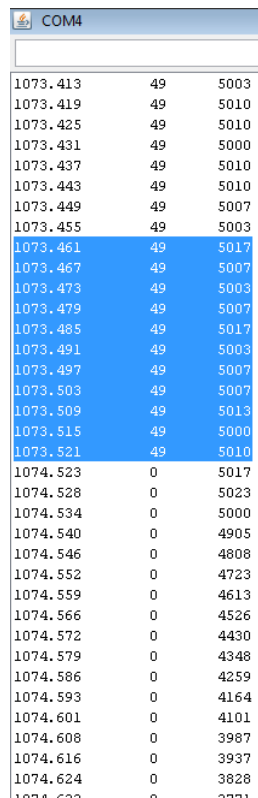


Importing Data to the MATLAB Workspace

Purpose:

The purpose of this guide is to go through the step-by-step process of importing data from the Arduino COM window or an Excel spreadsheet, into the MATLAB workspace. The Arduino IDE has a built-in serial monitor that can be used to communicate with an Arduino board. This window is used to control the motor throttle setting value of the Uno, which is then sent to the ESC using the Motor Testing Program. You can copy and paste the resulting data into MATLAB for analysis, or save them as a text file if you so choose. While recording data readings from a digital scale, such as during thrust and torque tests, it is convenient to record the data into an Excel spreadsheet for later import into MATLAB.

Step 1: Once a data run is complete, scroll to the bottom of the COM window and select all of the test data. Data we don't want can be filtered out later manually or using the Data Analysis program, but it is sometimes helpful to exclude data at this stage.



COM4		
1073.413	49	5003
1073.419	49	5010
1073.425	49	5010
1073.431	49	5000
1073.437	49	5010
1073.443	49	5010
1073.449	49	5007
1073.455	49	5003
1073.461	49	5017
1073.467	49	5007
1073.473	49	5003
1073.479	49	5007
1073.485	49	5017
1073.491	49	5003
1073.497	49	5007
1073.503	49	5007
1073.509	49	5013
1073.515	49	5000
1073.521	49	5010
1074.523	0	5017
1074.528	0	5023
1074.534	0	5000
1074.540	0	4905
1074.546	0	4808
1074.552	0	4723
1074.559	0	4613
1074.566	0	4526
1074.572	0	4430
1074.579	0	4348
1074.586	0	4259
1074.593	0	4164
1074.601	0	4101
1074.608	0	3987
1074.616	0	3937
1074.624	0	3828

Figure 1. Selecting data from the COM window

Step 2: Hold “Shift” (or equivalent for your computer) and click to select all the way up to the top of the data. As seen below in Figure 2, **make sure to include the column headings in the selection**. This makes MATLAB correctly name the columns during import, saving you some time.

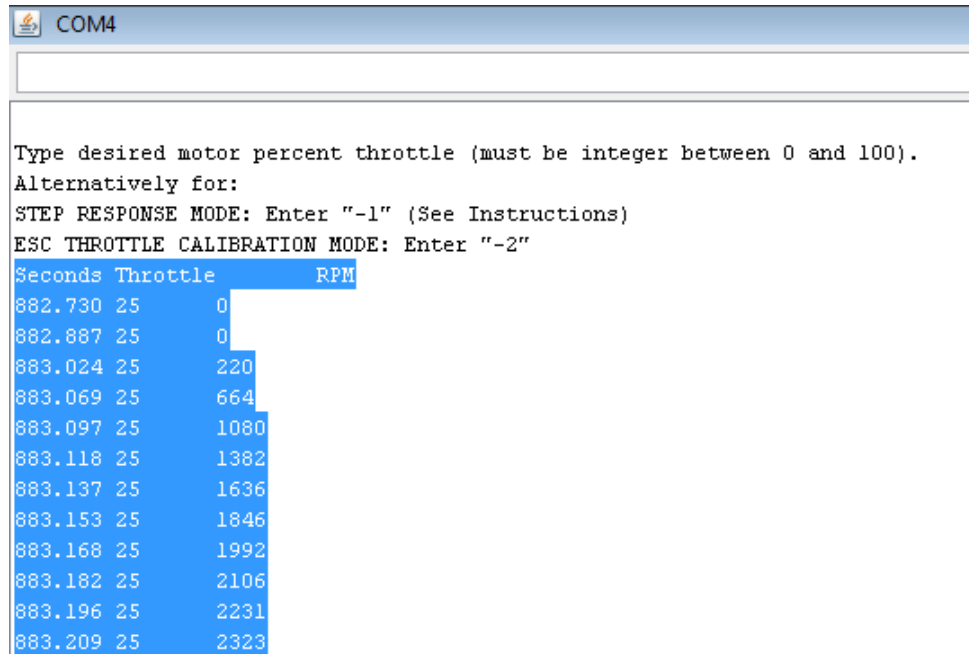


Figure 2. Selecting the rest of the data from the COM window (including headings)

Step 3: “Ctrl + C” (or equivalent) to copy the data, and Right-click + Paste into the MATLAB workspace window. This can be seen below in Figure 3.

