NAME

plot_cntl() – control of built-in plotting

DESCRIPTION

`plot_cntl(`*`s`*`)` – Selects built-in plotting features. The argument *s* is a string of comma- or space-delimited options. The following options may be preceded by a minus sign to turn the associated feature off, nothing (or an optional plus sign) to turn the feature on or a question mark to return a value of one or a zero that indicates whether the associated feature is currently on or off. For example,

```
plot_cntl("xlog,ylog")    # turns x and y log-axis mode on
plot_cntl("+xlog")        # turns x log-axis mode on
plot_cntl("-xlog")        # turns x log-axis mode off
print plot_cntl("?xlog")  # returns 1 if mode on, else 0
```

`"xlog"` – use a logarithmic *x* axis.
`"ylog"` – use a logarithmic *y* axis.
`"xexact"` – force *x*-axis minimum and maximum to be set to the scan endpoints (as opposed to being rounded).
`"colors"` – enable the use of colors.
`"dots"` – draw graphics-mode points with large dots.
`"lines"` – connect graphics-mode points with lines.
`"ebars"` – draw vertical lines through each point of length equal to the twice the square root of the *y* value.
`"persist"` – keep graphics-mode on after ordinary user input. `spec` errors, a `^C` from the keyboard or `plot_cntl("close")` restore text mode.
`"perpetual"` keep graphics-mode on continuously (appropriate if using X windows, for example).

Other options are:

`"font=`*`string`*`"` – sets the font used for the *x11* high-resolution plot windows. If the font is a scalable font, its size will be chosen to match the current size of the plot window. Otherwise, the font size will be fixed to either the value specified in the font name using the *X Logical Font Description Conventions* XLFD, or if the size fields are wild carded, the font size will be selected by the X server.
`"title=`*`string`*`"` – sets the title-bar text for *x11* high-resolution plot windows. In a comma-separated list of arguments to `plot_cntl()`, this one must appear last.
`"geometry=`*`string`*`"` – sets the size and position for *x11* high-resolution plot windows. The format of the geometry string is the same as expected by standard X windows applications, as in

```
plot_cntl("geometry=512x380-0+10")
```

As with the conventional syntax, not all parts of the geometry string are required.
`"colors=`*`bgnd`*`:`*`win`*`:`*`text`*`:`*`axis`*`:`*`symb`*`:..."` – assign colors for drawing the various graphics-mode elements. The values for *bgnd* (the background color of the area outside the axis), *win* (the background color of the area inside the axis), *text* (the color of the text), *axis* (the color of the axis) and *symb* ... (the color of the plotting symbols) are integers. The first 10 colors are standardized according to the following table:

0 – background (normally white or black)
1 – foreground (normally black or white)
2 – blue 3 – red
4 – green 5 – yellow

6 – cyan 7 – magenta
8 – white 9 – black

Other colors may be available depending on the particular device. You don't have to assign values to all colors.

`"colors[`*numb*`]"` – Returns the current color assignments, where *numb* is a number between zero and 67. Numbers zero through three return the colors assigned to the *bgnd*, *win*, *text* and *axis* elements respectively. Numbers from 4 through 67 return the colors assigned to the plotting symbols.

`"filter1"` – select filter 1 (the default). All plotting commands are directed to this filter.

`"filter2"`, `"filter3"`, `"filter4"`, `"filter5"` – select filters 2 through 5. Subsequent plotting commands are directed to the selected filter.

`"open"` – turn on graphics mode. If there is no graphics filter program currently active for the current filter number (see above), the filter program associated with the current `GTERM` variable is started. Recognized `GTERM` values are currently `vga`, `ega`, `cga`, `herc`, `x11`, and `sun`.

When using X11 graphics, you can assign a value to the built-in variable `DISPLAY` before a call of `plot_cntl("open")` to set the X display on which the plot will be drawn. If the filter is currently open, and you wish to change the display, use `plot_cntl("kill")` first. For example,

```
DISPLAY="host1:0"; plot_cntl("filter1,open")
DISPLAY="host2:0"; plot_cntl("filter2,open")
```

will create plot windows on two different hosts.

`"close"` – turn off graphics mode, unless the perpetual or persistent mode has been selected.

`"kill"` – turn off graphics mode and terminate graphics process.

`"erase"` – clear the graphics-mode screen (or the text screen if graphics mode is off).

`"addpoint"` – before a call to `array_plot()` or `data_plot()` will cause the plot to be made with minimal redrawing. Used for updated plotting during scans.

`"addline"` – before a call to `array_plot()` or `data_plot()` will prevent the current data from being erased and the new data from changing the ranges when the new data points are drawn. Used for plotting several data sets from different data groups or array elements on top of each other.

`"mca"` – before a call to `array_plot()` or `data_plot()` will cause the data points to be displayed using a minimal redrawing algorithm, appropriate for displaying data actively being accumulated by an MCA-type device. The `"dots"` and `"ebars"` modes must be turned off.

`"lp"` – before a call to `array_plot()` or `data_plot()` will generate printing instructions appropriate for plotting on a 132-column printer.