常见字符集和编码

Yan Min
2009.02.28

目录

1 字符集和编码方式

常见的字符集有ASCII、IA5（ITU-T T.50）、IRA5、ISO646、ISO8859系列、Code Page （CPXXX，如CP437、CP850等）、Unicode、UTF8、GB等等。

1.1 ASCII 和 IA5

ASCII、IA5（IRA5 和 ISO646），虽然是 8bit 编码，但只用 7bits，从 0x00 到 0x7F，两者之间有些细微差别，但大多常用字符一样，一般情况下我们可以认为是同一个字符集，不同字符列举如下：

HEX	ASCII	IA5
0x01	Start of Header(SOH)	TC1
0x02	Start of Text(STX)	TC2
0x03	End of Text(ETX)	TC3
0x04	End of Transmission(EOT)	TC4
0x05	Enquire(ENQ)	TC5
0x06	Acknowledge(ACK)	TC6
0x10	Data Link Escape(DLE)	TC7
0x15	Negative Acknowledge(NAK)	TC8
0x16	Synchronous Idle(SYN)	TC9
0x17	End of Transmission block(ETB)	TC10
0x1C	File Separator(FS)	IS1
0x1D	Group Separator(GS)	IS2
0x1E	Record Separator(RS)	IS3
0x1F	Unit Separator(US)	IS4
0x24	Dollar Sign($)	Currency
0x7E	Equivalence (~)	over-line

ASCII 码表见下图：

Ctrl	Dec	Hex	Char	Code	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
^@	0	00		NUL	32	20		64	40	@	96	60	`
^A	1	01		SOH	33	21	!	65	41	A	97	61	a
^B	2	02		STX	34	22	"	66	42	B	98	62	b
^C	3	03		ETX	35	23	#	67	43	C	99	63	c
^D	4	04		EOT	36	24	$	68	44	D	100	64	d
^E	5	05		ENQ	37	25	%	69	45	E	101	65	e
^F	6	06		ACK	38	26	&	70	46	F	102	66	f
^G	7	07		BEL	39	27	'	71	47	G	103	67	g
^H	8	08		BS	40	28	(	72	48	H	104	68	h
^I	9	09		HT	41	29	)	73	49	I	105	69	i
^J	10	0A		LF	42	2A	*	74	4A	J	106	6A	j
^K	11	0B		VT	43	2B	+	75	4B	K	107	6B	k
^L	12	0C		FF	44	2C	,	76	4C	L	108	6C	l
^M	13	0D		CR	45	2D	-	77	4D	M	109	6D	m
^N	14	0E		SO	46	2E	.	78	4E	N	110	6E	n
^O	15	0F		SI	47	2F	/	79	4F	O	111	6F	o
^P	16	10		DLE	48	30	0	80	50	P	112	70	p
^Q	17	11		DC1	49	31	1	81	51	Q	113	71	q
^R	18	12		DC2	50	32	2	82	52	R	114	72	r
^S	19	13		DC3	51	33	3	83	53	S	115	73	s
^T	20	14		DC4	52	34	4	84	54	T	116	74	t
^U	21	15		NAK	53	35	5	85	55	U	117	75	u
^V	22	16		SYN	54	36	6	86	56	V	118	76	v
^W	23	17		ETB	55	37	7	87	57	W	119	77	w
^X	24	18		CAN	56	38	8	88	58	X	120	78	x
^Y	25	19		EM	57	39	9	89	59	Y	121	79	y
^Z	26	1A		SUB	58	3A	:	90	5A	Z	122	7A	z
^[	27	1B		ESC	59	3B	;	91	5B	[	123	7B	{
^\	28	1C		FS	60	3C	<	92	5C	\	124	7C	|
^]	29	1D		GS	61	3D	=	93	5D	]	125	7D	}
^^	30	1E	▲	RS	62	3E	>	94	5E	^	126	7E	~
^-	31	1F	▼	US	63	3F	?	95	5F	_	127	7F	⌂*

* ASCII code 127 has the code DEL. Under MS-DOS, this code has the same effect as ASCII 8 (BS). The DEL code can be generated by the CTRL + BKSP key.

1.2 ISO 8859 字符集

ISO 8859 是 8bit 单字节字符集，只使用 0x00~0xFF 编码，它们的 0x00~0x7F 与 ASCII 码基本一致，各字符集的不同处在于如何利用 0x80~0xFF 的码位：

ISO 8859-1	Latin alphabet No.1	West European
ISO 8859-2	Latin alphabet No.2	Central and East European
ISO 8859-3	Latin alphabet No.3	South European, Maltese & Esperanto
ISO 8859-4	Latin alphabet No.4	North European
ISO 8859-5	Latin/Cyrillic alphabet	Slavic languages
ISO 8859-6	Latin/Arabic alphabet	Arabic
ISO 8859-7	Latin/Greek alphabet	modern Greek
ISO 8859-8	Latin/Hebrew alphabet	Hebrew and Yiddish
ISO 8859-9	Latin alphabet No.5	Turkish
ISO 8859-10	Latin alphabet No.6	Nordic (Sámi, Inuit, Icelandic)
ISO 8859-11	Latin/Thai alphabet	Thai
ISO 8859-12	Latin/Devanagari	Devanagari
ISO 8859-13	Latin alphabet No.7	Baltic Rim
ISO 8859-14	Latin alphabet No.8	Celtic
ISO 8859-15	Latin alphabet No.9	adds euro to -1 (8 changes)
ISO 8859-16	Latin alphabet No.10	South-Eastern Europe

ISO 8859-12 是预留给梵文字母的，印地文和尼泊尔文都使用这种字母表。由于印度制定了自己的编码 ISCII(Indian Script Code for Information Interchange)，所以这个编号 12 的就未被使用。

常用的是 ISO 8859-1 和 ISO 8859-15 这两个，这两者的区别在于：ISO 8859-15 去掉了一些不常用的符号，加入了诸如欧元（€）和 Š, š, Ž, ž, Œ, œ, and Ÿ，这样就涵盖了法语（French），芬兰语（Finnish）和爱沙尼亚语（Estonian）：

Position	0xA4	0xA6	0xA8	0xB4	0xB8	0xBC	0xBD	0xBE
8859-1	¤	¦	¨	´	¸	¼	½	¾
8859-15	€	Š	š	Ž	ž	Œ	œ	Ÿ

ISO 8859-1 的 0x00~0x1F,0x7F~0x9F 的字符段没有定义,因此 ISO 8859-15 更具有通用性,8859-1 和 8859-15 码值见下图：

