1 Printing in VISE Laboratory

For information specific to printing in the Purdue VISE laboratory, please refer to this link.

2 Printing Matlab Figures

The simplest way to print Matlab figures is usually through the menu in the figure window by following "`File->Print...`".

Matlab also has a built-in *print* command that allows you to print from the Matlab prompt. This is sometimes useful in a script if you want to automatically print your figures to files (e.g. pdf or postscript). The *print* command by itself will print the currently active figure window to the default printer. The "active" Matlab figure is usually the last figure window that was brought to the forefront. You can also choose the active figure window using the command

```
figure(n)
```

where `n` is the desired figure number.

The default printer is defined by your operating system (not within Matlab), and how to define this depends on your platform. In Matlab, you can specify the printer to use for each print job with the `-P` option.

```
print -Pprinter_name
```

You can print to file by specifying one of several options and the filename. For example, the following will print to a pdf file.

```
print -dpdf filename.pdf
```

There are many other output options available (e.g -dps, -djpeg, -dtiff). See Matlab's help for *print* for a listing.

If you use the subplot command to produce vertically stacked plots, you will often get the best looking output if you specify "tall" mode before printing the figure. This will stretch the subplots out to use the full height of the paper. To do this, select the active figure then issue the following command

```
orient('tall')
```

directly before you print. You can also obtain print output in landscape mode using

```
orient('landscape')
```

Questions or comments concerning this laboratory should be directed to Prof. Charles A. Bouman, School of Electrical and Computer Engineering, Purdue University, West Lafayette IN 47907; (765) 494-0340; bouman@ecn.purdue.edu

3 Printing Simulink System Windows

To print a Simulink system window, type the following command from the Matlab prompt

```
print -sSimulink_window -Pprinter_name
```

where `Simulink_window` is the name displayed along the top of the Simulink system window, and `printer_name` is the destination printer.

4 Printing Simulink Figures

Some Simulink simulations produce output in standard Matlab figure windows. These can be printed as any other Matlab figure, as described in Section 2 above. Make sure the simulation is completed or terminated before printing the output.

