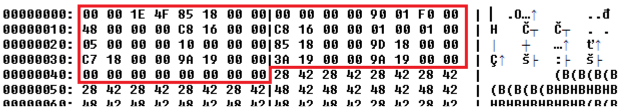
HMI file format



```
00000000: 00 00 1E 4F 85 18 00 00|00 00 00 00 90 01 F0 00  | |  .O…↑        ..đ
00000010: 48 00 00 00 C8 16 00 00|C8 16 00 00 01 00 01 00  | H    Č┬   Č┬   .  .
00000020: 05 00 00 00 10 00 00 00|85 18 00 00 9D 18 00 00  | |     ┼    …↑   ť↑
00000030: C7 18 00 00 9A 19 00 00|3A 19 00 00 9A 19 00 00  | Ç↑   š├   :├   š├
00000040: 00 00 00 00 00 00 00 00 28 42 28 42 28 42 28 42  |         (B(B(B(B
00000050: 28 42 28 42 28 42 28 42|48 42 48 42 48 42 48 42  | (B(B(B(BHBHBHBHB
00000060: 48 42 48 42 48 42 48 42|48 42 48 42 28 42 28 42  | HBHBHBHBHBHB(B(B
```

<u>File header</u>
block size = 72 bytes
position in file = 0x00000000

Offset	Length	Example*	Description
0x0000	BYTE	00	screen orientation (00 = horizontal, 01 = vertical)
0x0001	WORD	00 1E	major and minor version (00 1E = v0.30)
0x0003	BYTE	4F	?
0x0004	DWORD	00 00 18 85	absolute start positon of page data
0x0008	DWORD	00 00 00 00	?
0x000C	WORD	01 90	screen horizontal resolution in pixels (01 90 = 400)
0x000E	WORD	00 F0	screen vertical resolution in pixels (00 F0 = 240)
0x0010	DWORD	00 00 00 48	absolute start position of “image data”
0x0014	DWORD	00 00 16 C8	absolute start position of “font data”
0x0018	DWORD	00 00 16 C8	absolute start position of “component attributes data”
0x001C	WORD	00 01	number of pages
0x001E	WORD	00 01	number of components (including pages)
0x0020	WORD	00 05	number of images
0x0022	WORD	00 00	number of fonts
0x0024	WORD	00 10	number of component attributes
0x0026	WORD	00 00	?
0x0028	DWORD	00 00 18 85	absolute start position of “page data”
0x002C	DWORD	00 00 18 9D	absolute start position of “component data”
0x0030	DWORD	00 00 18 C7	absolute start position of “image property data”
0x0034	DWORD	00 00 19 9A	absolute start position of “font property data”
0x0038	DWORD	00 00 19 3A	absolute start position of “component attributes’ index data”
0x003C	DWORD	00 00 19 9A	total size of file (including header and all data)
0x0040	DWORD	00 00 00 00	?
0x0044	DWORD	00 00 00 00	?

<u>Page data</u>
block size = 24 bytes
size = block size x count stored in the file header at 0x001C
position in file = stored in the file header at 0x0028

Offset	Length	Example*	Description
0x0000	15 BYTEs		page name (00 padded and 00 terminated)
0x000F	BYTE	00	dummy
0x0010	WORD	00 00	starting component ID
0x0012	WORD	00 07	ending component ID
0x0016	DWORD	FF FF FF FF	terminator

Examples

68 6F 6D 65 00 00 00 00 00 00 00 00 00 00 00 00 00 00 07 00 FF FF FF FF
74 65 78 74 20 62 75 74 74 6F 6E 00 00 00 00 00 08 00 16 00 FF FF FF FF
70 72 6F 67 72 65 73 73 20 62 61 72 00 00 00 00 17 00 2B 00 FF FF FF FF

<u>Component data</u>
block size = 42 bytes
size = block size x count stored in the file header at 0x001E
position in file = stored in the file header at 0x002C

Offset	Length	Example*	Description
0x0000	15 CHARs		component name (00 padded and 00 terminated)
0x000F	BYTE	00	dummy
0x0010	WORD	00 00	start of ?
0x0012	WORD	00 04	length of ?
0x0014	WORD	00 0C	start X position on screen (00 0C = at pixel 12)
0x0016	WORD	00 A1	start Y position on screen (00 A1 = at pixel 161)
0x0018	WORD	00 7E	end X position -1 on screen (00 7E = at pixel 127, width is 115 pixels)
0x001A	WORD	00 B8	end Y position -1 on screen (00 B8 = at pixel 185, height is 24 pixels)
0x001C	WORD	00 01	load behaviour (00 00 = load commands, 00 01 = autoloading)
0x001E	8 BYTEs		?
0x0026	WORD	00 00	starting parameter (first ID in parameter data list is 0)
0x0028	WORD	00 0B	ending parameter