常见字符集和编码

ISO/IEC 8859-1 (Latin-1)																
	—0	**—1**	**—2**	**—3**	**—4**	**—5**	**—6**	**—7**	**—8**	**—9**	**—A**	**—B**	**—C**	**—D**	**—E**	**—F**
0—																
1—																
2—	SP 0020 ***32***	! 0021 ***33***	" 0022 ***34***	# 0023 ***35***	$ 0024 ***36***	% 0025 ***37***	& 0026 ***38***	' 0027 ***39***	(0028 ***40***	) 0029 ***41***	* 002A ***42***	+ 002B ***43***	, 002C ***44***	- 002D ***45***	. 002E ***46***	/ 002F ***47***
3—	0 0030 ***48***	1 0031 ***49***	2 0032 ***50***	3 0033 ***51***	4 0034 ***52***	5 0035 ***53***	6 0036 ***54***	7 0037 ***55***	8 0038 ***56***	9 0039 ***57***	: 003A ***58***	; 003B ***59***	< 003C ***60***	= 003D ***61***	> 003E ***62***	? 003F ***63***
4—	@ 0040 ***64***	A 0041 ***65***	B 0042 ***66***	C 0043 ***67***	D 0044 ***68***	E 0045 ***69***	F 0046 ***70***	G 0047 ***71***	H 0048 ***72***	I 0049 ***73***	J 004A ***74***	K 004B ***75***	L 004C ***76***	M 004D ***77***	N 004E ***78***	O 004F ***79***
5—	P 0050 ***80***	Q 0051 ***81***	R 0052 ***82***	S 0053 ***83***	T 0054 ***84***	U 0055 ***85***	V 0056 ***86***	W 0057 ***87***	X 0058 ***88***	Y 0059 ***89***	Z 005A ***90***	[005B ***91***	\ 005C ***92***	] 005D ***93***	^ 005E ***94***	_ 005F ***95***
6—	` 0060 ***96***	a 0061 ***97***	b 0062 ***98***	c 0063 ***99***	d 0064 ***100***	e 0065 ***101***	f 0066 ***102***	g 0067 ***103***	h 0068 ***104***	i 0069 ***105***	j 006A ***106***	k 006B ***107***	l 006C ***108***	m 006D ***109***	n 006E ***110***	o 006F ***111***
7—	p 0070 ***112***	q 0071 ***113***	r 0072 ***114***	s 0073 ***115***	t 0074 ***116***	u 0075 ***117***	v 0076 ***118***	w 0077 ***119***	x 0078 ***120***	y 0079 ***121***	z 007A ***122***	{ 007B ***123***	| 007C ***124***	} 007D ***125***	~ 007E ***126***	
8—																
9—																
A—	NBSP 00A0 ***160***	¡ 00A1 ***161***	¢ 00A2 ***162***	£ 00A3 ***163***	¤ 00A4 ***164***	¥ 00A5 ***165***	¦ 00A6 ***166***	§ 00A7 ***167***	¨ 00A8 ***168***	© 00A9 ***169***	ª 00AA ***170***	« 00AB ***171***	¬ 00AC ***172***	SHY 00AD ***173***	® 00AE ***174***	¯ 00AF ***175***
B—	° 00B0 ***176***	± 00B1 ***177***	² 00B2 ***178***	³ 00B3 ***179***	´ 00B4 ***180***	µ 00B5 ***181***	¶ 00B6 ***182***	· 00B7 ***183***	¸ 00B8 ***184***	¹ 00B9 ***185***	º 00BA ***186***	» 00BB ***187***	¼ 00BC ***188***	½ 00BD ***189***	¾ 00BE ***190***	¿ 00BF ***191***
C—	À 00C0 ***192***	Á 00C1 ***193***	Â 00C2 ***194***	Ã 00C3 ***195***	Ä 00C4 ***196***	Å 00C5 ***197***	Æ 00C6 ***198***	Ç 00C7 ***199***	È 00C8 ***200***	É 00C9 ***201***	Ê 00CA ***202***	Ë 00CB ***203***	Ì 00CC ***204***	Í 00CD ***205***	Î 00CE ***206***	Ï 00CF ***207***
D—	Ð 00D0 ***208***	Ñ 00D1 ***209***	Ò 00D2 ***210***	Ó 00D3 ***211***	Ô 00D4 ***212***	Õ 00D5 ***213***	Ö 00D6 ***214***	× 00D7 ***215***	Ø 00D8 ***216***	Ù 00D9 ***217***	Ú 00DA ***218***	Û 00DB ***219***	Ü 00DC ***220***	Ý 00DD ***221***	Þ 00DE ***222***	ß 00DF ***223***
E—	à 00E0 ***224***	á 00E1 ***225***	â 00E2 ***226***	ã 00E3 ***227***	ä 00E4 ***228***	å 00E5 ***229***	æ 00E6 ***230***	ç 00E7 ***231***	è 00E8 ***232***	é 00E9 ***233***	ê 00EA ***234***	ë 00EB ***235***	ì 00EC ***236***	í 00ED ***237***	î 00EE ***238***	ï 00EF ***239***
F—	ð 00F0 ***240***	ñ 00F1 ***241***	ò 00F2 ***242***	ó 00F3 ***243***	ô 00F4 ***244***	õ 00F5 ***245***	ö 00F6 ***246***	÷ 00F7 ***247***	ø 00F8 ***248***	ù 00F9 ***249***	ú 00FA ***250***	û 00FB ***251***	ü 00FC ***252***	ý 00FD ***253***	þ 00FE ***254***	ÿ 00FF ***255***
	—0	**—1**	**—2**	**—3**	**—4**	**—5**	**—6**	**—7**	**—8**	**—9**	**—A**	**—B**	**—C**	**—D**	**—E**	**—F**

常见字符集和编码

ISO-8859-15																
	–0	–1	–2	–3	–4	–5	–6	–7	–8	–9	–A	–B	–C	–D	–E	–F
0–	NUL 0000 0	SOH 0001 1	STX 0002 2	ETX 0003 3	EOT 0004 4	ENQ 0005 5	ACK 0006 6	BEL 0007 7	BS 0008 8	HT 0009 9	LF 000A 10	VT 000B 11	FF 000C 12	CR 000D 13	SO 000E 14	SI 000F 15
1–	DLE 0010 16	DC1 0011 17	DC2 0012 18	DC3 0013 19	DC4 0014 20	NAK 0015 21	SYN 0016 22	ETB 0017 23	CAN 0018 24	EM 0019 25	SUB 001A 26	ESC 001B 27	FS 001C 28	GS 001D 29	RS 001E 30	US 001F 31
2–	SP 0020 32	! 0021 33	" 0022 34	# 0023 35	$ 0024 36	% 0025 37	& 0026 38	' 0027 39	(0028 40	) 0029 41	* 002A 42	+ 002B 43	, 002C 44	- 002D 45	. 002E 46	/ 002F 47
3–	0 0030 48	1 0031 49	2 0032 50	3 0033 51	4 0034 52	5 0035 53	6 0036 54	7 0037 55	8 0038 56	9 0039 57	: 003A 58	; 003B 59	< 003C 60	= 003D 61	> 003E 62	? 003F 63
4–	@ 0040 64	A 0041 65	B 0042 66	C 0043 67	D 0044 68	E 0045 69	F 0046 70	G 0047 71	H 0048 72	I 0049 73	J 004A 74	K 004B 75	L 004C 76	M 004D 77	N 004E 78	O 004F 79
5–	P 0050 80	Q 0051 81	R 0052 82	S 0053 83	T 0054 84	U 0055 85	V 0056 86	W 0057 87	X 0058 88	Y 0059 89	Z 005A 90	[005B 91	\ 005C 92	] 005D 93	^ 005E 94	_ 005F 95
6–	` 0060 96	a 0061 97	b 0062 98	c 0063 99	d 0064 100	e 0065 101	f 0066 102	g 0067 103	h 0068 104	i 0069 105	j 006A 106	k 006B 107	l 006C 108	m 006D 109	n 006E 110	o 006F 111
7–	p 0070 112	q 0071 113	r 0072 114	s 0073 115	t 0074 116	u 0075 117	v 0076 118	w 0077 119	x 0078 120	y 0079 121	z 007A 122	{ 007B 123	| 007C 124	} 007D 125	~ 007E 126	DEL 007F 127
8–	PAD 0080 128	HOP 0081 129	BPH 0082 130	NBH 0083 131	IND 0084 132	NEL 0085 133	SSA 0086 134	ESA 0087 135	HTS 0088 136	HTJ 0089 137	VTS 008A 138	PLD 008B 139	PLU 008C 140	RI 008D 141	SS2 008E 142	SS3 008F 143
9–	DCS 0090 144	PU1 0091 145	PU2 0092 146	STS 0093 147	CCH 0094 148	MW 0095 149	SPA 0096 150	EPA 0097 151	SOS 0098 152	SGCI 0099 153	SCI 009A 154	CSI 009B 155	ST 009C 156	OSC 009D 157	PM 009E 158	APC 009F 159
A–	NBSP 00A0 160	¡ 00A1 161	¢ 00A2 162	£ 00A3 163	€ 20AC 164	¥ 00A5 165	Š 0160 166	§ 00A7 167	š 0161 168	© 00A9 169	ª 00AA 170	« 00AB 171	¬ 00AC 172	SHY 00AD 173	® 00AE 174	¯ 00AF 175
B–	° 00B0 176	± 00B1 177	² 00B2 178	³ 00B3 179	Ž 017D 180	µ 00B5 181	¶ 00B6 182	· 00B7 183	ž 017E 184	¹ 00B9 185	º 00BA 186	» 00BB 187	Œ 0152 188	œ 0153 189	Ÿ 0178 190	¿ 00BF 191
C–	À 00C0 192	Á 00C1 193	Â 00C2 194	Ã 00C3 195	Ä 00C4 196	Å 00C5 197	Æ 00C6 198	Ç 00C7 199	È 00C8 200	É 00C9 201	Ê 00CA 202	Ë 00CB 203	Ì 00CC 204	Í 00CD 205	Î 00CE 206	Ï 00CF 207
D–	Ð 00D0 208	Ñ 00D1 209	Ò 00D2 210	Ó 00D3 211	Ô 00D4 212	Õ 00D5 213	Ö 00D6 214	× 00D7 215	Ø 00D8 216	Ù 00D9 217	Ú 00DA 218	Û 00DB 219	Ü 00DC 220	Ý 00DD 221	Þ 00DE 222	ß 00DF 223
E–	à 00E0 224	á 00E1 225	â 00E2 226	ã 00E3 227	ä 00E4 228	å 00E5 229	æ 00E6 230	ç 00E7 231	è 00E8 232	é 00E9 233	ê 00EA 234	ë 00EB 235	ì 00EC 236	í 00ED 237	î 00EE 238	ï 00EF 239
F–	ð 00F0 240	ñ 00F1 241	ò 00F2 242	ó 00F3 243	ô 00F4 244	õ 00F5 245	ö 00F6 246	÷ 00F7 247	ø 00F8 248	ù 00F9 249	ú 00FA 250	û 00FB 251	ü 00FC 252	ý 00FD 253	þ 00FE 254	ÿ 00FF 255
	–0	–1	–2	–3	–4	–5	–6	–7	–8	–9	–A	–B	–C	–D	–E	–F

