NAME

hp2xx – A HPGL converter into some vector- and raster formats

USAGE

hp2xx [-options] [hpgl-file(s)]

OPTION SUMMARY

Option Format Default   Description

General options:
```
 -c  char *  11111111  Pen color(s). Valid: 1 to 8 digits of 0-7 each.
                 0=off, 1=black, 2=red, 3=green, 4=blue, 5=cyan,
                 6=magenta, 7=yellow.
 -f  char *  ""        Name of output file. "" = autom., "-" = stdout
 -l  char *  ""        Name of optional log file
 -m  char *  pre       Mode. Valid (some are compile-time options):
                 mf,cad,dxf,em,emf,epic,eps,escp2,fig,jpg,gpt,hpgl,
                          rgip,pcl,pcx,pic,img,pbm,png,pre,svg,tiff,pdf,nc
 -p  char *  11111111  Pensize(s) [dots] (default), [1/10 mm] (mf, ps).
                 Valid: 1 to 8 digits of 0-9 (or characters A-Z for
                          widths beyond 0.9mm) each.
 -P  int    0:0      Selected page range (m:n) (0 = 0:0 = all pages)
 -q         off      Quiet mode. No diagnostic output.
 -r  double 0.0      Rotation angle [deg]. -r90 gives landscape
 -s  char *  hp2xx.swp Name of swap file
```

Size controls:
```
 -a  double 1.0      Aspect factor. Valid: > 0.0
 -h  double 200      Height [mm] of picture
 -w  double 200      Width  [mm] of picture
 -x  double  -     Preset xmin value of HPGL coordinate range
 -X  double  -      Preset xmax value of HPGL coordinate range
 -y  double  -     Preset ymin value of HPGL coordinate range
 -Y  double  -      Preset ymax value of HPGL coordinate range
 -z  double 1.0      Z engagement (working depth) (used in nc output only)
 -Z  double -1.0      Z retraction depth (used in nc output only)
 -t         off      True size mode. Inhibits effects of -a -h -w
```

HPGL handling controls:
```
 -n         off      No filling of polygons; draws outline instead
 -N         off      Ignore PS commands, calculate plot size as needed
 -e  int  0          Extend IW clipping limits by given amount
 -M  int     0       Remap pen no.0 commands to given pen
```

Raster format controls:
```
 -d  int    75       DPI value (x or both x&y)
 -D  int    75       DPI value (y only)
```

PCL only:
```
 -F         off      Send a FormFeed after graphics data
 -i         off      Pre-initialize printer
 -S  int    0       (Deskjet) Special commands: 0=off,1=b/w,3=CMY,4=CMYK
 -d  (see above)      Valid ONLY 300, 150, 100, 75
 -D  (see above)       INVALID for PCL!
```

EPS, PCL, and some previews:
```
 -o  double 0.0       x offset [mm] of picture (left  margin)
 -O  double 0.0       y offset [mm] of picture (upper margin)
```

    -C   Modify -o -O to center picture within -w -h frame

TIFF only:
  -S   int   0   Compression: 0/1=off,2=RLE,3=G3FAX,4=G4FAX,
                            6=OJPEG,7=JPEG,8=deflate

Preview on PC's (DOS):
  -V   int   18   VGA mode byte (decimal)

  -H   Show help.

DESCRIPTION

hp2xx reads HPGL ASCII source files, interprets them, and converts them into either another vector-oriented format or one of several rasterfile formats. Currently, its HPGL parser recognizes a large subset of HPGL/2. Some high-level functions related to polygon filling are missing. Also, only some of the fixed space vector fonts and none of the variable space arc fonts are supported. Beside these limitations, *hp2xx* has proven to work with many HP-GL sources without any trouble.

GENERAL OPTIONS

hp2xx reads from stdin or from a file if any given on the command line. If no output file name is given (default), the output automatically goes into a file whose name is derived from the input file name and the current mode. For example, *hp2xx -m pcl foo.hpgl* writes the output to a file "foo.pcl". Use option *-f outfile* to specify your output file name explicitly, or -I -f- to write to stdout, e.g. when piping into a queue.

