The HP-GL Story told by Tom Tremble, HP San Diego Division

As far as I remember, the original HP-GL language was developed by NORM JOHNSON and (I think) DALE SHAPPER, while working on the 9872A, a four pen A3 vector plotter intended for use with Loveland's desktop programmable calculators (9825A, etc).

The story is said that some sort of mnemonic command language using alphabetic characters for the commands, and integer numeric (digits 0 to 9 and the minus sign) parameters as the data was desired. Only common punctuation (periods, commas, and colons, and semi-colons were allowed as separators, etc. This structure could be handled easily by any computing device and interface method whereas the binary format used by other devices of the time was difficult to impossible on some systems. Two characters were chosen for the commands because "one character didn't allow for enough commands, and three characters was too much to deal with".

Over the years, HP-GL was extended to HP-GL/2 (this when large format devices exceeded the dimensions that a 16 bit integer could specify using plotter units of 1016 "plotter units" to the inch, 40 per millimeter) and to add more functionality. When the LaserJet III was in development, the Boise division decided that vector plotting capability was needed in addition to the PCL language. The San Diego R&D Language team decided that if the HP-GL language was to be "owned" by San Diego in the future (in stead of by Boise), we had best take charge. Hence, a team was put together to create "a re-usable HP-GL/2 language definition and source code". There is a core set of commands which all devices must have, and several types of extensions groups which are used by devices based on their capabilities and purpose -- i.e., the color group, technical extensions, etc. This project was called "Starship" and resulted in the "Starship Code" which is used today in most higher end LaserJet's, DeskJet 1600, all DesignJets, and perhaps a few other products, be they raster or vector devices (all are raster print engines today). The definition has stayed essentially fixed since then.

HP-GL

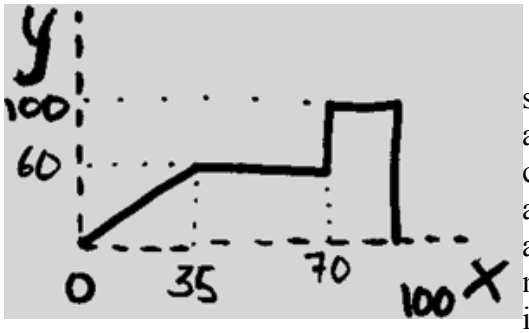
The HP-GL (Graphics Language) originated as a teamwork in the San Diego Division of Hewlett-Packard in the early 1970's. A veteran from those early days, Tom Tremble, has written a few words for us about ***The story of HP-GL***. And a programming language it is, having become an industry standard. HP-GL was the original language, meant to support HP's first series of X/Y-plotters (The 9872 family), being a flatbed machine with a moving arm and on this arm a moving pen holder. With wires wound up around two motors and pens located in slots along e front, the gthraphics presented was of really

high quality, being the first high-level language plotter on the market. The plotter was delivered with both HP-IB (IEEE488 8-bit bus standard), and RS232 (model 7221B).

As time went on, HP delivered a huge numbers of different series, with different models within each series. The original firmware was built around 16-bit addressing, limiting the physical size to A0 (E) using a +- scheme for 16 bit integer data. (The resolution was 40 units/mm). Internally the electronics parces the incoming ASCII data and translates the two-letter commands into motor movements in the X and Y-directions, while also taking care of the pen selection and movement up and down. The HP-GL example gives you a sample of HP-GL programming.

Finally it was decided to change the core of HP-GL in order to allow for longer plots and also looking into the field of graphical plotting. The firmware code was completely re-written, using 32bit addressing and thus allowing for much longer plots. And in order to avoid special versions for every plotter family the commands were structured differently. This new version was called HP-GL/2.

HP-GL Example



Here is an example of a simple drawing, some outline of something we want to have as a series of lines on our paper. In user coordinates the x varies from 0-->100, and y also 0-->100. All of HP's HP-GL plotters have a hardware resolution of 40 increments per millimeter, which constitutes the absolute increments. A reletive movement of 8000 plotter units means a position change of 8000/40mm=200 mm.

In our little example, we want these lines to walk over the whole area of the paper we load into our unit. Regardless of the size, and the number of plotter units we use, we must map our xy-values into whatever paper we have using a SC (scale) command. This command locates our range between the outer limits we are able to use, P1 (lower left) and P2 (upper right). In this simplified example the chain of commands will thus become:

IN;	INitialize the printer, resetting all HP-GL parametres to preset values. The ";" serves as a delimiter in the original HP-GL, separating the commands. A separator character is not used in HP-GL/2.
SC0,100,0,100;	SCale the x-values (mapping them) such that the complete x-direction width is now having the value range X_{min}=0 and X_{max}=100. The same will be valid for the Y-axis.
SP1;	Select Pen number 1. In the "old" days when pens were pens in a carousel, the plotter could rotate the carousel and pick one of the pens. In this case the pen in position 1. (Whatever color and width it might have).
PU;	Pen Up, lift the "pen" up, no line is drawn on the media.
PA0,0;	Plot Absolute(with the pen up) from wherever you were to the position 0,0 (x,y).
PD;	Pen Down, lover the pens to hit the media
PA35,60,70,60, 70,100,100,100,100,0;	Move now the penn through the different points, like (35,60), (70,60), (70,100), (100,100), (100,0)
PU;	Lift the pen
SP0;	Select Pen 0, Meaning "no pen", to be sure nothing is plotted. In the old fashion pen plotters the pen previously used will be returned to its position in the pen carriage.
NR; or PG;	Not Ready will for a pen plotter with a sheet of paper loaded put the paper forward, ready to be taken out. PG; (PaGe) is the most important command for all of HP's HP-GL or HP-GL/2 units. This command terminates the command sequences given to the plotter. And we also have to add the semicolon (;) as the last character, even in HP-GL/2. "The Plot Is Over", spit out the paper and be ready for the next plot. In the diagnostic section of this CD you will find a simple sequence for unit testing from DOS.

Finally, just to inform you: All of our large plotters and printers have the X-direction vertical. This is due to the fact that the longest axis (X-axis) runs along the media path. And we just defined it like this!