1.3 代码页（Code Page）

1.3.1 CP437

CP437 也就是 IBM PC 或 MS-DOS 字符，有时说的 OEM-US 或 DOS-US 就是指这个字符集，这是一种普遍运用的字符集。它的基础字符集是基于 ASCII，但又不同于 ASCII:
1). 由下图对比可知（上栏是 ASCII，下栏是 CP437），控制符 0x00~0x1F 段 CP437 是一些诸如笑脸，牌和音乐的符号。

	—0	—1	—2	—3	—4	—5	—6	—7	—8	—9	—A	—B	—C	—D	—E	—F
0—	NUL 0000 0	SOH 0001 1	STX 0002 2	ETX 0003 3	EOT 0004 4	ENQ 0005 5	ACK 0006 6	BEL 0007 7	BS 0008 8	HT 0009 9	LF 000A 10	VT 000B 11	FF 000C 12	CR 000D 13	SO 000E 14	SI 000F 15
1—	DLE 0010 16	DC1 0011 17	DC2 0012 18	DC3 0013 19	DC4 0014 20	NAK 0015 21	SYN 0016 22	ETB 0017 23	CAN 0018 24	EM 0019 25	SUB 001A 26	ESC 001B 27	FS 001C 28	GS 001D 29	RS 001E 30	US 001F 31

	—0	—1	—2	—3	—4	—5	—6	—7	—8	—9	—A	—B	—C	—D	—E	—F
0—	FSP 2007 0	☺ 263A 1	☻ 263B 2	♥ 2665 3	♦ 2666 4	♣ 2663 5	♠ 2660 6	• 2022 7	◘ 25D8 8	○ 25CB 9	◙ 25D9 10	♂ 2642 11	♀ 2640 12	♪ 266A 13	♫ 266B 14	☼ 263C 15
1—	► 25BA 16	◄ 25C4 17	↕ 2195 18	‼ 203C 19	¶ 00B6 20	§ 00A7 21	▬ 25AC 22	↨ 21A8 23	↑ 2191 24	↓ 2193 25	→ 2192 26	← 2190 27	∟ 221F 28	↔ 2194 29	▲ 25B2 30	▼ 25BC 31

0x7F 在 ASCII 的是 DEL，CP437 就是个房子样的图形 ⌂ 2302 127 。

2). 高位 0x80~0xFF，CP437 是一些欧洲字母，制表符，希腊字母等等。

3). 0x20~0x7E 是标准的 ASCII 字母。

GSM 里用到的 8bit 字符集有时就是指 CP437。

CP437 码表如下图所示：

常见字符集和编码

	—0	—1	—2	—3	—4	—5	—6	—7	—8	—9	—A	—B	—C	—D	—E	—F
0—	FSP 2007 0	☺ 263A 1	☻ 263B 2	♥ 2665 3	♦ 2666 4	♣ 2663 5	♠ 2660 6	• 2022 7	◘ 25D8 8	○ 25CB 9	◙ 25D9 10	♂ 2642 11	♀ 2640 12	♪ 266A 13	♫ 266B 14	☼ 263C 15
1—	► 25BA 16	◄ 25C4 17	↕ 2195 18	‼ 203C 19	¶ 00B6 20	§ 00A7 21	▬ 25AC 22	↨ 21A8 23	↑ 2191 24	↓ 2193 25	→ 2192 26	← 2190 27	∟ 221F 28	↔ 2194 29	▲ 25B2 30	▼ 25BC 31
2—	SP 0020 32	! 0021 33	" 0022 34	# 0023 35	$ 0024 36	% 0025 37	& 0026 38	' 0027 39	(0028 40	) 0029 41	* 002A 42	+ 002B 43	, 002C 44	- 002D 45	. 002E 46	/ 002F 47
3—	0 0030 48	1 0031 49	2 0032 50	3 0033 51	4 0034 52	5 0035 53	6 0036 54	7 0037 55	8 0038 56	9 0039 57	: 003A 58	; 003B 59	< 003C 60	= 003D 61	> 003E 62	? 003F 63
4—	@ 0040 64	A 0041 65	B 0042 66	C 0043 67	D 0044 68	E 0045 69	F 0046 70	G 0047 71	H 0048 72	I 0049 73	J 004A 74	K 004B 75	L 004C 76	M 004D 77	N 004E 78	O 004F 79
5—	P 0050 80	Q 0051 81	R 0052 82	S 0053 83	T 0054 84	U 0055 85	V 0056 86	W 0057 87	X 0058 88	Y 0059 89	Z 005A 90	[005B 91	\ 005C 92	] 005D 93	^ 005E 94	_ 005F 95
6—	` 0060 96	a 0061 97	b 0062 98	c 0063 99	d 0064 100	e 0065 101	f 0066 102	g 0067 103	h 0068 104	i 0069 105	j 006A 106	k 006B 107	l 006C 108	m 006D 109	n 006E 110	o 006F 111
7—	p 0070 112	q 0071 113	r 0072 114	s 0073 115	t 0074 116	u 0075 117	v 0076 118	w 0077 119	x 0078 120	y 0079 121	z 007A 122	{ 007B 123	| 007C 124	} 007D 125	~ 007E 126	⌂ 2302 127
8—	Ç 00C7 128	ü 00FC 129	é 00E9 130	â 00E2 131	ä 00E4 132	à 00E0 133	å 00E5 134	ç 00E7 135	ê 00EA 136	ë 00EB 137	è 00E8 138	ï 00EF 139	î 00EE 140	ì 00EC 141	Ä 00C4 142	Å 00C5 143
9—	É 00C9 144	æ 00E6 145	Æ 00C6 146	ô 00F4 147	ö 00F6 148	ò 00F2 149	û 00FB 150	ù 00F9 151	ÿ 00FF 152	Ö 00D6 153	Ü 00DC 154	¢ 00A2 155	£ 00A3 156	¥ 00A5 157	Pts 20A7 158	ƒ 0192 159
A—	á 00E1 160	í 00ED 161	ó 00F3 162	ú 00FA 163	ñ 00F1 164	Ñ 00D1 165	ª 00AA 166	º 00BA 167	¿ 00BF 168	⌐ 2310 169	¬ 00AC 170	½ 00BD 171	¼ 00BC 172	¡ 00A1 173	« 00AB 174	» 00BB 175
B—	░ 2591 176	▒ 2592 177	▓ 2593 178	| 2502 179	┤ 2524 180	╡ 2561 181	╢ 2562 182	╖ 2556 183	╕ 2555 184	╣ 2563 185	║ 2551 186	╗ 2557 187	╝ 255D 188	╜ 255C 189	╛ 255B 190	┐ 2510 191
C—	└ 2514 192	┴ 2534 193	┬ 252C 194	├ 251C 195	─ 2500 196	┼ 253C 197	╞ 255E 198	╟ 255F 199	╚ 255A 200	╔ 2554 201	╩ 2569 202	╦ 2566 203	╠ 2560 204	═ 2550 205	╬ 256C 206	╧ 2567 207
D—	╨ 2568 208	╤ 2564 209	╥ 2565 210	╙ 2559 211	╘ 2558 212	╒ 2552 213	╓ 2553 214	╫ 256B 215	╪ 256A 216	┘ 2518 217	┌ 250C 218	█ 2588 219	▄ 2584 220	▌ 258C 221	▐ 2590 222	▀ 2580 223
E—	α 03B1 224	β 03B2 225	Γ 0393 226	π 03C0 227	Σ 03A3 228	σ 03C3 229	µ 00B5 230	τ 03C4 231	Φ 03A6 232	Θ 0398 233	Ω 03A9 234	δ 03B4 235	∞ 221E 236	⌀ 2205 237	∈ 2208 238	∩ 2229 239
F—	≡ 2261 240	± 00B1 241	≥ 2265 242	≤ 2264 243	⌠ 2320 244	⌡ 2321 245	÷ 00F7 246	≈ 2248 247	° 00B0 248	∙ 2219 249	· 00B7 250	√ 221A 251	ⁿ 207F 252	² 00B2 253	■ 25A0 254	NBSP 00A0 255
	—0	—1	—2	—3	—4	—5	—6	—7	—8	—9	—A	—B	—C	—D	—E	—F