The program scans the current HPGL source, converts all drawing commands into elementary vectors, saves these in a temporary file, and concurrently determines the maximum coordinate range used. It then processes the vectors by mapping them into a user-defined coordinate system, preserving the aspect ratio of the original data.

This coordinate system by default fits into a window of size 200 mm by 200 mm. To change the size of this bounding window, use *-h height* and *-w width* to set the (max.) desired height and width of your output picture; optionally use *-a aspectfactor* to alter the aspect ratio by the given factor (aspectfactor < 1 narrows your picture). The generated picture will always fit into the window defined by -h height and -w width, padded with background color at the lower or right margin if needed.

A second way of defining sizes is relying on the size the picture would actually show if plotted on a sheet of paper by a HP-compatible plotter. By activating flag -I -t (true size), options -a, -h, and -w are ignored, and the sizes are derived from the HP-GL file assuming that 1 HP unit = 1/40 mm.

Option *-r rotation_angle (in degrees)* allows you to rotate the object prior to all scaling operations. Its main use is to facilitate landscape format: -r90 rotates your whole picture, e.g. from portait to landscape format. However, any reasonable rotation angle is valid.

By naming a file with option *-l log_file* you can redirect the diagnostic outputs into the given file, even without a redirection mechanism for stderr like in UNIX shells (e. g., DOS). Option *-q* (quiet) gets rid of them completely.

If you need to process a series of similar objects which should be translated into exactly the same coordinate system, there is a way to override the auto-scaling: First, run all files separately and note the infos on the used coordinate ranges. Then, pick a range that will cover ALL your pictures. You can now assign defaults to the internally generated range limits by specifying *-x xmin, -X xmax, -y ymin,* and *-Y ymax.* NOTE: Clipping is only supported via the IW command ! If any picture coordinate exceeds your limits, they will be overwritten.

Use option *-m mode* to select the program mode, i.e. the output format. Currently supported: mode = "mf" (Metafont), "em" (emTeX cial{} commands), "epic" (line drawing using TeX macros within epic.sty), "eps" (PostScript), "dxf" (Autocad), "emf/emp" (MS Enhanced Metafile / Printing - available in Windows-built executables only), "svg" (Scalable Vector Graphics), "fig" (XFig 3.2), "gpt" (GnuPlot

ascii), "hpgl" (simplified HP-GL, e.g. for import tasks), "pcl" (HP-PCL Level 3 format (suitable for printing on a HP Laserjet II, DeskJet, or compatible printer), "escp2" (Epson Esc/P2 printer commands, suitable for printing on Epson Stylus models), "img" (GEMs IMG format), "jpg" (JPEG image), "pdf" (Adobe Portable Document format), "pbm" (Portable Bit Map / Portable PixMap for color plots), "pcx" (PC-Paintbrush format, also accepted by MS-Paintbrush / Windows 3.0 and many other PC based pixel renderers), "png" (Portable Network Graphics format), "nc" (CNC G-code, for engravings), or "rgip" (Uniplex RGIP). There is also a preview option "pre" which supports VGA cards (DOS), ATARI, AMIGA, X11 servers, and Sunview. Default mode is "pre". (As some of these modes rely on external libraries, they may not be builtin by default, and not be available in prebuilt binaries supplied e.g. in Linux distributions. The usage messsage generated when hp2xx is invoked without parameters will always list exactly those modes that are actually available.)

If you use a raster format, the picture is rasterized by default into a 75 DPI resolution image. Use option *-d DPI_value* to change the resolution, e.g. -d300 will cause a HP LJ-II compatible 300 dpi rasterization. There is a way of specifying a different resolution for y direction: *-D DPI_y_value*

Some programs were found to generate HPGL output with too tight clipping bounds, which lead, for example, to some parts of text characters clipped off. Use option *-e extraclip* to add some extra amount of space to clip areas to workaround such mistakes. For example, -e 40 will add 40 extra plotter units to every side of clipping box which is 1 mm in true size.