Examples

68 6F 6D 65 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04 00 00 00 00 00 8F 01 EF 00 01 00 **0C 00 0F 00 FF FF FF FF** 00 00 0B 00
74 30 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04 00 29 00 0C 00 A1 00 7E 00 B8 00 01 00 **12 00 16 00 FF FF 1B 00** 0C 00 1F 00
74 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 2D 00 29 00 93 00 A1 00 09 01 B8 00 01 00 **12 00 16 00 FF FF 1B 00** 20 00 33 00

*Data is stored in little-endian, examples shown are big-endian

Image properties data

block size = 23 bytes
size = block size x count stored in the file header at 0x0020
position in file = stored in the file header at 0x0030

Offset	Length	Example*	Description
0x0000	DWORD	00 00 00 00	?
0x0004	BYTE	00	?
0x0005	DWORD	00 00 09 00	relative start position of font data
0x0009	WORD	00 18	width of image (00 18 = 24 pixels)
0x000B	WORD	00 18	height of image (00 18 = 24 pixels)
0x000D	DWORD	00 00 04 80	size of image data
0x0011	BYTE	01	?
0x0012	DWORD	00 00 04 80	size of image data ?
0x0016	BYTE	00	?

Examples

```
00 00 00 00 00 00 00 00 00 18 00 18 00 80 04 00 00 01 80 04 00 00 00
00 00 00 00 00 80 04 00 00 18 00 18 00 80 04 00 00 01 80 04 00 00 00
00 00 00 00 00 00 00 00 00 E0 01 10 01 00 FC 03 00 01 00 FC 03 00 00
```

Image data

size = size of image data at image property 0x000D
position in file = stored in the file header at 0x0010 + relative start position of image data at image property 0x0005
Image data is 2-bytes per pixel, RGB565 image data in little-endian byte order.

Font properties data

block size = 27 bytes
size = block size x count stored in the file header at 0x0022
position in file = stored in the file header at 0x0034

Offset	Length	Example*	Description
0x0000	DWORD	00 00 00 00	?
0x0004	BYTE	00	character set (00 = English charset, 01 = Chinese charset)
0x0005	BYTE	20	width of a character (20 = 32 pixels)
0x0006	BYTE	20	height of a character (20 = 32 pixels)
0x0007	WORD	00 00	? (for English charset 00 00, for Chinese F7 A1)
0x0009	WORD	00 00	? (for English charset 00 00, for Chinese FE A1)
0x000B	DWORD	00 00 00 5F	number of characters in font file
0x000F	WORD	00 00	?
0x0011	WORD	00 15	font name length -1
0x0013	DWORD	00 00 05 F5	file size of font data
0x0017	DWORD	00 10 40 66	relative start position of font data

Examples

```
00 00 00 00 01 20 20 A1 F7 A1 FE 51 20 00 00 00 00 0F 00 90 28 10 00 00 00 00 00
00 00 00 00 00 10 20 00 00 00 00 5F 00 00 00 00 00 15 00 D6 17 00 00 90 28 10 00
00 00 00 00 00 08 10 00 00 00 00 5F 00 00 00 00 00 04 00 F5 05 00 00 66 40 10 00
```

Font data

block size = width of a character x (height of a character / 8)
size = block size x number of characters in font file
position in file = stored in the file header at 0x0014 + relative start position of font data at font property 0x0017
Font data is 1-bit per pixel image data. Font height is always multiples of 8 pixels (=1 byte).

Component attributes’ index data

block size = 6 bytes
size = block size x count stored in the file header at 0x0024
position in file = stored in the file header at 0x0038

Offset	Length	Example*	Description
0x0000	DWORD	00 00 00 23	relative start position in parameter data
0x0004	WORD	00 59	data size in bytes

Examples

```
00 00 00 00 03 00
03 00 00 00 20 00
23 00 00 00 59 00
7C 00 00 00 33 00
AF 00 00 00 4F 00
```

If you have any idea about the missing fileds (marked with a red question mark),
have any correction or a question please send an e-mail to uniko(at)vnet.hu

Thanks!

*Data is stored in little-endian, examples shown are big-endian