1.3.2 CP850、CP858

CP437 缺少了一些重要的西方语言字符：

1). 西班牙（Spanish）文 Á, Í, Ó, Ú，法语（French）À, Â, È, Ê, Ë, Ì, Î, Ï, Ò, Ô, Œ, œ, Ù, Û 和波兰语（Portuguese）Ã, ã, Õ, õ。
2). 有德语的原音（umlauts for German）字母 Ä, ä, Ö, ö, Ü, ü ，但 ß 需用(β，beta symbol)来表示。
3). 有斯堪的那维亚（Scandinavian）字母 Æ, æ, Å, å, Ä, ä，但缺 Ø and ø，而是分别用¢（cent）和¥（yen）来替换表示。

因为 CP437 是早期定义的，后续又定义了诸如如 CP850、CP852 或 CP737 字符集。下面重点说一下 CP850 和 CP858。

CP850 弥补了 CP437 不足的部分，具体请参看下图：

8–	Ç 00C7 ***128***	ü 00FC ***129***	é 00E9 ***130***	â 00E2 ***131***	ä 00E4 ***132***	à 00E0 ***133***	å 00E5 ***134***	ç 00E7 ***135***	ê 00EA ***136***	ë 00EB ***137***	è 00E8 ***138***	ï 00EF ***139***	î 00EE ***140***	ì 00EC ***141***	Ä 00C4 ***142***	Å 00C5 ***143***
9–	É 00C9 ***144***	æ 00E6 ***145***	Æ 00C6 ***146***	ô 00F4 ***147***	ö 00F6 ***148***	ò 00F2 ***149***	û 00FB ***150***	ù 00F9 ***151***	ÿ 00FF ***152***	Ö 00D6 ***153***	Ü 00DC ***154***	ø 00F8 ***155***	£ 00A3 ***156***	Ø 00D8 ***157***	× 00D7 ***158***	ƒ 0192 ***159***
A–	á 00E1 ***160***	í 00ED ***161***	ó 00F3 ***162***	ú 00FA ***163***	ñ 00F1 ***164***	Ñ 00D1 ***165***	ª 00AA ***166***	º 00BA ***167***	¿ 00BF ***168***	® 00AE ***169***	¬ 00AC ***170***	½ 00BD ***171***	¼ 00BC ***172***	¡ 00A1 ***173***	« 00AB ***174***	» 00BB ***175***
B–	▒ 2591 ***176***	▓ 2592 ***177***	▓ 2593 ***178***	│ 2502 ***179***	┤ 2524 ***180***	Á 00C1 ***181***	Â 00C2 ***182***	À 00C0 ***183***	© 00A9 ***184***	╣ 2563 ***185***	║ 2551 ***186***	╗ 2557 ***187***	╝ 255D ***188***	¢ 00A2 ***189***	¥ 00A5 ***190***	┐ 2510 ***191***
C–	└ 2514 ***192***	┴ 2534 ***193***	┬ 252C ***194***	├ 251C ***195***	─ 2500 ***196***	┼ 253C ***197***	ã 00E3 ***198***	Ã 00C3 ***199***	╚ 255A ***200***	╔ 2554 ***201***	╩ 2569 ***202***	╦ 2566 ***203***	╠ 2560 ***204***	═ 2550 ***205***	╬ 256C ***206***	¤ 00A4 ***207***
D–	ð 00F0 ***208***	Ð 00D0 ***209***	Ê 00CA ***210***	Ë 00CB ***211***	È 00C8 ***212***	ı 0131 ***213***	Í 00CD ***214***	Î 00CE ***215***	Ï 00CF ***216***	┘ 2518 ***217***	┌ 250C ***218***	█ 2588 ***219***	▄ 2584 ***220***	¦ 00A6 ***221***	Ì 00CC ***222***	▀ 2580 ***223***
E–	Ó 00D3 ***224***	ß 00DF ***225***	Ô 00D4 ***226***	Ò 00D2 ***227***	õ 00F5 ***228***	Õ 00D5 ***229***	µ 00B5 ***230***	þ 00FE ***231***	Þ 00DE ***232***	Ú 00DA ***233***	Û 00DB ***234***	Ù 00D9 ***235***	ý 00FD ***236***	Ý 00DD ***237***	¯ 00AF ***238***	´ 00B4 ***239***
F–	SHY 00AD ***240***	± 00B1 ***241***	‗ 2017 ***242***	¾ 00BE ***243***	¶ 00B6 ***244***	§ 00A7 ***245***	÷ 00F7 ***246***	¸ 00B8 ***247***	° 00B0 ***248***	¨ 00A8 ***249***	· 00B7 ***250***	¹ 00B9 ***251***	³ 00B3 ***252***	² 00B2 ***253***	■ 25A0 ***254***	NBSP 00A0 ***255***

