

APPENDIX B

The ASCII Character Set

Originally, the American Standard Code for Information Interchange (ASCII) specified the meaning of code numbers transmitted across telephone lines one byte at a time. Frequently, the data were also stored on paper tape. These data controlled communication between two simple devices like a teletype machines. They had to not only deliver characters, but also manage the communications link and control the behavior of the teletype by forcing the print mechanism back to the first column (Carriage Return, CR), advancing the platen to the next print row (Line Feed, LF) or skipping to the column whose number was the next multiple of 8 (tabbing, HT). The first 32 values were set aside as non-printable characters that performed these control tasks. While most of the control characters are now unused, we still make use of the three mentioned above as ‘\r’, ‘\n’, and ‘\t’ to control the behavior of text presented in a window or on a document.

The original ASCII table defined a mapping, whereby a specific set of printable characters was assigned the numerical values 32–126. This was sufficient to represent the number symbols, the lowercase and uppercase alphabet, and all the common punctuation marks. However, as the need arose to represent more international characters, this numerical range was inadequate, and the next 128 values were assigned to meet this need. There is no universal agreement on this second mapping. The following table shows the first 128 values used by the MATLAB language.

When a still broader set of characters was required by the international community, it became necessary to use multiple bytes to encode the symbols. A Unicode Character Set was defined, followed by an international agreement on how to transmit these codes efficiently called UTF-8, the UCS Transformation Format.

Two totally irrelevant historical observations:

1. Astute observers will note that the values 0–127 occupy only the lower 7 bits of one byte of data. The 8th bit was used as an error detection bit during transmission. An agreement was required in transmitting between two machines as to the *parity* of the transmission. Even parity meant that there would always be an

B-2

Appendix B The ASCII Character Set

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
16	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
32		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
48	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
64	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
96	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
112	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

- even number of bits set, and the 8th bit was set or reset to ensure that this was true. If there were an odd number of bits set in a byte, the system knew that the data had been corrupted.
- ASCII 127 is another non-printing control character used when editing paper tape. Since the value 127 has all the bits set and a hole in the tape signified 1, if operators made a mistake when typing a message, they would back the tape up in the punch and hit DEL to make holes all across the byte, thereby erasing the erroneous character.