While processing large pictures at high resolution on low-memory machines, typically under DOS, the program may start swapping. Optionally change the swap file by using *-s swapfile,* e.g. to speed up processing by swapping to a RAM disk.

Unless the hpgl file specifies its own selection of pen widths and colors (for up to 256 pens), a carousel of 8 pens is simulated. You can specify pen sizes and colors for each of these pens via options *-p string* and *-c string.* "string" must consist of 1..8 digits (0-9 for size, 0-7 for color). Digit number n (counting from left) corresponds to pen number n. The digit value is this pen's color or size in internal units. The pen width unit corresponds to 1/10 mm - using pen widths beyond 0.9mm is possible by using the letters of the latin alphabet, so that A=1mm, B=1.1mm etc. The default size is 1 for all pens. Colors are assigned according to: 0=off, 1=black, 2=red, 3=green, 4=blue, 5=cyan, 6=magenta, 7=yellow. Examples of use: *-p22222222 -c33333333* changes all pensizes to 2 units, all colors to green *-p302 -c407* makes pen #1 a blue pen of size 3 , pen #3 a yellow pen of size 2, suppresses all drawing with pen #2, and keeps all other pen sizes and colors. Setting either -p or -c will override the equivalent HPGL/2 commands (PC,PW) in the HP-GL file.

Sometimes, HP-GL files contain several pages of plotter output. *hp2xx* recognizes the HP-GL commands for "feed-forward", "pause" or "new page", and by default draws each image as a separate page (saving to sequentially numbered output files, or opening a new preview window for each). You can select any particular page range by using option *-P firstpage:lastpage* which causes *hp2xx* to skip all drawing commands except those on the given pages. Please note that even if only a single page is actually drawn, *hp2xx* will nonetheless process the whole HP-GL file. This makes sure that effects of early pages on internal modes indeed influence later pages, as on a real plotter.

VECTOR FORMATS

Supported vector formats are: *TeX/Metafont, emTex-specials, TeX/epic-Macros, Autocad DXF CNC G-code XFig 3.2, GnuPlot ASCII, Simplified HP_GL, Uniplex RGIP Scalable Vector Graphics (SVG) Adobe PDF(if libpdf is available)* and -I PostScript. Use *-m mf* to convert a HPGL drawing into a Metafont character to be included into a TeX document as the character "Z" of a special font that you may create. Edit the metafont source, e.g., to change the letter "Z" for another, or to change the line thickness, which is set to 0.4pt by default. The other TeX-related modes ("cad" for TeXcad compatible code, "em" for employing cial{em:line} macros, and "epic" for drawing lines with macros from "epic.sty") address different compromises to cope with TeX's poor line drawing capability and are generally not recommended nor fully supported. Feel free to experiment with them -- they generate ASCII output that should be "input" into TeX/LaTeX documents.

Use option *-p pensize(s)* for control over pensize: The actual Metafont or PostScript pensize will be "pensize * 0.1 mm", with pensize = 0 - 9 (0 = no drawing). The same applies to

In PostScript mode *(-m eps),* you may also need to use options -o and -O (see below) for proper margins on paper since *hp2xx* puts your picture "flush" to the left and upper paper limit by default.

RASTER FORMATS

The following formats are supported: HP-PCL, Esc/P2, PCX, PIC, IMG, JPG, PBM/PPM, PNG, TIFF, and previews. (PNG and TIFF formats rely on external libpng,zlib and libtiff, JPG relies on libjpeg. Versions built on MS windows systems - or versions linked against libEMF on other platforms - may additionally support EMF generation and printing.)

Addition of other formats is made easy for programmers because of *hp2xx's* modular structure. The program allocates a bitmap on a line-by-line basis, swapping lines to disk if needed, and plots into this bitmap. Depending on the selected format, a conversion module is then activated, which can easily be replaced by other converters. Add more formats if you like!