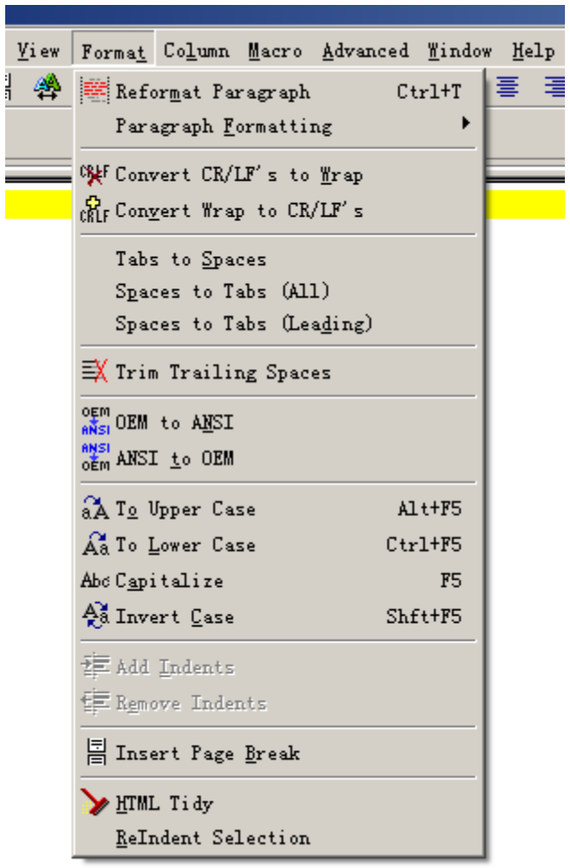
0x00~0x7F 段和 CP437 一样。

CP858 和 CP850 不同处在于，0xD5 位置的 ı（dotless -i ）换成了欧元符€（euro）。

1.3.3 CP1252

CP1252 就是 Windows-1252，也就是 ANSI 的代码页，是 ISO 8859-1 的扩展集。

Ultra Edit 工具菜单 Format 栏里有 2 个选项，一个是 OEM to ANSI，另一个是 ANSI to OEM，指的就是 CP437 和 CP1252 之间的互转。



CP1252 的码表见下图所示：

常见字符集和编码

Windows-1252 (CP1252)

	−0	−1	−2	−3	−4	−5	−6	−7	−8	−9	−A	−B	−C	−D	−E	−F
0−	NUL 0000 *0*	SOH 0001 *1*	STX 0002 *2*	ETX 0003 *3*	EOT 0004 *4*	ENQ 0005 *5*	ACK 0006 *6*	BEL 0007 *7*	BS 0008 *8*	HT 0009 *9*	LF 000A *10*	VT 000B *11*	FF 000C *12*	CR 000D *13*	SO 000E *14*	SI 000F *15*
1−	DLE 0010 *16*	DC1 0011 *17*	DC2 0012 *18*	DC3 0013 *19*	DC4 0014 *20*	NAK 0015 *21*	SYN 0016 *22*	ETB 0017 *23*	CAN 0018 *24*	EM 0019 *25*	SUB 001A *26*	ESC 001B *27*	FS 001C *28*	GS 001D *29*	RS 001E *30*	US 001F *31*
2−	SP 0020 *32*	! 0021 *33*	" 0022 *34*	# 0023 *35*	$ 0024 *36*	% 0025 *37*	& 0026 *38*	' 0027 *39*	(0028 *40*	) 0029 *41*	* 002A *42*	+ 002B *43*	, 002C *44*	- 002D *45*	. 002E *46*	/ 002F *47*
3−	0 0030 *48*	1 0031 *49*	2 0032 *50*	3 0033 *51*	4 0034 *52*	5 0035 *53*	6 0036 *54*	7 0037 *55*	8 0038 *56*	9 0039 *57*	: 003A *58*	; 003B *59*	< 003C *60*	= 003D *61*	> 003E *62*	? 003F *63*
4−	@ 0040 *64*	A 0041 *65*	B 0042 *66*	C 0043 *67*	D 0044 *68*	E 0045 *69*	F 0046 *70*	G 0047 *71*	H 0048 *72*	I 0049 *73*	J 004A *74*	K 004B *75*	L 004C *76*	M 004D *77*	N 004E *78*	O 004F *79*
5−	P 0050 *80*	Q 0051 *81*	R 0052 *82*	S 0053 *83*	T 0054 *84*	U 0055 *85*	V 0056 *86*	W 0057 *87*	X 0058 *88*	Y 0059 *89*	Z 005A *90*	[005B *91*	\ 005C *92*	] 005D *93*	^ 005E *94*	_ 005F *95*
6−	` 0060 *96*	a 0061 *97*	b 0062 *98*	c 0063 *99*	d 0064 *100*	e 0065 *101*	f 0066 *102*	g 0067 *103*	h 0068 *104*	i 0069 *105*	j 006A *106*	k 006B *107*	l 006C *108*	m 006D *109*	n 006E *110*	o 006F *111*
7−	p 0070 *112*	q 0071 *113*	r 0072 *114*	s 0073 *115*	t 0074 *116*	u 0075 *117*	v 0076 *118*	w 0077 *119*	x 0078 *120*	y 0079 *121*	z 007A *122*	{ 007B *123*	| 007C *124*	} 007D *125*	~ 007E *126*	DEL 007F *127*
8−	€ 20AC *128*		‚ 201A *130*	ƒ 0192 *131*	„ 201E *132*	… 2026 *133*	† 2020 *134*	‡ 2021 *135*	ˆ 02C6 *136*	‰ 2030 *137*	Š 0160 *138*	‹ 2039 *139*	Œ 0152 *140*		Ž 017D *142*	
9−		‘ 2018 *145*	’ 2019 *146*	“ 201C *147*	” 201D *148*	• 2022 *149*	– 2013 *150*	— 2014 *151*	˜ 02DC *152*	™ 2122 *153*	š 0161 *154*	› 203A *155*	œ 0153 *156*		ž 017E *158*	Ÿ 0178 *159*
A−	NBSP 00A0 *160*	¡ 00A1 *161*	¢ 00A2 *162*	£ 00A3 *163*	¤ 00A4 *164*	¥ 00A5 *165*	¦ 00A6 *166*	§ 00A7 *167*	¨ 00A8 *168*	© 00A9 *169*	ª 00AA *170*	« 00AB *171*	¬ 00AC *172*	SHY 00AD *173*	® 00AE *174*	¯ 00AF *175*
B−	° 00B0 *176*	± 00B1 *177*	2 00B2 *178*	3 00B3 *179*	´ 00B4 *180*	µ 00B5 *181*	¶ 00B6 *182*	· 00B7 *183*	¸ 00B8 *184*	1 00B9 *185*	º 00BA *186*	» 00BB *187*	¼ 00BC *188*	½ 00BD *189*	¾ 00BE *190*	¿ 00BF *191*
C−	À 00C0 *192*	Á 00C1 *193*	Â 00C2 *194*	Ã 00C3 *195*	Ä 00C4 *196*	Å 00C5 *197*	Æ 00C6 *198*	Ç 00C7 *199*	È 00C8 *200*	É 00C9 *201*	Ê 00CA *202*	Ë 00CB *203*	Ì 00CC *204*	Í 00CD *205*	Î 00CE *206*	Ï 00CF *207*
D−	Ð 00D0 *208*	Ñ 00D1 *209*	Ò 00D2 *210*	Ó 00D3 *211*	Ô 00D4 *212*	Õ 00D5 *213*	Ö 00D6 *214*	× 00D7 *215*	Ø 00D8 *216*	Ù 00D9 *217*	Ú 00DA *218*	Û 00DB *219*	Ü 00DC *220*	Ý 00DD *221*	Þ 00DE *222*	ß 00DF *223*
E−	à 00E0 *224*	á 00E1 *225*	â 00E2 *226*	ã 00E3 *227*	ä 00E4 *228*	å 00E5 *229*	æ 00E6 *230*	ç 00E7 *231*	è 00E8 *232*	é 00E9 *233*	ê 00EA *234*	ë 00EB *235*	ì 00EC *236*	í 00ED *237*	î 00EE *238*	ï 00EF *239*
F−	ð 00F0 *240*	ñ 00F1 *241*	ò 00F2 *242*	ó 00F3 *243*	ô 00F4 *244*	õ 00F5 *245*	ö 00F6 *246*	÷ 00F7 *247*	ø 00F8 *248*	ù 00F9 *249*	ú 00FA *250*	û 00FB *251*	ü 00FC *252*	ý 00FD *253*	þ 00FE *254*	ÿ 00FF *255*