5 Matlab help on print

```
 PRINT Print figure or model. Save to disk as image or M-file.
    SYNTAX:
      print
        PRINT alone sends the current figure to your current printer.
        The size and position of the printed output depends on the figure's
        PaperPosition[mode] properties and your default print command
        as specified in your PRINTOPT.M file.

      print -s
        Same as above but prints the current Simulink model.

      print -device -options
        You can optionally specify a print device (i.e., an output format such
        as tiff or PostScript or a print driver that controls what is sent to
        your printer) and options that control various characteristics  of the
        printed file (i.e., the resolution, the figure to print
        etc.). Available devices and options are described below.

      print -device -options filename
        If you specify a filename, MATLAB directs output to a file instead of
        a printer. PRINT adds the appropriate file extension if you do not
        specify one.

      print( ... )
        Same as above but this calls PRINT as a MATLAB function instead of
        a MATLAB command. The difference is only in the parenthesized argument
        list. It allows the passing of variables for any of the input
        arguments and is especially useful for passing the handles
        of figures and/or models to print and filenames.

      Note: PRINT will produce a warning when printing a figure with a
      ResizeFcn.  To avoid the warning, set the PaperPositionMode to ’auto’
      or match figure screen size in the PageSetup dialog.

    BATCH PROCESSING:
        You can use the function form of PRINT, which is useful for batch
        printing. For example, you can use a for loop to create different
        graphs and print a series of files whose names are stored in an array:

        for i=1:length(fnames)
            print(’-dpsc’,’-r200’,fnames(i))
        end

    SPECIFYING THE WINDOW TO PRINT
        -f<handle>   % Handle Graphics handle of figure to print
        -s<name>     % Name of an open Simulink model to print
        h            % Figure or model handle when using function form of PRINT

      Examples:
        print -f2    % Both commands print Figure 2 using the default driver
        print( 2 )   % and operating system command specified in PRINTOPT.

        print -svdp  % prints the open Simulink model named vdp

    SPECIFYING THE OUTPUT FILE:
        <filename>   % String on the command line
        ’<filename>’ % String passed in when using function form of PRINT

      Examples:
        print -dps foo
        fn = ’foo’; print( gcf, ’-dps’, fn )
        Both save the current figure to a file named ’foo.ps’ in the current
        working directory. This file can now be printed to a
        PostScript-compatible printer.

    COMMON DEVICE DRIVERS
        Output format is specified by the device driver input argument. This
        argument always starts with ’-d’ and falls into one of several
        categories:
      Microsoft Windows system device driver options:
        -dwin      % Send figure to current printer in monochrome
        -dwinc     % Send figure to current printer in color
        -dmeta     % Send figure to clipboard (or file) in Metafile format
```

```
      -dbitmap   % Send figure to clipboard (or file) in bitmap format
      -dsetup    % Bring up Print Setup dialog box, but do not print
      -v         % Verbose mode, bring up the Print dialog box
                   which is normally suppressed.

    Built-in MATLAB Drivers:
      -dps       % PostScript for black and white printers
      -dpsc      % PostScript for color printers
      -dps2      % Level 2 PostScript for black and white printers
      -dpsc2     % Level 2 PostScript for color printers

      -deps      % Encapsulated PostScript
      -depsc     % Encapsulated Color PostScript
      -deps2     % Encapsulated Level 2 PostScript
      -depsc2    % Encapsulated Level 2 Color PostScript

      -dhpgl     % HPGL compatible with Hewlett-Packard 7475A plotter
      -dill      % Adobe Illustrator 88 compatible illustration file
      -djpeg<nn> % JPEG image, quality level of nn (figures only)
                   E.g., -djpeg90 gives a quality level of 90.
                   Quality level defaults to 75 if nn is omitted.
      -dtiff     % TIFF with packbits (lossless run-length encoding)
                   compression (figures only)
      -dtiffnocompression % TIFF without compression (figures only)
      -dpng      % Portable Network Graphic 24-bit truecolor image
                   (figures only)

    Other output formats are possible by using the GhostScript application
    supplied with MATLAB. For a full listing see the online help
    for GHOSTSCRIPT, use the command 'help private/ghostscript'.
    An example of some of the device drivers supported via GhostScript are:
      -dljet2p   % HP LaserJet IIP
      -dljet3    % HP LaserJet III
      -ddeskjet  % HP DeskJet and DeskJet Plus
      -dcdj550   % HP Deskjet 550C (UNIX only)
      -dpaintjet % HP PaintJet color printer
      -dpcx24b   % 24-bit color PCX file format, 3 8-bit planes
      -dppm      % Portable Pixmap (plain format)

    Examples:
      print -dwinc  % Prints current Figure to current printer in color
      print( h, '-djpeg', 'foo') % Prints Figure/model h to foo.jpg

  PRINTING OPTIONS
    Options only for use with PostScript and GhostScript drivers:
```

```
    -loose     % Use Figure's PaperPosition as PostScript BoundingBox
    -append    % Append, not overwrite, the graph to PostScript file
    -tiff      % Add TIFF preview, EPS files only (implies -loose)
    -cmyk      % Use CMYK colors instead of RGB
    -adobecset % Use Adobe PostScript standard character set encoding

  Options for PostScript, GhostScript, Tiff, Jpeg, and Metafile:
    -r<number> % Dots-per-inch resolution. Defaults to 90 for Simulink,
                 150 for figures in image formats and when
                 printing in Z-buffer or OpenGL mode,  screen
                 resolution for Metafiles and 864 otherwise.
                 Use -r0 to specify screen resolution.
  Example:
    print -depsc -tiff -r300 matilda
    Saves current figure at 300 dpi in color EPS to matilda.eps
    with a TIFF preview (at 72 dpi for Simulink models and 150 dpi
    for figures). This TIFF preview will show up on screen if
    matilda.eps is inserted as a Picture in a Word document, but
    the EPS will be used if the Word document is printed on a
    PostScript printer.

  Other options for figure windows:
    -Pprinter  % Specify the printer. On Windows and Unix.
    -noui      % Do not print UI control objects
    -painters  % Rendering for printing to be done in Painters mode
    -zbuffer   % Rendering for printing to be done in Z-buffer mode
    -opengl    % Rendering for printing to be done in OpenGL mode
    A note on renderers: when printing figures, MATLAB does not always
    use the same renderer as on screen. This is for efficiency reasons.
    There are cases, however, where the printed output is not exactly
    like the screen representation because of this. In these instances
    specifying -zbuffer or -opengl will more likely give you output that
    emulates the screen.

See the Using MATLAB Graphics manual for more information on printing.

See also printopt, printdlg, orient, imwrite, hgsave, saveas.
```