

The ASCII font package

A. Syropoulos* and R.W.D. Nickalls†

30 May, 2006

Abstract

The `ascii` package is a $\text{\LaTeX} 2_{\epsilon}$ implementation of the earlier $\text{\LaTeX} 2.09$ version¹, and provides glyph and font access commands which allow the ASCII font to be easily typeset. The ASCII font is encoded according to the *IBM PC Code Page 437 C0 Graphics*.

	'0	'1	'2	'3	'4	'5	'6	'7	
'00x	␣	⊕	⊗	♥	♦	♣	♠	•	"0x
'01x	■	○	◉	♂	♀	♪	♫	⊛	
'02x	▶	◀	↕	!!	¶	§	—	‡	"1x
'03x	↑	↓	→	←	L	↔	▲	▼	
'04x	□	!	"	#	\$	%	&	'	"2x
'05x	()	*	+	,	-	.	/	
'06x	0	1	2	3	4	5	6	7	"3x
'07x	8	9	:	;	<	=	>	?	
'10x	@	A	B	C	D	E	F	G	"4x
'11x	H	I	J	K	L	M	N	O	
'12x	P	Q	R	S	T	U	V	W	"5x
'13x	X	Y	Z	[\]	^	_	
'14x	`	a	b	c	d	e	f	g	"6x
'15x	h	i	j	k	l	m	n	o	
'16x	p	q	r	s	t	u	v	w	"7x
'17x	x	y	z	{		}	~	△	
	"8	"9	"A	"B	"C	"D	"E	"F	

Table 1: The ASCII font.

*Xanthi, Greece, email: asyropoulos@yahoo.com

†Department of Anaesthesia, City Hospital Campus, Nottingham University Hospitals, Nottingham, UK; email: dicknickalls@compuserve.com

¹Ramasubramanian R, Nickalls RWD and Reed, M (1994). Since this article is not currently available in the TUGboat archive on CTAN, we include it in this package for completeness.

1 Introduction

This package makes available the graphical representation of the ASCII characters as defined in the *IBM PC Code Page 437 C0 Graphics*. Since it is sometimes necessary to be able to typeset the glyphs representing the ASCII control characters (typically in programming or interface documents), we have created a new font that contains all the necessary glyphs. Table ?? shows the the graphical representation of the various ASCII control characters. Notice that these graphical representations are now part of the Unicode standard as individual characters, which, however, do not represent the actual control characters. In other words, the visual representations of the the control characters are now Unicode characters. In addition, we have included the “NO BREAK SPACE” character, since the accompanying font includes the standard ISO8859-7 characters.

This package is a L^AT_EX 2_ε update on the original 1994 L^AT_EX 2.09 version. Since the original TUGboat article (Ramasubramanian, Nickalls and Reed, 1994) is not currently available in the journal’s archive on CTAN, we include it in this package for completeness (see `asciisty1994.tex`). In the original ASCII package the character ASCII 124 (glyph `|`, `\asciivert`) was incorrectly represented as the `!` glyph (`\splitvert`). We correct this error in the present package, but for the purposes of backward compatibility we have continued to make the `\splitvert` command available for `|`.

2 Installation

The ASCII package contains the following files:

```
ascii.dtx
ascii.ins
ASCII.afm
ASCII.tfm
ASCII.pfb
ascii.map
ascii.sty
README
ascii2006.tex   This file
ascii2006.pdf
ascii2006.ps
asciisty1994.tex   The original 1994 TUGboat article on ASCII.sty
asciisty1994.pdf
asciisty1994.ps
```

Installation involves (a) placing the files into the appropriate directories, (b) adding an additional map-file command to some configuration files (e.g. for `dvips` and `pdftex`), (c) updating the T_EX file database. These are now described in turn.

Note that since the internal fontname is ASCII, it is important to maintain the uppercase name for the three font files (`.afm`, `.tfm`, `.pfb`).