1.4 GB 和 BIG5

GB2312

GB2312(1980 年)一共收录了 7445 个字符，包括 6763 个汉字和 682 个其它符号。汉字区的内码范围高字节从 B0-F7,低字节从 A1-FE,占用的码位是 72*94=6768。其中有 5 个空位是 D7FA-D7FE。
GB2312-80 中共收录了 7545 个字符，用两个字节编码一个字符。每个字符最高位为 0。
GB2312-80 编码简称国标码。

作用：国家简体中文字符集，兼容 ASCII。
位数：使用 2 个字节表示，能表示 7445 个符号，包括 6763 个汉字，几乎覆盖所有高频率汉字。
范围：高字节从 A1 到 F7，低字节从 A1 到 FE。将高字节和低字节分别加上 0XA0 即可得到编码。

GB12345

1990 年制定了繁体字的编码标准 GB12345-90《信息交换用汉字编码字符集第一辅助集》，目的在于规范必须使用繁体字的各种场合，以及古籍整理等。该标准共收录 6866 个汉字（比 GB2312 多 103 个字，其它厂商的字库大多不包括这些字），纯繁体的字大概有 2200 余个。
双字节编码
范围：

A1A1~FEFE
A1-A9：符号区，增加竖排符号
B0-F9：汉字区，包含 6866 个汉字

GBK

GBK 编码(Chinese Internal Code Specification)是中国大陆制订的，等同于 UCS 的新的中文编码扩展国家标准。GBK 编码能够用来同时表示繁体字和简体字，而 GB2312 只能表示简体字，GBK 是兼容 gb2312 编码的。GBK 工作小组于 1995 年 10 月，同年 12 月完成 GBK 规范。该编码标准兼容 GB2312，共收录汉字 21003 个、符号 883 个，并提供 1894 个造字码位，简、繁体字融于一库。Windows95/98 简体中文版的字库表层编码就采用的是 GBK，通过 GBK 与 UCS 之间一一对应的码表与底层字库联系。
双字节编码，GB2312-80 的扩充，在码位上和 GB2312-80 兼容。
范围：

8140~FEFE（剔除 xx7F）共 23940 个码位。
包含 21003 个汉字，包含了 ISO/IEC 10646-1 中的全部中日韩汉字。

作用：它是 GB2312 的扩展，加入对繁体字的支持，兼容 GB2312。
位数：使用 2 个字节表示，可表示 21886 个字符。
范围：高字节从 81 到 FE，低字节从 40 到 FE。

GB18030

GB 18030-2000 全称是《信息技术信息交换用汉字编码字符集基本集的扩充》，由信息产业部和原国家质量技术监督局于 2000 年 3 月 17 日联合发布，作为国家强制性标准自发布之日起实施。

为了适应信息处理技术快速发展的需要，1998 年 10 月，由信息产业部电子四所、北京大学计算机技术研究所、北大方正集团、新天地公司、四通新世纪公司、 中科院软件所、长城软件公司、中软总公司、金山软件公司和联想公司的技术人员组成标准起草组。在标准研制过程中，全国信息技术标准化技术委员会多次召集标 准起草组和知名公司对标准草案进行充分地研究论证，并

且特邀了微软公司、惠普公司、Sun 公司和 IBM 公司等参加，广泛征求意见。标准起草组经过反复斟酌 和验证，提出了标准制定原则——与 GB2312 信息处理交换码所对应的事实上的内码标准兼容，在字汇上支持 GB13000.1 的全部中、日、韩(CJK)统一汉字字符和全部 CJK 扩充 A 的字符，并且确定了编码体系和 27484 个汉字，形成兼容性、扩展性、前瞻性兼 备的方案。

GB18030-2005 最主要的变化是增加了 CJK 统一汉字扩充 B。它还去掉了单字节编码的欧元符号（0x80）。

该标准采用单字节、双字节和四字节三种方式对字符编码。

作用：它解决了中文、日文、朝鲜语等的编码，兼容 GBK。
位数：它采用变字节表示(1 ASCII，2，4 字节)。可表示 27484 个文字。
范围：1 字节从 00 到 7F；2 字节高字节从 81 到 FE，低字节从 40 到 7E 和 80 到 FE；4 字节第一三字节从 81 到 FE，第二四字节从 30 到 39。

BIG5

是目前台湾、香港地区普遍使用的一种繁体汉字的编码标准，包括 440 个符号，一级汉字 5401 个、二级汉字 7652 个，共计 13060 个汉字。 BIG5 又称大五码或五大码，1984 年由台湾财团法人信息工业策进会和五间软件公司宏碁 (Acer)、神通 (MiTAC)、佳佳、零壹 (Zero One)、大众(FIC)创立，故称大五码。Big5 码的产生，是因为当时台湾不同厂商各自推出不同的编码，如倚天码、IBM PS55、王安码等，彼此不能兼容；另一方面，台湾政府当时尚未推出官方的汉字编码，而中国大陆的 GB2312 编码亦未有收录繁体中文字。

Big5 字符集共收录 13,053 个中文字，该字符集在中国台湾使用。耐人寻味的是该字符集重复地收录了两个相同的字：“兀”(0xA461 及 0xC94A)、“毀”(0xDCD1 及 0xDDFC)。

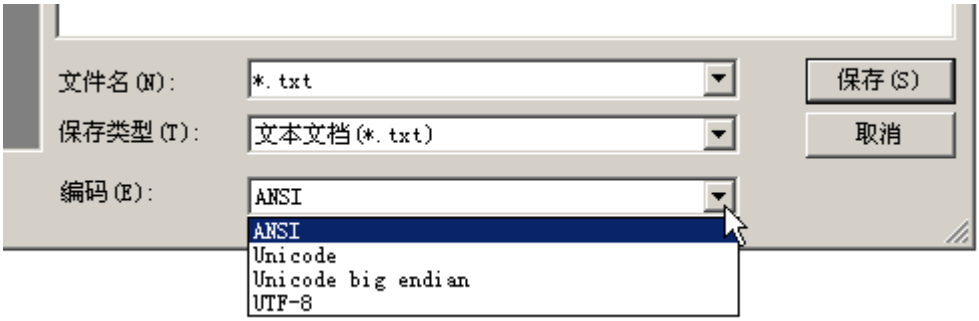
Big5 码使用了双字节储存方法，以两个字节来编码一个字。第一个字节称为“高位字节”，第二个字节称为“低位字节”。高位字节的编码范围 0xA1-0xF9，低位字节的编码范围 0x40-0x7E 及 0xA1-0xFE。

尽管 Big5 码内包含一万多个字符，但是没有考虑社会上流通的人名、地名用字、方言用字、化学及生物科等用字，没有包含日文平假名及片假字母。例如台湾视“着”为“著”的异体字，故没有收录“着”字。康熙字典中的一些部首用字(如“⺍”、“疒”、“⻍”、“⺹”等)、常见的人名用字(如“塱”、“煊”、“栢”、“喆”等) 也没有收录到 Big5 之中。

作用：统一繁体字编码。
位数：使用 2 个字节表示，表示 13053 个汉字。
范围：高字节从 A1 到 F9，低字节从 40 到 7E，A1 到 FE。

1.5 Unicode

使用 Windows 记事本的“另存为”，可以在 GBK、Unicode、Unicode big endian 和 UTF-8 这几种编码方式间相互转换。同样是 txt 文件，Windows 是怎样识别编码方式的呢？



