

## Appendix A – The EBCDIC Character Set

The EBCDIC (Extended Binary Coded Decimal Interchange Code) was developed in 1963 and 1964 by IBM for use on its System/360 line of computers. It was created as an extension of the BCD (Binary Coded Decimal) encoding that existed at the time.

EBCDIC code uses eight binary bits to encode a character set; it can encode 256 characters. The codes are binary numeric values, traditionally represented as two hexadecimal digits.

Character codes 0x00 through 0x3F and 0xFF represent control characters.

0x0D is the code for a carriage return; this moves the cursor back to the left margin.

0x20 is used by the ED (Edit) instruction to represent a packed digit to be printed.

0x21 is used by the ED (Edit) instruction to force significance.

All digits, including leading 0's, from this position will be printed.

0x25 is the code for a line feed; this moves the cursor down but not horizontally.

0x2F is the BELL code; it causes the terminal to emit a “beep”.

Character codes 0x40 through 0x7F represent punctuation characters.

0x40 is the code for a space character: “ ”.

0x4B is the code for a decimal point: “.”.

0x4E is the code for a plus sign: “+”.

0x50 is the code for an ampersand: “&”.

0x5B is the code for a dollar sign: “\$”.

0x5C is the code for an asterisk: “\*”.

0x60 is the code for a minus sign: “-”.

0x6B is the code for a comma: “,”.

0x6F is the code for a question mark: “?”.

0x7C is the code for the commercial at sign: “@”.

Character codes 0x81 through 0xA9 represent the lower case Latin alphabet.

0x81 through 0x89 represent the letters “a” through “i”,

0x91 through 0x99 represent the letters “j” through “r”, and

0xA2 through 0xA9 represent the letters “s” through “z”.

Character codes 0xC1 through 0xE9 represent the upper case Latin alphabet.

0xC1 through 0xC9 represent the letters “A” through “I”,

0xD1 through 0xD9 represent the letters “J” through “R”, and

0xE2 through 0xE9 represent the letters “S” through “Z”.

Character codes 0xF0 through 0xF9 represent the digits “0” through “9”.

### **NOTES:**

1. The control characters are mostly used for network data transmissions. The ones listed above appear frequently in user code for terminal I/O.
2. There are gaps in the codes for the alphabetical characters. This is due to the origins of the codes for the upper case alphabetic characters in the card codes used on the IBM-029 card punch.
3. One standard way to convert an EBCDIC digit to its numeric value is to subtract the hexadecimal number 0xF0 from the character code.

### An Abbreviated Table: The Common EBCDIC

Code	Char.	Comment	Code	Char.	Comment	Code	Char.	Comment
			80			C0	}	Right brace
			81	a		C1	A	
			82	b		C2	B	
			83	c		C3	C	
			84	d		C4	D	
			85	e		C5	E	
			86	f		C6	F	
			87	g		C7	G	
0C	FF	Form feed	88	h		C8	H	
0D	CR	Return	89	i		C9	I	
16	BS	Back space	90			D0	{	Left brace
25	LF	Line Feed	91	j		D1	J	
27	ESC	Escape	92	k		D2	K	
2F	BEL	Bell	93	l		D3	L	
40	SP	Space	94	m		D4	M	
4B	.	Decimal	95	n		D5	N	
4C	<		96	o		D6	O	
4D	(		97	p		D7	P	
4E	+		98	q		D8	Q	
4F		Single Bar	99	r		D9	R	
50	&		A0			E0	\	Back slash
5A	!		A1	~	Tilde	E1		
5B	\$		A2	s		E2	S	
5C	*		A3	t		E3	T	
5D	)		A4	u		E4	U	
5E	;		A5	v		E5	V	
5F		Not	A6	w		E6	W	
60	-	Minus	A7	x		E7	X	
61	/	Slash	A8	y		E8	Y	
6A	‡	Dbl. Bar	A9	z		E9	Z	
6B	,	Comma	B0	^	Carat	F0	0	
6C	%	Percent	B1			F1	1	
6D	_	Underscore	B2			F2	2	
6E	>		B3			F3	3	
6F	?		B4			F4	4	
79	'	Apostrophe	B5			F5	5	
7A	:	Colon	B6			F6	6	
7B	#	Sharp	B7			F7	7	
7C	@	At Sign	B8			F8	8	
7D	'	Apostrophe	B9			F9	9	
7E	=	Equals	BA	[	Left Bracket			
7F	"	Quote	BB	]	R. Bracket			