Placing the files

The various files should be placed where your T_EX system will be able to find them—either in the main directory tree, or in the user’s local directory tree. For example, Linux systems which use the standard T_EX Directory Structure (TDS) typically locate the main `.../texmf/` directory tree at `/usr/share/texmf/...`, with the corresponding local tree being typically located at `/usr/local/share/texmf/...`

- `ascii.sty` → `.../texmf/tex/latex/ascii/`
- `ASCII.afm` → `.../texmf/fonts/afm/public/ascii/`

- ASCII.tfm → `.../texmf/fonts/tfm/public/ascii/`
- ASCII.pfb → `.../texmf/fonts/type1/public/ascii/`
- ascii.map → `.../texmf/fonts/dvips/ascii/`
- *.tex, *.dvi, *.pdf, *.ps → `.../texmf/doc/ascii/`

Update dvips configuration file (`config.ps`)

We let dvips know the location of the `ascii.map` file by including the following command

```
p +ascii.map
```

in the appropriate place in the dvips configuration file (`config.ps`) which is typically located at `.../texmf/dvips/config/config.ps`

Update pdftex configuration file (`pdftex.cfg`)

We let pdftex know the location of the `ascii.map` file by including the following map command

```
map +ascii.map
```

in the appropriate place in the pdftex configuration file (`config.ps`) which is typically located at `.../texmf/pdftex/config/pdftex.cfg`

Update T_EX file database

Finally, once all the package files are correctly located and the configuration files updated, we need to update the T_EX file database. On a Linux teTeX system this is done by running the `texhash` command (as root) as follows:

```
# texhash
```

Now, the ASCII package can be used in a L^AT_EX document by using the following command in the preamble.

```
\usepackage{ascii}
```

3 The ascii font

Control codes

The commands for accessing the control characters are given in Table ???. For example, a `\large` double music note ♯ (`\S0`) is typeset using the command `\large\S0`.

Other symbols

The standard access commands for the remaining glyphs of the ASCII font (see Table ???) do not give the expected results, and so we have had to define a number of commands to provide access to the more useful characters—these are shown in Table ???. Notice that all these glyph access commands have been implemented using the `xspace` package and therefore one does need to take any special precaution when using them.

The command `\textascii` is a font switching command and its argument is typeset using the ASCII font. For example, the glyph for the ‘synchronous idle’ control code `—` (`\SYN`, see Table ???) which is most easily typeset using the intuitive command `\large\SYN`, can also be typeset using this character’s ASCII font hex code `16h` as in the command `\large\textascii{\char"16}` i.e. `—`.

Command	Character
<code>\asciispace</code>	␣
<code>\asciiquotedbl</code>	"
<code>\asciishash</code>	#
<code>\asciidollar</code>	\$
<code>\asciipercent</code>	%
<code>\asciampersand</code>	&
<code>\asciiquoteacute</code>	'
<code>\asciibackslash</code>	\
<code>\asciicircum</code>	^
<code>\asciiunderscore</code>	_
<code>\asciiquotegrave</code>	`
<code>\asciilbrace</code>	{
<code>\asciivert</code>	
<code>\asciirbrace</code>	}
<code>\asciitilde</code>	~
<code>\splitvert</code>	!

Table 2:

ASCII control codes						
Dec	Code	Hex	Name	Glyph	ASCII.STY	Unicode
0	NUL	00h		^_\bullet	\NUL	NUL
1	SOH	01h	Start of heading	☺	\SOH	2400h
2	STX	02h	Start of text	☹	\STX	263Ah
3	ETX	03h	End of text	♥	\ETX	263Bh
4	EOT	04h	End of transmission	♦	\EOT	2665h
5	ENQ	05h	Enquiry	♣	\ENQ	2666h
6	ACK	06h	Acknowledge	♠	\ACK	2663h
7	BEL	07h	Bell	•	\BEL	2660h
8	BS	08h	Back space	◻	\BS	2022h
9	HT	09h	Horizontal tab	○	\HT	25D8h
10	LF	0Ah	Linefeed	◻	\LF	25EFh
11	VT	0Bh	Vertical tab	♂	\VT	25D9h
12	FF	0Ch	Formfeed	♀	\FF	2642h
13	CR	0Dh	Carriage return	♂	\CR	2640h
14	SO	0Eh	Shift out	♂	\SO	266Ah
15	SI	0Fh	Shift in	♂	\SI	266Bh
16	DLE	10h	Data link escape	♂	\DLE	263Ch
17	DC1	11h	Device control 1 (XON)	▶	\DCa	25BAh
18	DC2	12h	Device control 2	◀	\DCb	25C4h
19	DC3	13h	Device control 3 (XOFF)	↕	\DCc	2195h
20	DC4	14h	Device control 4	⚡	\DCd	203Ch
21	NAK	15h	Negative acknowledge	§	\NAK	00B6h
22	SYN	16h	Synchronous idle	§	\SYN	00A7h
23	ETB	17h	End transmission block	▬	\ETB	25ACh
24	CAN	18h	Cancel	↕	\CAN	21A8h
25	EM	19h	End of medium	↑	\EM	2191h
26	SUB	1Ah	Substitute (EOF)	↓	\SUB	2193h
27	ESC	1Bh	Escape	→	\ESC	2192h
28	FS	1Ch	File separator	←	\FS	2190h
29	GS	1Dh	Group separator	↔	\GS	2319h
30	RS	1Eh	Record separator	▲	\RS	2194h
31	US	1Fh	Unit separator	▼	\US	25B2h
127	DEL	7Fh	Delete	◻	\DEL	2302h
—	—	A0h	—	^_\bullet	\NBSP	00A0h
						Unicode name
						White smiling face
						Black smiling face
						Black heart suit
						Black diamond suit
						Black club suit
						Black space suit
						Bullet
						Inverse bullet
						Large circle
						Inverse white circle
						Male sign
						Female sign
						Eighth note
						Beamed 16th notes
						White sun with rays
						Black right-pointing pointer
						Black left-pointing pointer
						Up down arrow
						Double exclamation mark
						Pilcrow sign
						Section sign
						Black rectangle
						Up down arrow with base
						Upwards arrow
						Downwards arrow
						Rightwards arrow
						Leftwards arrow
						Turned not sign
						Left right arrow
						Black up-pointing triangle
						Black down-pointing triangle
						NO BREAK SPACE

Table 3: ASCII table: the control codes.

4 History

The character code *7-bit American Standard Code for Information Interchange* (ASCII) is at the very heart of computing technology, and consists of two main components, namely (a) the control codes (ASCII 0–31, 127), and (b) all the rest (ASCII 32–126). ASCII originally developed from a 7-bit teleprinter code used by Bell Data Services, initially becoming an ASA standard in 1963. This was soon revised to include lowercase characters, among others, (ANSI 1967). The latest revision is known as ANSI-X3.4-1986. A useful reference is the ASCII entry in the online *Wikipedia* (<http://en.wikipedia.org/wiki/ASCII>).

The ASCII font represents the first 127 characters of the original character set of the IBM PC (August 1981) known as ‘Code page 437’, which can be seen under this entry in *Wikipedia*. Apparently, the glyphs were chosen to be the same as those used by the widely used word-processors made by Wang Laboratories (http://en.wikipedia.org/wiki/Code_page_437).

Control codes

The 33 control characters (ASCII 0–31 plus DEL) are classified into six functional categories as follows. For a detailed analysis and description of the use of each control code, see Appendix I in Ramasubramanian and Nickalls (1995).

- Transmission control characters (SOH, STX, ETX, EOT, ENQ, ACK, DLE, NAK, SYN, ETB)
- Format effectors (BS, HT, LF, VT, FF, CR)
- Code extension control characters (SO, SI, ESC)
- Device control characters (CD1, DC2, DC3, DC4)
- Informations separators (FS, GS, RS, US)
- Other control characters (NUL, BEL, CAN, EM, SUB, DEL)

References

- Ramasubramanian R., Nickalls RWD & Reed MA (1994). ASCII.STY A new style-option and encoded font with IBM graphics control characters for use with T_EX & L^AT_EX. *TUGboat*; 15(2), 98–103.
- Ramasubramanian R and Nickalls RWD (1995). Interfacing the IBM PC to medical equipment; the art of serial communication. (Cambridge University Press).