举例来说，输入“123”，对应 Hex 编码如下：
ANSI：31 32 33
Unicode：FF FE 31 00 32 00 33 00
Unicode big endian：FE FF 00 31 00 32 00 33
UTF-8：FF FE FF FE 31 00 32 00 33 00

ANSI 就是 CP1252，Unicode 和 UTF-8 暂且放下，后续解释，先解释一下什么是 big endian。

big endian 和 little endian 是 CPU 处理多字节数的不同方式。例如“汉”字的 Unicode 编码是 6C49。那么写到文件里时，究竟是将 6C 写在前面，还是将 49 写在前面？如果将 6C 写在前面，就是 big endian，将 49 写在前面，就是 little endian。

“endian”这个词出自《格列佛游记》。小人国的内战就源于吃鸡蛋时是究竟从大头(Big-Endian)敲开还是从小头(Little-Endian)敲开，由此曾发生过六次叛乱，其中一个皇帝送了命，另一个丢了王位。

我们一般将 endian 翻译成“字节序”，将 big endian 和 little endian 称作“大尾”和“小尾”。

Unicode 字符集(简称为 UCS),国际标准组织于 1984 年 4 月成立 ISO/IEC JTC1/SC2/WG2 工作组，针对各国文字、符号进行统一性编码。Unicode 只与 ASCII 兼容，更准确地说，是和 ISO-8859-1 兼容，与 GB 码不兼容。例如“汉”字的 Unicode 编码是 6C49，而 GB 是 BABA。

从 Unicode2.0 开始，Unicode 字符集内容与 ISO10646 的 BMP（Basic Multilingual Plane）相同，其目前最新的版本是 V5.1。Unicode 编码后的大小是一样的。例如一个英文字母 "a" 和一个汉字 "好"，编码后都是占用的空间大小是一样的，都是两个字节。Unicode 可以用来表示所有语言的字符，而且是定长双字节（也有四字节的）编码，包括英文字母在内。所以也可以说它是不兼容 ISO-8859-1 编码的，也不兼容任何编码。不过，相对于 ISO-8859-1 编码来说，Unicode 编码只是在前面增加了一个 0 字节，比如字母'a'为"00 61"。

需要说明的是，定长编码便于计算机处理，而 Unicode 又可以用来表示所有字符，所以在很多软件内部是使用 Unicode 编码来处理的，比如 java。

Unicode字符集有两种格式：UCS-2和UCS-4。顾名思义，UCS-2就是用两个字节编码，UCS-4就是用4个字节（实际上只用了31位，最高位必须为0）编码。除非另外指定，否则大多数的字节都是

这样的(Bigendian convention)。将一个ASCII的文件转换成UCS-2只需简单地在每个 ASCII字节前插入0x00。如果要转换成UCS-4，则必须在每个ASCII字节前插入三个0x00。

UCS-2 有 2^16=65536 个码位，UCS-4 有 2^31=2147483648 个码位。

UCS-4 根据最高位为 0 的最高字节分成 2^7=128 个 group。每个 group 再根据次高字节分为 256 个 plane。每个 plane 根据第 3 个 字节分为 256 行 (rows)，每行包含 256 个 cells。当然同一行的 cells 只是最后一个字节不同，其余都相同。

group 0 的 plane 0 被称作 Basic Multilingual Plane, 即 BMP。或者说 UCS-4 中，高两个字节为 0 的码位被称作 BMP。

将 UCS-4 的 BMP 去掉前面的两个零字节就得到了 UCS-2。在 UCS-2 的两个字节前加上两个零字节，就得到了 UCS-4 的 BMP。而目前的 UCS-4 规范中还没有任何字符被分配在 BMP 之外。

Unicode 常见的编码方式分别是 UTF-7、UTF-8、UTF-16 和 UTF-32。

IETF 的 RFC2781 和 RFC3629 清晰严谨的描述了 UTF-16 和 UTF-8 的编码方法。

UTF-7

A Mail-Safe Transformation Format of Unicode(RFC1642)。这是一种使用 7 位 ASCII 码对 Unicode 码进行转换的编码。它的设计目的仍然是为了在只能传递 7 为编码的邮件网关中传递信息。UTF-7 对英语字母、数字和常见符号直接显示，而对其他符号用修正的 Base64 编码。

符号+和-号控制编码过程的开始和暂停。所以乱码中如果夹有英文单词，并且相伴有+号和-号，这就有可能是 UTF-7 编码。

UTF-8

UTF-8 是一种变长字节编码方式。对于某一个字符的 UTF-8 编码，如果只有一个字节则其最高二进制位为 0；如果是多字节，其第一个字节从最高位开始，连续的二进制位值为 1 的个数决定了其编码的位数，其余各字节均以 10 开头。UTF-8 最多可用到 6 个字节。

U-00000000 – U-0000007F	0*xxxxxxx*
U-00000080 – U-000007FF	110*xxxxx* 10*xxxxxx*
U-00000800 – U-0000FFFF	1110*xxxx* 10*xxxxxx* 10*xxxxxx*
U-00010000 – U-001FFFFF	11110*xxx* 10*xxxxxx* 10*xxxxxx* 10*xxxxxx*
U-00200000 – U-03FFFFFF	111110*xx* 10*xxxxxx* 10*xxxxxx* 10*xxxxxx* 10*xxxxxx*
U-04000000 – U-7FFFFFFF	1111110*x* 10*xxxxxx* 10*xxxxxx* 10*xxxxxx* 10*xxxxxx* 10*xxxxxx*

因此 UTF-8 中可以用来表示字符编码的实际位数最多有 31 位，即上表中 x 所表示的位。除去那些控制位（每字节开头的 10 等），这些 x 表示的位与 UNICODE 编码是一一对应的，位高低顺序也相同。

实际将 UNICODE 转换为 UTF-8 编码时应先去除高位 0，然后根据所剩编码的位数决定所需最小的 UTF-8 编码位数。

因此那些基本 ASCII 字符集中的字符（UNICODE 兼容 ASCII）只需要一个字节的 UTF-8 编码（7 个二进制位）便可以表示。

UTF8 编码后的大小是不一定，一般来讲，英文字母都是用一个字节表示，而汉字使用三个字节。例如一个英文字母"a" 和一个汉字 "好"，编码后占用的空间大小就不样了，前者是一个字节，后者是三个字节。编码的方法是从低位到高位。

如“汉”字的 Unicode 编码是 6C49,6C49 在 0800-FFFF 之间,所以肯定要用 3 字节模板：1110xxxx 10xxxxxx 10xxxxxx。将 6C49 写成二进制是：0110 110001 001001， 用这个比特流依次代替模板中的 x，得到：11100110 10110001 10001001，即 E6 B1 89。

值得说明的是,虽然 UTF 编码对汉字使用 3 个字节,但即使对于汉字网页,UTF 编码也会比 Unicode 编码节省，因为网页中包含了很多的英文字符。

UTF-16

UTF-16 以 16 位为单元对 UCS 进行编码，Unicode 中不同部分的字符都同样基于现有的标准。这是为了便于转换。从 0x0000 到 0x007F 是 ASCII 字符,从 0x0080 到 0x00FF 是 ISO-8859-1 对 ASCII 的扩展。希腊字母表使用从 0x0370 到 0x03FF 的代码，斯拉夫语使用从 0x0400 到 0x04FF 的代码，美国使用从 0x0530 到 0x058F 的代码，希伯来语使用从 0x0590 到 0x05FF 的代 码。中国、日本和韩国的象形文字（总称为 CJK）占用了从 0x3000 到 0x9FFF 的代码。

对于小于 0x10000 的 UCS 码，UTF-16 编码就等于 UCS 码对应的 16 位无符号整数。对于不小于 0x10000 的 UCS 码，定义了一个算法。不过由于实际使用的 UCS2，或者 UCS4 的 BMP 必然小于 0x10000，所以就目前而言，可以认为 UTF -16 和 UCS-2 基本相同。但 UCS-2 只是一个编码方案，UTF-16 却要用于实际的传输，所以就不得不考虑字节序的问题。