Option *-p pensize(s)* controls the size (in pixels) of the virtual plotting pen. The only implemented shape of the pen tip is a square of the given length. pen sizes of 5...9 units will be acccepted but replaced by 4 units. Specifying -p4 when in 75 DPI mode will make pretty clumsy pictures, while you may prefer -p2 over -p1 when in 300 DPI.

PCX: The size of a PCX picture is controlled via its specified height and the current DPI value. To create a high-resolution PCX image, just increase the DPI value as desired. PCX format does not accept offsets.

IMG: See PCX.

PBM/PPM: See PCX for options. If your hpgl file is not monochrome, hp2xx will automatically create a PPM (portable pixmap) file instead of a PBM bitmap. (Use -c11111111 to force generation of PBM from a color hpgl file). Depending on the compile-time option PBM_ASCII, hp2xx will create ascii or binary pbm (ppm) files - usually the more efficient binary format should be preferred.

(Unsupported options) PIC, PAC: ATARI ST screens (640x400 pixels) can easily be dumped to files. Programs such as STAD accept graphics by including such screen dump files. Graphics filling more than one screenful may be split into screen-size blocks and loaded/mounted blockwise. *hp2xx* converts to ATARI bitmap format by trying to fit the resulting picture into a single screen equivalent (max. 400 rows, max. 80 Bytes (640 pixel) per row). If it succeeds, *hp2xx* produces a single output file. Specify ONLY its base name (option -f), since *hp2xx* adds the file extension ".pic" or ".pac" automatically. Do NOT try to work on more than one HPGL file simultaneously! Do NOT use more than 6 characters for the file name, and avoid digits. If more screen blocks are required horizontally and/or vertically, *hp2xx* will automatically split the picture into separate files, counting them columnwise (top-to-bottom and left-to-right), adding a two-digit number to the given file name. A maximum of 10 columns is supported. The picture is padded with background color at its right and lower margins, if needed. PAC features file compression, PIC does not.

PCL: HP-PCL Level 3 format, most useful for direct printer output. Due to this action, there have been added some extra flags and options: Use flag *-i* to send a printer initialization sequence before the actual image. Among other things, this will instruct the printer which paper size to use. Flag *-F* adds a Form Feed (FF, hex 0C) after the image is completed, which is what you may want most of the time. However, overlay printing of several files is feasible by omitting -F.

For additional control of the picture's final position on paper, you may add x or y offsets using *-o X_offset* or *-O Y_offset. E.g., -o 20 -O 30 will give you 30 mm additional top margin* and 20 mm additional left margin. Option -C modifies these offsets to center the picture within the frame defined by -w -h.

The option -C will attempt to center the drawing on the paper automatically. Note also that *hp2xx* now honors any PS (page size) commands in the hpgl file, which can also create white space around the actual drawing.

The option *-N* will make hp2xx ignore any PS commands given in the hpgl file, and recalculate the

image size based on the actual geometry instead.

The option *-n* will make hp2xx ignore any polygon filling commands, rendering only their outlines. This may serve both as a work-around for hp2xx' limited polyfill support, and improve clarity of thumbnail images of PCB designs and the like.

The option *-M pennumber* will remap any color or drawing commands from pen 0 to the specified pen (which should typically be otherwise unused in the drawing). Historically, selecting pen 0 instructed a pen plotter to put away the pen and stop drawing, while modern inkjet plotters can use it like any other color. Due to this ambiguity, hp2xx will draw the background of raster graphics in the pen 0 color, unless this option is used.

For DeskJet / DeskJet Plus / DeskJet 500 / Deskjet 550 printers, there are some special printer commands. Activate them with option *-S n.* n=0 switches them off, n=1 activates black/white mode, n=3 (DJ500C and DJ550 only) supports CMY color data, n=4 (DJ550C only) supports CMYK color data. Any n!=0 activates PLC data compression (TIFF mode: 2).

