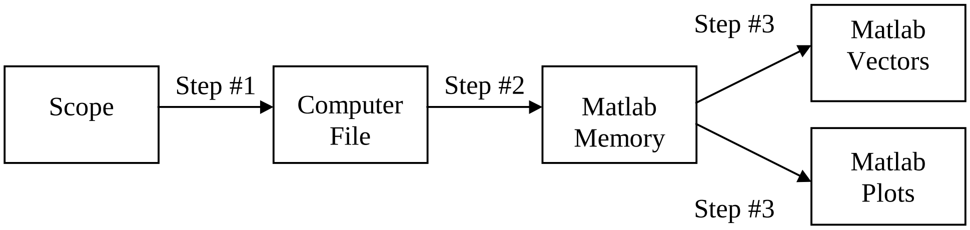
1. Importing Data into Matlab

1.1. Overview of the Process

Taking data from the oscilloscope and placing it into a useable format in Matlab is actually a 3 step process as is shown in the following:

Step #1. Move data from scope to computer using HP Benchlink Software
Step #2. Import the data into Matlab (using hpimport() function)
Step #3. Access the data in Matlab (using hpread() function)



1.2. Acquiring Data using the HP Benchlink Software

1.2.1. Functionality

The HP Benchlink software is a program that interfaces with the oscilloscope on the bench, downloading the channel data to the computer. The user is then able to save this data to a file, to be used later in other programs, such as Matlab.

1.2.2. Procedure

1.2.2.1. Turn on the Oscilloscope

1.2.2.2. Connect a waveform source to scope and adjust the scope controls to display the waveform on the scope screen just as it is to be downloaded onto the computer.

1.2.2.3. Start the HP Benchlink Scope Program by following the following path from the Start Menu: Start/Programs/HP Benchlink Suite/HP Benchlink Scope.

1.2.2.4. If the program tells you that the interface settings do not match the initialization file, say “Yes” to the prompt, select “SICL using HP 82341 cards” for the interface option, and then click “Search.” When the scope appears as an option, select it and click “OK.”

1.2.2.5. Select from the “Waveform” Menu, “New.” Select which channels from which to download data by placing a check in the box next to it. Ensure that the waveform is on screen of the scope and click “OK.” A window should appear with waveform(s) in it.

1.2.2.6. For future captures, simply press the “F7” key when the waveform appears correctly on the scope, and they will be captured appropriately.

1.2.2.7. To save the waveform to file (for import into Matlab), select from the “Waveform” menu, “Save as Time and Amplitude.” When the save dialog appears, be sure to change the file type to “Time, Amplitude (*.prn).” This is the only file type that will work with the import scripts for Matlab. Change the directory in which to save to “C:\ece316” This is the only directory in which you will be able to save. Save the file with a significant name that distinguishes it from any other; a file log might even be appropriate. Click the “Save” button to save the file which is now ready for import into Matlab.

(cont’d)

1.3. Using hpimport()

1.3.1. Functionality

This function reads oscilloscope data from a file created by the HP Benchlink Software. This function only imports the data into the working memory of Matlab, but it does not grant instant access to any of the channel data, nor does it make any plots. The hpread() function must be used after the hpimport() function is used. The motivation for this separation of functionality is in the speed of the hpimport() operation: hpimport() has to read from a file, which usually takes some time. With the current set-up, the file is imported once (using hpimport()), and then may be used various times (using hpread()).

1.3.2. Procedure

1.3.2.1. Copy the file “hpimport.m” into the “C:\ece316” directory from the class website.

1.3.2.2. Make sure that the current directory for the Matlab workspace is “C:\ece316.” If it is not, change the directory using the command, cd C:\ece316.

1.3.2.3. Call the hpimport() command, passing the name of the file in single quotes. The routine returns a subscripted data type(like an object in C++), which is then used by the hpread() routine to access and plot data from individual channels. An example follows in Matlab code:

```
scope_data_member = hpimport('practice.prn');
```

The scope data member, 'scope_data_member', contains all the necessary data from the file and will be used by hpread().

1.3.2.4. If a new file is saved by the HP Benchlink software, then hpimport() must be executed again on the new file, saving the data to a new data member:

```
scope2_data_member = hpimport('prac2.prn');
```

The data from the new file is located in 'scope2_data_member' and the data from the previous file is still located in ‘scope_data_member’.

1.4. Using hpread()

1.4.1. Functionality

This routine plots and returns individual-channel scope data taken from the oscilloscope. It is meant to be used multiple times on one set of data, unlike the hpimport() function, and it must be used AFTER the hpimport() function.. It can plot the data taken from the scope, return the data in Matlab horizontal vectors, or both. It is also able to accept style information with respect to the plot styles of Matlab. Hpread() will label each plot with the Channel Name, and the axis with the appropriate units (volts and seconds); these labels can be changed at any time.

1.4.2. Procedure

1.4.2.1. Copy the file “hpread.m” into the “C:\ece316” directory from the class website.

1.4.2.2. Make sure that the current directory for the Matlab workspace is “C:\ece316.” If it is not, change the directory using the command, cd C:\ece316.

(cont’d)

1.4.2.3. A routine call to hpread() is done in the following format:

```
[time,amp] = hpread(scope_data_member, 'selected_channel', 'style');
```

The parameter, 'scope_data_member', is a subscripted set of data created by the hpimport() routine. The data MUST first be downloaded into this data member (using hpimport()) before a call to hpread() may be made.

