Controlling a Graphics Plotter with a Handheld Programmable Calculator

by Robert M. Miller and Randy A. Coverstone

THE ADVENT OF THE HP-IL (Hewlett-Packard Interface Loop)* transformed the HP-41C from a programmable calculator into a true computer system with mass memory, a variety of printers, and a video display. However, one capability still lacking was some form of graphics output. The HP 7470A Graphics Plotter with an HP-IL interface and the HP 82184A Plotter Module (Fig. 1) was developed to satisfy that need.

As a computer, the HP-41C possesses a powerful and flexible instruction set, but is limited in speed and memory. While it is possible to connect and control a plotter using only the control functions built into the interface module, the resulting graphics programs would consume most of the available RAM and execution would be painfully slow. The 82184A Plotter Module was designed to free the user from these limitations. By providing a high-level set of graphic commands in external ROM, the HP-41C's system RAM is entirely available for the user's application programs. Further, all of the graphics primitives are written in machine language to minimize execution time.

The HP-41C with the 82184A Plotter Module, the 82160A HP-IL module, and the 7470A Graphics Plotter make up a low-cost, yet powerful graphics solution. The Plotter Module allows HP-41C users to produce bar and line charts as well as function and point plots on paper and transparencies. It also provides the capability to produce bar code of HP-41C programs and data on either the 7470A or the HP 82162A Thermal Printer. The bar code can then be read back into the HP-41C via the HP 83153A Optical Wand.[3]

*HP-IL is an interface system for low-power, portable systems introduced late in 1981. HP-IL will be discussed in next month's issue. Also see references 1 and 2.

Command Set

The 82184A Plotter Module is an 8K-word unit (word length is 10 bits) which adds 52 microcoded commands to the HP-41C's function set, along with a sophisticated interactive plotting program written in RPN (reverse Polish notation). Many of the command names and functions are based on the graphic command sets of the HP 9845 Computer and the HP-85 Personal Computer. This allows users familiar with these products to master the plotter module's commands quickly.

The module's functions are divided into three categories:

- Plotting primitives. These commands set up the plotter, scale the plotting area to any convenient user units, provide windowing, move the pen from one location to another, change pens, raise and lower the pen, draw and

Fig. 1. *The HP 82184A Plotter Module and HP 82160A HP-IL Module are easily inserted into two of the four I/O ports at the top of the HP-41C Programmable Calculator, providing low-cost graphic output to an HP 7470A Graphics Plotter.*

label axes, and digitize points.
- Bar-code primitives. This category contains commands that create the bar-code bit patterns for user programs and data (numeric, sequential, alpha, and alpha append), as well as commands to plot bar code.
- Interactive plotting program. This program prompts the user for the necessary data and then creates a complete plot. It has a wide range of options and is easily extended by user-contributed subroutines. The program provides data to three subroutines that initialize the plotting area, plot the function or data, and annotate the plot. These three subroutines are also available for use in other user programs.

Plotting on the 7470A

Like other HP plotters, the 7470A's platen is divided into addressable units, called absolute plotter units (APUs). On the 7470A there are 40 APUs to a millimeter. To move the pen to a new location on the platen, a plot command is sent to the plotter followed by X and Y coordinates in APUs. However, for the vast majority of plotting uses, these default units are inappropriate. Therefore it is useful to be able to superimpose another scale on top of the APUs.

Executing the plotter module's PINIT command maps a default scale, called the graphic units (GU) scale, onto the area specified by the plotter's lower left and upper right reference points (P1 and P2, respectively). On the 7470A the GU scale provides 0 to 100 units in the Y direction and 0 to 138.9 units in the X direction, using the default settings of P1 and P2.

In addition to scaling the entire platen, it is convenient to be able to specify a portion of the platen as the active plotting area and then superimpose a new scale on this area. Then, after a function has been plotted (Fig. 2), annotation can be done outside of this area so as not to obscure the data. The plotter module's LOCATE statement allows the user to specify any desired subset of the plotter limits, and the SCALE command maps any desired user units (UU) to the plotting area.