Esc/P2: This is the control language used in the Epson Stylus family of inkjets. *hp2xx* currently does not address more than one line of nozzles in the print head, so printing, while exact, is extremely slow. Users might prefer piping the output of the PostScript module through *ghostscript* until this issue is resolved.

PNG: Support for the Portable Network Graphics format relies on *libpng* which is available from www.libpng.org.

PRE: Preview on all machines. Use options -h -w -o -O -C to define the screen size and position of your output (-o -O -C may not always apply). Under X11, you can pan around an image that is larger than the screen size by 'dragging' it with the mouse (pressing button 1 while moving the mouse in the desired direction). Any other mouse button or keyboard key will terminate the preview. For VGA cards (DOS), option *-V VGAmode* gives you a simple way to utilize SVGA modes. Please take care not to define larger windows than your graphics device can handle, as the results are unpredictable. As hp2xx uses standard BIOS calls to set pixels on VGA cards (slow but portable), you can select any hi-res mode supported by your system by simply specifying the mode byte with this option.

TIFF: The tagged image file format is supported by most graphics and image manipulation programs. Support for TIFF in *hp2xx* relies on the *TIFF library* available from www.libtiff.org, which offers several means of image compression. The *-S* commandline option selects between them as follows: *-S 0 or -S 1: no compression -S 2: RLE (run length encoding) -S 3: Group 3 FAX (monochrome) -S 4: Group 4 FAX (monochrome) -S 5: GIF (not available by default, because of the UNISYS patent) -S 6: JPEG ('old' TIFF 6.0 style) -S 7: JPEG -S 8: deflate*

EXAMPLES

        % hp2xx -m pcx -f my_output.pcx -d300 -p2222 -h50 -a 1.2 my_input.hp

        creates a PCX file at 300 DPI of height 50 mm, using an aspect factor
        of 1.2 and a pen size of 2 pixels for pens 1-4.

        % my_hpgl_generator | hp2xx -f- -o20 -O30 -F -q | lpr -P my_PCL_printer

        HPGL output is piped through hp2xx; the resulting PCL code is piped to
        the printer queue, giving an image of height 100 mm at 75 DPI.
        An additional left margin of 20mm and upper margin of 30mm is created.
        A formfeed will be added (handy if your printer queue does not).

        % hp2xx my_input.hp

        Preview on screen or into window.

ORIGINAL AUTHOR

Heinz W. Werntges, Physikal. Biologie, Geb. 26.12,
Heinrich-Heine-Universitaet,
D-40225 Duesseldorf, Germany.

MAINTAINER SINCE V 3.30

Martin Kroeker, daVeg GmbH,
Schottener Weg 2
D-64289 Darmstadt, Germany.
mk@daveg.com or martin@ruby.chemie.uni-freiburg.de

ATARI features & PIC, PAC, IMG modes are due to Norbert Meyer, Duesseldorf.
AMIGA version & PBM mode are due to Claus Langhans, Kelkheim (Ts.)
X11 previewer is due to Michael Schoene, Duesseldorf.
Thanks for VAX support and a lot of testing to
  Michael Schmitz & Gerhard Steger, Duesseldorf
Many OS/2 helps were due to Host Szillat, Berlin.
(Later contributors: See TEXINFO file).

DIAGNOSTICS

The number of ignored and/or unknown HPGL commands is given. You will be informed if swapping
starts. Progress is indicated by a logarithmic count of internal vectors during scanning and plotting, or
by dots during (raster mode) output, where each dot corresponds to 10 scan lines.

BUGS

There still are many non-implemented HPGL commands.

The color assignment of some X11 servers leaves something to be desired.

Color is only partially supported (not all possible formats).

VGA preview: Color "magenta" shows as brown on some VGA cards.

To match the specified sizes on your display during preview, you may have to calibrate it using -d -D,
e.g. by overwriting the 75 DPI default.

Only little testing has been done on TeX-related and ATARI formats, so be prepared for bugs there, and
PLEASE report them -- thank you!

SEE ALSO

bm2font(1), F. Sowa's raster-to-TeXfont converter.