由于 0x00 在 c 语言及操作系统文件名等中有特殊意义,故很多情况下需要 UTF-8 编码保存文本，去掉这个 0x00。举例如下：

UTF-16: 0x0080 = 0000 0000 1000 0000
UTF-8: 0xC280 = 1100 0010 1000 0000

UTF-32

采用 4 字节。

Unicode中的汉字

在 Unicode 5.0 的 99089 个字符中，有 71226 个字符与汉字有关。它们的分布如下：

Block 名称	开始码位	结束码位	字符数
CJK 统一汉字	4E00	9FBB	20924
CJK 统一汉字扩充 A	3400	4DB5	6582
CJK 统一汉字扩充 B	20000	2A6D6	42711
CJK 兼容汉字	F900	FA2D	302
CJK 兼容汉字	FA30	FA6A	59
CJK 兼容汉字	FA70	FAD9	106
CJK 兼容汉字补充	2F800	2FA1D	542

如果不算兼容汉字，Unicode 目前支持的汉字总数是 20924+6582+42711=70217。

这里有一个细节。在早期的 Unicode 版本中，CJK 统一汉字区的范围是 0x4E00-0x9FA5，也就是我们经常提到的 20902 个汉字。当前版本的 Unicode 增加了 22 个字符，码位是 0x9FA6-0x9FBB。

UTF的字节序和BOM

UTF-8 以字节为编码单元,没有字节序的问题。UTF-16 以两个字节为编码单元,在解释一个 UTF-16 文本前，首先要弄清楚每个编码单元的字节序。例如收到一个“奎”的 Unicode 编码是 594E，“乙”的 Unicode 编码是 4E59。如果我们收到 UTF-16 字节流“594E”，那么这是 “奎”还是“乙”？

Unicode 规范中推荐的标记字节顺序的方法是 BOM。BOM 不是“Bill Of Material”的 BOM 表，而是 Byte Order Mark。BOM 是一个有点小聪明的想法。

在 UCS 编码中有一个叫做"ZERO WIDTH NO-BREAK SPACE"的字符，它的编码是 FEFF。而 FFFE 在 UCS 中是不存在的字符，所以不应该出现在实际传输中。UCS 规范建议我们在传输字节流前，先传输字符"ZERO WIDTH NO-BREAK SPACE"。

这样如果接收者收到 FEFF，就表明这个字节流是 Big-Endian 的；如果收到 FFFE，就表明这个字节流是 Little-Endian 的。因此字符"ZERO WIDTH NO-BREAK SPACE"又被称作 BOM。

UTF-8 不需要 BOM 来表明字节顺序，但可以用 BOM 来表明编码方式。字符"ZERO WIDTH NO-BREAK SPACE"的 UTF-8 编码是 EF BB BF。所以如果接收者收到以 EF BB BF 开头的字节流，就知道这是 UTF-8 编码了。

Windows 就是使用 BOM 来标记文本文件的编码方式的。

1.6 MIME

MIME 是“多用途网际邮件扩充协议”的缩写，在 MIME 协议之前，邮件的编码曾经有过 UUENCODE 等编码方式 ，但是由于 MIME 协议算法简单，并且易于扩展，现在已经成为邮件编码方式的主流，不仅是用来传输 8 bit 的字符，也可以用来传送二进制的文件 ，如邮件附件中的图像、音频等信息，而且扩展了很多基于 MIME 的应用。从编码方式来说，MIME 定义了两种编码方法 Base64 与 QP(Quote-Printable)

Base64

按照 RFC2045 的定义，Base64 被定义为：Base64 内容传送编码被设计用来把任意序列的 8 位字节描述为一种不易被人直接识别的形式。

在设计这个编码的时候，设计人员最主要考虑了 3 个问题：

1. 是否加密？
2. 加密算法复杂程度和效率
3. 如何处理传输？

加密是肯定的，但是加密的目的不是让用户发送非常安全的 Email。这种加密方式主要就是“防君子不防小人”。即达到一眼望去完全看不出内容即可。

基于这个目的加密算法的复杂程度和效率也就不能太大和太低。和上一个理由类似，MIME 协议等用于发送 Email 的协议解决的是如何收发 Email，而并不 是如何安全的收发 Email。因此算法的复杂程度要小，效率要高，否则因为发送 Email 而大量占用资源，路就有点走歪了。

但是，如果是基于以上两点，那么我们使用最简单的恺撒法即可，为什么 Base64 看起来要比恺撒法复杂呢？这是因为在 Email 的传送过程中，由于历史原 因，Email 只被允许传送 ASCII 字符，即一个 8 位字节的低 7 位。因此，如果您发送了一封带有非 ASCII 字符（即字节的最高位是 1）的 Email 通 过有“历史问题”的网关时就可能会出现问题。网关可能会把最高位置为 0！很明显，问题就这样产生了！因此，为了能够正常的传送 Email，这个问题就必须 考虑！所以，单单靠改变字母的位置的恺撒之类的方案也就不行了。关于这一点可以参考 RFC2046。

基于以上的一些主要原因产生了 Base64 编码。

Base64 编码要求把 3 个 8 位字节（3*8=24）转化为 4 个 6 位的字节（4*6=24），之后在 6 位的前面补两个 0，形成 8 位一个字节的形式。

QP (Quote-Printable)

通常缩写为“Q”方法，其原理是把一个 8 bit 的字符用两个 16 进制数值表示，然后在前面加“=” 。 所 以 我 们 看 到 经 过 QP 编 码 后 的 文 件 通 常 是 这 个 样 子 ：=B3=C2=BF=A1=C7=E5=A3=AC=C4=FA=BA=C3=A3=A1。

1.7 GSM 7-bit Default Alphabet

此表请参看 ETSI TS 123.038 的定义说明，归纳如下图所示：

GSM 7bit default alphabet table															
	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	@	£	$	¥	è	é	ù	ì	ò	Ç	LF	Ø	ø	CR	Å	å
1x	Δ	_	Φ	Γ	Λ	Ω	Π	Ψ	Σ	Θ	Ξ	ESC	Æ	æ	ß	É
2x	SP	!	"	#	¤	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	¡	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	Ä	Ö	Ñ	Ü	§
6x	¿	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	ä	ö	ñ	ü	à
1B 0x											1					
1B 1x					^							2				
1B 2x									{	}						\
1B 3x													[	~	]	
1B 4x	|															
1B 5x																
1B 6x						€										
1B 7x																

1）. This code is defined as a Page Break character and may be used for example in compressed CBS messages. Any mobile station which does not understand the GSM 7 bit default alphabet table extension mechanism will treat this character as Line Feed.
2）. This code value is reserved for the extension to another extension table. On receipt of this code, a receiving entity shall display a space until another extension table is defined. It is not intended that this extension mechanism should be used as an alternative to UCS2 to enhance the 7bit default alphabet character repertoire for national specific character sets.
3）. This code represents the EURO currency symbol. The code value is that used for the character "e". Therefore a receiving entity which is incapable of displaying the EURO currency symbol will display the character "e" instead.

NOTE: This code is an escape to an extension of the GSM 7 bit default alphabet table. A receiving entity which does not understand the meaning of this escape mechanism shall display it as a space character.

具体请参见 SMS 相关 3GPP 协议。

1.8 结语

最后，希望看了这篇文章之后不要混淆字符集和字符编码的概念，还有对以上谈到的各种编码方式的原因有大致的了解，象 UTF-8 这类是为了解析 Unicode 这种字符集而制定，而 base64 这类是为了解决实际的网络应用而制定。

2. 转码

码表之间的相互转换代码请参考 convcode.c 。