

FONT.FMT

PSIONICS FILE - FONT.FMT

=====

Format of Font files

Last modified 1994-09-13

=====

A font file holds a font for loading with the GLOADFONT keyword. There are twokinds of font files, standard and fast. The formats are related; the notation "A // B" means A for normal fonts, and B for fast fonts.

The font file consists of a header, a width table, and a bitmap, with nointervening space. The header is 62 bytes, and has the format:

Offset 0 (byte): 'F' (70)

Offset 1 (byte): '0' (79) // 'N' (78)

Offset 2 (byte): 'N' (78) // '1' (49)

Offset 3 (byte): 227 // 197

Offset 4 (byte): 48 // 16

Offset 5 (byte): 48 // 16

Offset 6 (word): checksum (see below) Offset 8 (word): size from offset 10 to end of bitmap data

Offset 10 (word): lowest character code

Offset 12 (word): highest character code

Offset 14 (word): height of font

Offset 16 (word): descent of font

Offset 18 (word): ascent of font

Offset 20 (word): width of numeric characters

Offset 22 (word): widest character in font

Offset 24 (word): flag bits:

Bit 0: the font uses ASCII for codes 32 to 126 Bit 1: the font uses Code Page 850 for codes 128 to 255

Bit 2: bold font

Bit 3: italic font

Bit 4: serified font

Bit 5: monospaced font

Offset 26 to 41: font name (padded with spaces) Offset 42 (word): [appears to be size of the width table in bytes]

Offset 44 (word): [unknown: 0, 0, 0, 0]

Offset 46 (word): [unknown: 0, 0, 0, 0]

Offset 48 (word): [unknown: -9, -7, 72, 46] Offset 50 (word): [appears to be height of font again: 8, 8, 6, 9]

Offset 52 (word): [appears to be width of bitmap in bytes: 256, 256, 10, 6] Offset 54 (word): [unknown: 0, 0, 0, 0]

Offset 56 (word): [appears to be 8 * height of font: 64, 64, 48, 72] Offset 58 (word): [unknown: 2, 2, 2, 2]

Offset 60 (word): [unknown: 0, 0, 0, 0]

The meanings of the fields from offset 42 onwards are not known. The numbers shown are their values in the system normal font, the system bold font, the system digit font, and the Spreadsheet small font (the first two are fast

fonts and the latter two normal fonts).

The checksum is the $X^{16}+X^{12}+X^5+1$ polynomial, applied to the width table and the bitmap. See the system call GenCrc.

The header is followed, at offset 62, by the width table. For normal fonts, the header contains one word for each character in the font, plus an extra word. Thus if the font contains characters 48 to 52, there will be 6 words in the table. The first word in the table corresponds to the lowest character code, up to the last word but one, which corresponds to the highest character code. For each entry except the last, if the character exists in the font, then the corresponding word is twice the horizontal position of the start of the character in the bitmap. If the character does not exist, then the word is the same as the following word, but with the bottom bit set. The final entry in the table is twice the width of the bitmap. Thus the bottom bit indicates

whether the character exists or not, and for any character which exists, its width is given by subtracting the entry from the following entry and dividing by two (this applies equally to the last character, which is why the extra entry is there).

For fast fonts, the width table is 256 bytes, from offsets 62 to 317 inclusive. Each byte holds the width (from 0 to 8) of the corresponding character.

The bitmap immediately follows the width table (thus for normal fonts it starts at offset $(66 + (\text{max_code} - \text{min_code}) * 2)$, while for fast fonts it starts at offset 318). Its size is calculated from the size field in the header, and will always be a multiple of the height. The data for each row is kept together, with the rows in order from top to bottom. For each row, the data represents blocks of 8 pixels per byte; the bytes are in left to right order, and the least significant bit

in the byte represents the leftmost pixel of the block. For normal fonts, each character occupies a number of columns, with all the characters that exist in the font being placed side by side in code order.

For a fast font, the character occupies the left side of a block of 8 columns, again in code order (i.e. character code C occupies byte C of each row, so its top row is at offset $318+C$, its second row at offset $574+C$, and so on).

Revision #1

Created Thu, Jan 24, 2019 10:25 AM by Alex

Updated Thu, Jan 24, 2019 10:26 AM by Alex