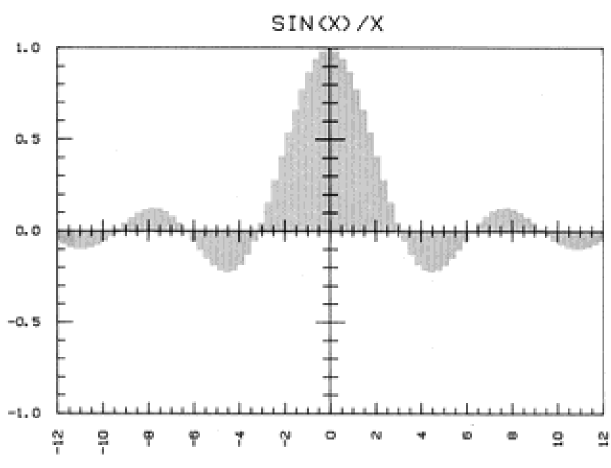


Fig. 2. *Example of a function plotted by an HP-41C, 82184A Plotter Module, 82160A HP-IL Module, and a 7470A Graphics Plotter.*

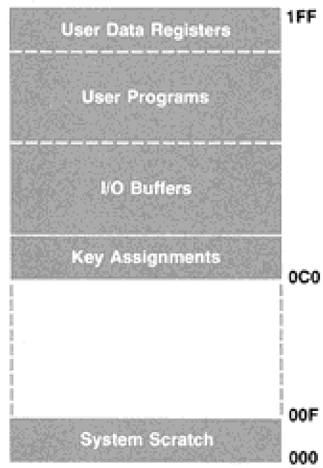


Fig. 3. *Memory map of HP-41C system RAM.*

I/O Buffer

To perform these functions, the plotter module must keep all scale factors, as well as P1, P2, and the endpoints of the area specified by LOCATE in memory. Additional space is needed for status information and temporary storage of bar-code geometry parameters. The module stores all of this information in an I/O buffer.

I/O buffers are created in the HP-41C's memory above any key assignments (see Fig. 3). For the plotter module, a 26-register I/O buffer is needed. The I/O buffer has a header register which tells the operating system and any module scanning memory that an I/O buffer has been found, to which module the buffer belongs, and how many registers there are in the buffer. The drawback to using the I/O buffer structure is that its location in the HP-41C's memory is not fixed. It may be shifted up or down depending on the number of key assignments and the presence or absence of other I/O buffers. Thus, each command must do a memory search to find the I/O buffer before the data it holds can be used. The advantages of the I/O buffer are that data is protected from inadvertent modification, and that its use is totally transparent to the user.

Creating Bar Code

The HP-41C uses a two-level bar code,[4] meaning it is composed of two different bar widths. Narrow bars represent 0 and wide bars represent 1. Spaces between bars serve only as delimiters and carry no information.

Two powerful user-language programs are provided in the plotter module's manual to facilitate producing bar code. The first is a collection of subroutines that labels and plots one row of program or data bar code. The second, PLOTBC, is interactive and provides a quick and easy method for the novice user to produce bar code of programs and data. An example of the output of this program appears in Fig. 4.

Developing a set of plotting parameters that would produce bar code of optimal geometry proved to be challenging. Variations between pens and within the lifetime of a pen result in slightly different ink flow rates and line widths. This difference is not significant for most plotting

SINX/X

ROW 1: LINES 1-4

ROW 2: LINES 5-11

Fig. 4. *Bar-code functions and powerful user language programs given in the 82184A Plotter Module's user manual make it easy to obtain plots of bar-code programs as shown.*

uses, but it is enough to slightly alter the geometry of the bar code produced. Thus, parameters set to produce suitable bar code with a new pen may later produce unreadable bar code as the pen begins to wear. Plotting discrete bars would have required the pen to be dropped as many as 1584 times for a single page of bar code. Pen wear caused by repeated nib impact against the paper is minimized by connecting all of the bars in a row at the top (Fig. 4), so that the pen is only lowered once at the start of a row of bar code. This also speeds bar-code production by eliminating the time delays associated with raising and lowering the pen.

