■ ■■ ■■■■■
DC1	Q	81	81	■ ■■ ■■■■■	STOP	STOP	STOP		■■■ ■■■■■ ■■
DC2	R	82	82	■ ■■ ■■■■■					

P.S.程序中可用“1”表示一个单元宽度的黑条，“0”表示一个单元宽度的白条。则上图中的 Value=0 的条空可表示为 110110011000。

终止符有 13 个模块宽度。条空表示为 1100011101011，末尾多两个“11”；

3.5 典型 CODE128 码



3.6 条码说明

◆ 例如，某 128 码的值为 ROC12345，则其逻辑和条码如下

模块	数值	逻辑型态
B 式	Start	11010010000
B 式	R	11000101110
B 式	O	10001110110
B 式	C	10001000110
B 式	I	10011100110
B 式转 C 式		10111011110
C 式	23	11101101110
C 式	45	10111011000
C 式	Stop	1100011101011



◆ 某一个 128 码的值为 5418781，则其逻辑及条码如下

模块	数值	逻辑型态
C	Start	110100111100

C	54	11101011000
C	18	11001110010
C	78	11000010100
C	14	11000100010
C	Stop	1100011101011



4 EAN-128

EAN-128 码，现称 GS1-128 码，是专用于 GS1 系统中的条码，可以标注商品的附加信息，在山品信息的标志、茶品的跟踪与追溯中有广泛的用途。EAN-128 码来自于 CODE-128 码，在字符集、条空规则上与 CODE-128 完全一致。

4.1 组成

- ◆ EAN-128 码组成同 CODE-128
- ◆ 应用标识符使用分隔符码字 FNC1 表示

4.2 特征

同 CODE-128;

4.3 条码说明

- ◆ 如果字符串起始的字符是超过 4 个的数字字符时，应该采用 START C。数据中间如果有 4 个或 4 个以上连续的数字也要变换为字符集 C。
- ◆ 如果数据中的应用标识符使用的是预定义程度的 AI，则可以省去数据后的分隔符。

◆ 例子

例 1 (01) 16903128100250 (13) 091020 (15) 100420 用条码可以表示为

|START C|FNC1|01|16|90|31|28|10|02|50|13|09|10|20|15|10|04|20|校验码|STOP|

例 2 (02) 16903128100250 (37) 100 (10) 091000S

|START C|FNC1|02|16|90|31|28|10|02|50|37|10|CODE B| 0 |CODE
C|FNC1|10|09|10|00|CODE B|S|校验符|STOP|

5 标准 2 of 5 (INDUSTRY 2 OF 5 STANDARD)

5.1 组成:

- ◆ 由 5 条黑条组成，其中有 3 条细黑条，2 条宽黑条
- ◆ 黑条与黑条之间及字与字之间偶用一白条分隔，所以白条不表示资料
- ◆ 起始及结束码不同
- ◆ 数字 0~9

5.2 特征

- ◆ 一串资料可以改变程度，但只有数字可编
- ◆ 粗细比例 1: 3,允许误差±25%
- ◆ 分散式
- ◆ 无验证码

5.3 编码表

字母	S1	S2	S3	S4	S5
1	1	0	0	0	1
2	0	1	0	0	1
3	1	1	0	0	0
4	0	0	1	0	1
5	1	0	1	0	0
6	0	1	1	0	0
7	0	0	0	1	1
8	1	0	0	1	0
9	0	1	0	1	0
0	0	0	1	1	0
START	1	1	0		
STOP	1	0	1		

S1 - S5 = 黑條 1-5

1=寬條 , 0=細條

P.S. 程序中：“1”表示 1 个黑条宽度单元，因为粗细比是 1:3，显然粗黑条表示为“111”，又因为每两个黑条使用细白条分隔，所以上表中数字 1 可以表示为：“111010101011”

6 交错式 25 码(INTERLEAVED25)