1.4.2.4. The parameter, 'selected_channel', is a string that is entered by the user, contained in single quotes, selecting which channel is to be plotted/returned. The options are as follows:

‘chan1’, ‘chan2’, ‘chan3’, ‘chan4’

Keep in mind that not all of the channels will contain data, as not all of them might not be turned on for use at one time.

1.4.2.5. The parameter 'style' selects the style of graphing used in the plot. The options are the same as defined in PLOT. (For information of the different styles with which to plot, execute the command help plot in the Matlab command window. The styles will be listed in the displayed help file.) There is also a new twist: Inputting for the style parameter, 'noplot', causes the routine to skip the plot of the channel and only returns the channel data.

1.4.2.6. The returned parameters [time,amp] contain the time(x) axis and the amplitude(y) axis data for the selected channel. They are returned in horizontal vectors of equal length. The function call may be made without the return parameters:

```
hpread(scope_data_member, 'selected_channel', 'style')
```

To suppress the return parameters from being displayed to the screen, place a semicolon after the routine call:

```
hpread(scope_data_member, 'selected_channel', 'style');
```

(cont’d)

1.5. Other Strategies

1.5.1. Multiple Plots on 1 Page

Multiple plots may be made on one page using a few simple Matlab commands. This is very helpful when multiple plots need to be printed, as it saves paper and time.

The first command to use is subplot(). This command allows the users to divide the plotting screen into a matrix-like field, and select which field will have the actual plot of the data. As the subplot command only divides the screen, but does not actually make the plot; another command, such as hpread(), is used to plot the data in the selected sub-field. A typical call sequence using the subplot() command follows:

```
%partition the figure into 4 rows of plots and 2 columns of plots
m = 4;
n = 2;

%plot the waveform data from hplot1 in the #1 space on the plot figure
subplot(m,n,1);
hpread(hplot1,'chan1');
title('Readback @ f = 100Hz');

% plot the waveform data from hplot2 in the #2 space on the plot figure
subplot(m,n,2);
hpread(hplot2,'chan1');
title('Readback @ f = 500Hz');

.
.
.
```

As one can see, the subplot command takes as parameters the number of rows of plots, the number of columns of plots and the selection number, respectively. The selection number chooses in which field the plot will be placed. It starts counting at the upper-left corner of the window and counts first across, and then down.

One can also see that the hpread() command was used to make the actual plot once the sub-field had been specified. The title() command was also used to add a title to each individual sub-field plot.

NOTE: To obtain best results, plot no more than 4 subplots on one page; more than this could cause overlapping of the plots. For this number, it is best to divide the screen into 3 rows and 2 columns of plots, and then only use the corner plots (selection numbers 1,2,5 and 6).

For more help on the subplot() command, execute the command, help subplot, in the Matlab command window.

(cont’d)

1.5.2. Making Multiple Pages of Plots in the Same M-file Script

If multiple pages of plots are required for one m-file, then the figure command may be used to force a new plot figure (window) to be created. This would be useful if one needed to make 8 plots and wanted to only plot 4 subplots per page. A typical use follows:

```
.
.
.

subplot(m,n,5);
hpread(hplot5,'chan1');
title('Readback @ f = 2000Hz');

subplot(m,n,6);
hpread(hplot6,'chan1');
title('Readback @ f = 2500Hz');

figure;

subplot(m,n,1);
hpread(hplot7,'chan1');
title('Readback @ f = 3000Hz');

subplot(m,n,2);
hpread(hplot8,'chan1');
title('Readback @ f = 4000Hz');

.
.
.
```

For more information, type help figure at the Matlab command prompt.

(cont’d)

1.5.3. Transferring Files to/from the ECE AFS Space

Keeping your files on one's own AFS account or disks is important, as the safety of the files is not guaranteed on the lab bench machines. To upload files to/from the ECE AFS space use the following steps:

1.5.3.1. Open the WS FTP (ftp client program) from the following path off the start menu: Start/Programs/Ws_ftp/Ws_ftp 95 LE.

1.5.3.2. If the connect screen (Session Profile) window does not appear automatically, click on the "Connect" Button at the bottom of the application window. As there are many different hosts from which to choose, there is no one correct host; therefore, enter in (substitute Greek letter here).ece.cmu.edu for the host name. (An example is pi.ece.cmu.edu.) Make sure that the "Anonymous" checkbox is NOT checked. Enter in your user name and password, but DO NOT SAVE YOUR PASSWORD! Click "New" and the computer will connect to the remote server.

1.5.3.3. To upload files to one's ECE AFS space, change the directory of the left-hand side of the window (local computer) to "C:\ece316". The select which of the local file(s) are to be uploaded to the directory. Change the directory of the right-hand side (remote computer) to the desired directory and click on the "➔" button in between the two windows to upload the file(s).

1.5.3.4. To download files from the ECE AFS Space, change the directories just as when uploading files, selecting this time which of the files on the right-hand side (remote computer) are to be downloaded. Then click on the "←" button to perform the actual download.

1.5.4. Printing from Matlab

Should you find it necessary to print the plots from Matlab, simply select "Print" from the "File" menu and select the print to which you need to print. Keep in mind that printing plots may not always be necessary, and could take a significant amount of time and include confusion among plots if many different groups print them. Check with the TA to see which printer to which it is best to send your plots.

IMPORTANT NOTE: Be sure to use the strategies set forth in sections 1.4.1 and 1.4.2 of this document, placing multiple scope captures on one page. If each capture is sent to the printers separately, for each group, there would be a massive overload of the printer and a large waste of paper! Keep in mind this is very useful for multiple scope captures (actual waveforms taken directly from the scope), but for curves that may be manually assembled in Matlab, multiple plots per page may not be acceptable. Ask the TA any necessary questions.