Default parameters were chosen to produce the most consistently readable bar code for the widest range of pen nib conditions and ink flow. A 0.3-mm pen is assumed for these parameters and good results were obtained with both fiber tip and transparency pens. Recognizing that users may choose to use different pens and that the pen nib width changes with use, a command is provided to alter the bar code parameters. BCSIZE allows users to specify (in APUs) any or all of the bar-code parameters.

Creating bar code with the plotter module is a two-step process. Data is entered into either the X register or the ALPHA register of the HP-41C and then the appropriate bar-code function is executed. This creates the bit pattern of the desired bar code and places it in the ALPHA register. A second command, BC, must then be executed to plot the bar code on the 7470A.

Interactive Plotting Program

NEWPLOT is an interactive plotting program that can plot functions or any arbitrary set of points with minimum overhead. The program prompts the user for 1) the name of a user program that, given an X coordinate, will compute the Y coordinate, 2) scaling information, and 3) either the number of points to be plotted or the increment between points to be plotted. At that point the user can examine and edit any or all of the data base. The user can specify the placement of the X and Y axes, the number of major and minor tics per axis, and the number of labels per axis. Data buffers can be created, edited, and plotted. Other built-in features allow the user to alter the line type and pen used for plotting, and to do scatter plots. It is also possible to change one or more parameters of the plot and then to redo the complete plot with one keystroke.

Acknowledgments

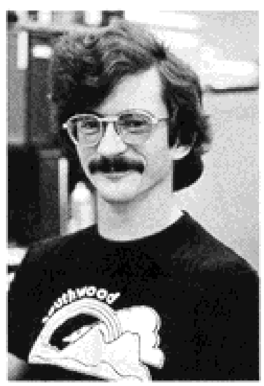
Thaddeus Konar was primarily responsible for the bar-code plotting commands. Dave Conklin provided ideas and guidance throughout the project.

References

1. S. Harper, C. Landsness, and R. Quick, "Interface System Weds Instruments to Small Computers," Electronic Design, December 24, 1981, pp. 78-93.
2. R. Katz, "The Hewlett-Packard Interface Loop—HPIL," Byte, April 1982, pp. 76-92.
3. J.J. Uebbing, D.L. Lubin, and E.G. Weaver, Jr., "Handheld Scanner Makes Reading Bar Codes Easy and Inexpensive," Hewlett-Packard Journal, Vol. 32, no. 1, January 1981.
4. D.R. Conklin and T.L. Revere III, "Reading Bar Codes for the HP-41C Programmable Calculator," Hewlett-Packard Journal, Vol. 32, no. 1, January 1981.

Robert M. Miller

Bob Miller joined HP after receiving a BS degree in computer science at California State Polytechnic University in late 1980. He also has a BA degree in English literature awarded in 1973 by LaSalle College, Pennsylvania. The HP-41C Plotter Module was his first HP project and now he is a project leader for HP-75C application software. Bob was born in Philadelphia, Pennsylvania and now lives in Corvallis, Oregon. His interests include backpacking, reading, swimming, and playing volleyball.



Randy A. Coverstone

Randy Coverstone was born in Goshen, Indiana, and attended the University of Evansville, Indiana, receiving a BSME degree in 1975. After earning an MSME degree and the degree of Mechanical Engineer at the Massachusetts Institute of Technology in 1978, he joined HP. Randy worked on the chart advance for the 9872 Plotter, which he discussed in an earlier HP Journal article, and the servo design of the 7470A. He is a visiting lecturer on applied controls at the University of California at San Diego. He is married, has a new baby daughter, and lives in San Diego, California. His outside interests include designing recumbent bicycles and playing works by Bach and Scott Joplin on the piano.