6.1 组成

- ◆ 由 5 条粗细黑条组成，其中 3 条细黑条，2 条宽黑条
- ◆ 由 5 条粗细白条组成，其中 3 条细白条，2 条宽白条
- ◆ 黑条置于奇数位，白条置于偶数位，故又称交错式 2OF5
- ◆ 起始码和结束码不同
- ◆ 只能表示 0~9
- ◆ 必须偶数位配对

6.2 特性

- ◆ 资料程度一定是偶数
- ◆ 只能表示数字 0~9
- ◆ 连续式
- ◆ 不一定使用检验码

- ◆ 粗细比 1:2 或 1:3，误差 10%
- ◆ 粗细印刷比例若误差太大，可能解错码

6.3 校验方法

交叉 25 码的校验位计算方法依然是 Mod 10：

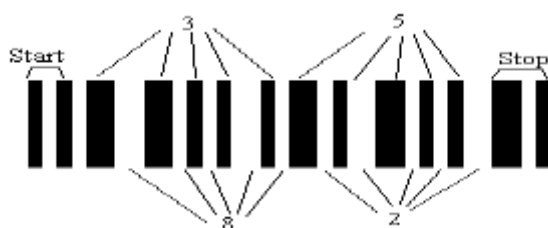
1. 字符个数为偶数时为:10 的倍数-[(奇数位的数字之和<从左至右)+(偶数位数字之和)*3 个位数]
2. 字符个数为奇数时为：10 的倍数-[(偶数位的数字之和<从左至右)+（奇数位数字之和）*3 个位数]

如：

514362 的校验位为 $10 * X - [(5+4+6) + 3 * (1+3+2)] = 7$ （因加校验位后个数为奇数，故前面加 0 后为 05143627。

76534 的校验位为 $10 * X - [(6+3) + 3 * (7+5+4)] = 3$

6.4 条码说明



6.5 编码表

字母	S1	S2	S3	S4	S5
1	1	0	0	0	1
2	0	1	0	0	1
3	1	1	0	0	0
4	0	0	1	0	1
5	1	0	1	0	0
6	0	1	1	0	0
7	0	0	0	1	1
8	1	0	0	1	0
9	0	1	0	1	0
0	0	0	1	1	0
START	0(BAR)	0	(BAR)0		
STOP	1	0	(BAR)0		

S1 — S5 = BARS/SPACES 1–5
1 = WIDE BAR/ SPACE
0 = NARROW BAR/SPACE

P.S.程序中，可以’N’表示窄条，’W’表示宽条，因为黑条白条的相对位置固定，所以上表其实是一个 pattern 表。比如字母 1 表示为 W N N N W。虽然粗细比允许 1:3，考虑到解错码的可能性，建议粗条使用 2 个单位宽度。

因为交错式 25 码必须是成对的数字，所以可以把相邻两个数字作为一个模块。比如条码 1234，可以分组为 12 | 34；取“12”为例，1 在奇数位置，为黑条，对应 pattern 为 WNNNW，2 在偶数位置，为白条，对应 pattern 为 NWNW，那么 12 合起来作为一个模块，混合 pattern 为 WNNWNNNNWW,这个 pattern 里，奇数位的 W 表示粗黑条，奇数位的 N 表示细黑条，偶数位的 W 表示粗白条，偶数位的 N 表示细白条，因此 12 的条空表示为：11010010101100

6.6 典型交错式 25 码



7 CODABAR 码

7.1 组成

- ◆ 由 7 条黑白相间、粗细不同的黑白条组成，其中包括 4 条黑条，3 条白条
- ◆ 字与字之间为一细白条
- ◆ 4 个起始和结束码，可互相配对，但一般都用相同。
- ◆ 数字：0~9， 特别符号：-、\$、=、/、.、+， 起始/结束码：A、B、C、D

7.2 特征

- ◆ 四种不同的起始/结束码
- ◆ 分散式
- ◆ 18 种宽度
- ◆ 不使用检查码
- ◆ 起始/结束码可以用来传递信息

7.3 校验方法

对应值之和,MOD16

步骤一.假设条形码数据码为 A37859B

步骤二.将各别字元相对值加总 $16+3+7+8+5+9+17=65$

步骤三.用为基数来除以相对值加总 $65/16=4....1$

步骤四.再以减掉余数,以求出相对值 $16-1=15$

7.4 编码表

字母	S1	L1	S2	L2	S3	L3	S4
1	0	0	0	0	1	1	0
2	0	0	0	1	0	0	1
3	1	1	0	0	0	0	0
4	0	0	1	0	0	1	0
5	1	0	0	0	0	1	0
6	0	1	0	0	0	0	1
7	0	1	0	0	1	0	0
8	0	1	1	0	0	0	0
9	1	0	0	1	0	0	0
0	0	0	0	0	0	1	1
—	0	0	0	1	1	0	0
\$	0	0	1	1	0	0	0
=	1	0	0	0	1	0	0
/	1	0	1	0	0	0	1
•	1	0	1	0	1	0	0
+	0	0	1	0	1	0	1
a	0	0	1	1	0	1	0
b	0	1	0	1	0	0	1
c	0	0	0	1	0	1	1
d	0	0	0	1	1	1	0
t	0	0	1	1	0	1	0
n	0	1	0	1	0	0	1
★	0	0	0	1	0	1	1
e	0	0	1	1	1	1	0

S1-S4:黑条 L1-L3:白条

0:窄, 1:宽

7.5 典型 CODABAR 码

UNIVERSITY LIBRARY



3 9349 00376 0473

8 UPC-A(和 EAN-8)

UPC-A 与 EAN-8 的编码方式相同，资料长度不同

8.1 特性

- ◆ 资料长度固定-12 位
- ◆ 只能用来表示数字 0~9
- ◆ 固定使用一位 检查码
- ◆ 连续式
- ◆ 使用四种宽度

8.2 组成

- ◆ 每一个字由 7 个 ELEMENTS 组成
- ◆ 有 4 中粗细黑白条混合编码
- ◆ 左护线 101， 中心弧线吗 01010， 右护线码 101

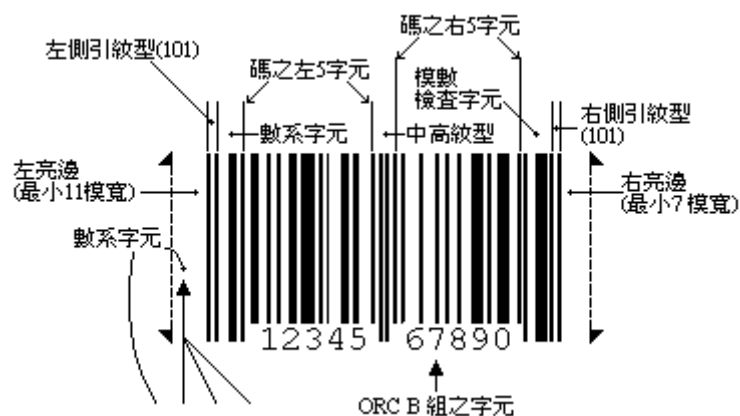
8.3 校验方法

X. $3 + Y = Z$, $10 - Z =$ 检查号码

X 为奇数位之和，Y 为偶数位之总和

Z 为奇数 + 偶数之总和的个位数

8.4 条码说明



8.5 编码表

DECIMAL	LEFT (A)	互補	RIGHT (C)
	(ODD PARITY -0)		(EVEN PARITY)
0	0001101		1110010
1	0011001		1100110
2	0010011		1101100
3	0111101		1000010
4	0100011		1011100
5	0110001		1001110
6	0101111		1010000
7	0111011		1000100
8	0110111		1001000
9	0001011		1110100

8.6 典型 UPC-A 和 EAN-8 码



9 EAN-13

9.1 特性

- ◆ 资料长度固定， 13 为
- ◆ 只能用来表示数字资源
- ◆ 固定使用一位 检查码
- ◆ 连续式
- ◆ 使用四种宽度

- ◆ 左护线码 101， 中心护线码 01010， 右护线码 101

9.2 校验方法

$X + Y \cdot 3 = Z$ 10 - Z = 检查號碼

X 为奇数位之和，Y 为偶数位之总和

Z 为奇数 + 偶数之总和的个位数

9.3 编码表

- ◆ EAN-13 的右半部编码如下表

DECIMAL	RIGHT (C)
0	1110010
1	1100110
2	1101100
3	1000010
4	1011100
5	1001110
6	1010000
7	1000100
8	1001000
9	1110100

- ◆ EAN-13 的左半部编码步骤如下：

1. 先查出 EAN13 最前面的国码数字
2. 根据国码数字查下表，O 代表 ODD PARITY, E 代表 EVEN PARITY

PARITY PATTERN	
1	00E0EE
2	00EE0E
3	00EEE0
4	0E00EE
5	0EE00E
6	0EEE00
7	0E0E0E
8	0E0EE0
9	0EE0E0

3. 上表的PARITY PATTERN 也代表EAN左半部6为数中每位 PARITY PATTERN
4. 依照 PARITY PATTERN 对照下表壳得出左边的编码

CHARACTER VALUE	ODD PARITY (O)	EVEN PARITY (E)
0	0001101	0100111
1	0011001	0110011
2	0010011	0011011
3	0111101	0100001
4	0100011	0011101
5	0110001	0111001
6	0101111	0000101
7	0111011	0010001
8	0110111	0001001
9	0001011	0010111

10 UPC-E

10.1 特性

- ◆ 资料长度固定，6 位
- ◆ 只能用来表示数字
- ◆ 使用一位检查码
- ◆ 连续式
- ◆ 使用四种宽度
- ◆ 每个 UPC-E 码都有一个相对应的 UPC-A 码存在
- ◆ 左护线码为 101， 有护线码为 010101

10.2 编码步骤

- ◆
- 1 依下表把断码先变成原来的长码（UPC-A）

長 碼 (UPC-A)												短 碼 (UPC-E)							
國 碼		廠 商 號 碼					商 品 號 碼					C							C
F1	F2																		
0	0	M1	M2	M3	M4	M5	A1	A2	A3	A4	A5	C	Z1	Z2	Z3	Z4	Z5	Z6	C
0	0	M1	M2	0	0	0	0	0	A3	A4	A5	C	M1	M2	A3	A4	A5	0	C
0	0	M1	M2	1	0	0	0	0	0	A4	A5	C	M1	M2	A3	A4	A5	1	C
0	0	M1	M2	2	0	0	0	0	0	A4	A5	C	M1	M2	A3	A4	A5	2	C
0	0	M1	M2	M3	0	0	0	0	0	A4	A5	C	M1	M2	M3	A4	A5	3	C
0	0	M1	M2	M3	M4	0	0	0	0	0	A5	C	M1	M2	M3	M4	A5	4	C
0	0	M1	M2	M3	M4	M5	0	0	0	0	A5(5-9)	C	M1	M2	M3	M4	M5	A5(5-9)	C

- 2 由 UPC-A 算出检查码之值
- 3 依检查码之值对照下表取得 PATTERN

(檢 核 碼)	UPC E 資料之數字組					
0	B	B	B	A	A	A
1	B	B	A	B	A	A
2	B	B	A	A	B	A
3	B	B	A	A	A	B
4	B	A	B	B	A	A
5	B	A	A	B	B	A
6	B	A	A	A	B	B
7	B	A	B	A	B	A
8	B	A	B	A	A	B
9	B	A	A	B	A	B

- 4 依据上表的数字再对照下表即可得 UPC-E 的编码

Character Value	(A)	(B)
0	0001101	0100111
1	0011001	0110011
2	0010011	0011011
3	0111101	0100001
4	0100011	0011101
5	0110001	0111001
6	0101111	0000101
7	0111011	0010001
8	0110111	0001001
9	0001011	0010111