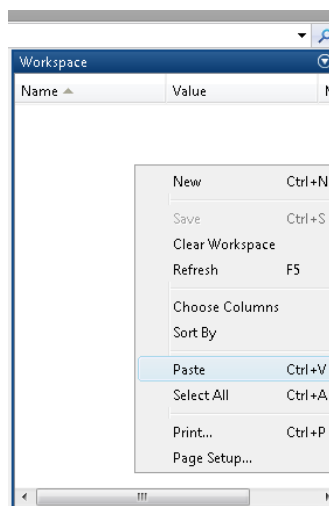


Figure 3. Pasting data into the MATLAB workspace

Step 4: The “Import” window will open, as seen below in Figure 4. MATLAB is able to automatically populate the column headings as well (if not, columns must be labeled “Seconds”, “Throttle”, and “RPM”). Select the green checkbox “Import Selection” at the top right of the Import window and MATLAB will import and parse the data (Figure 5).

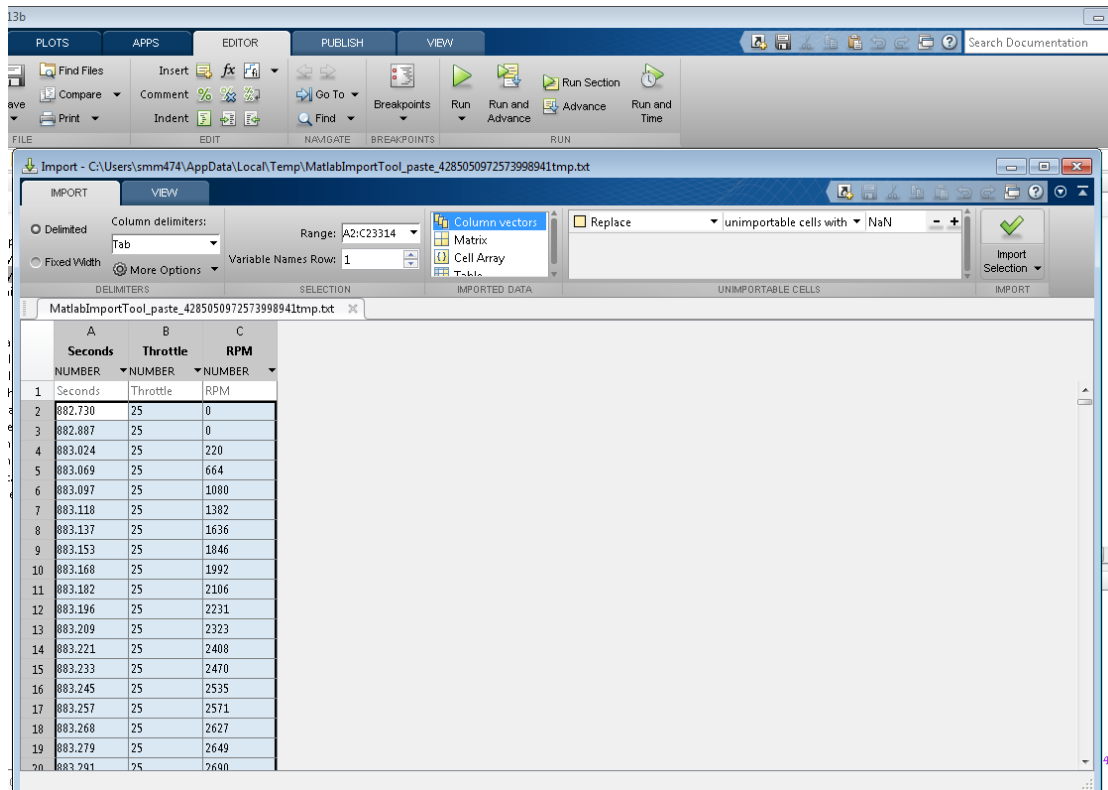


Figure 4. Importing the selection into the workspace

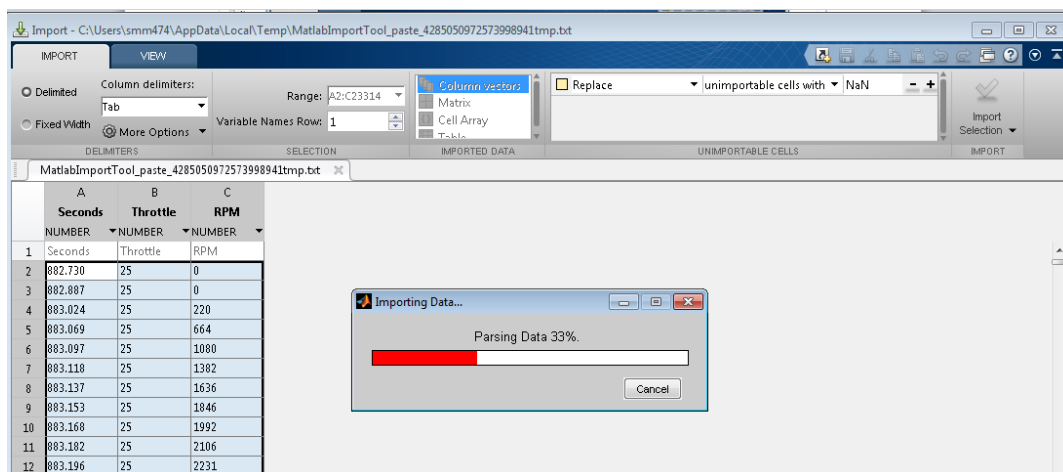


Figure 5. Parsing the data

The data is now imported to the MATLAB workspace (Figure 6) as individual column vectors.



Figure 6. Data imported to workspace

Step 5: Copy raw scale measurements from an Excel sheet, as seen below in Figure 7. If Excel was not used, you can manually enter a column vector of measurements.

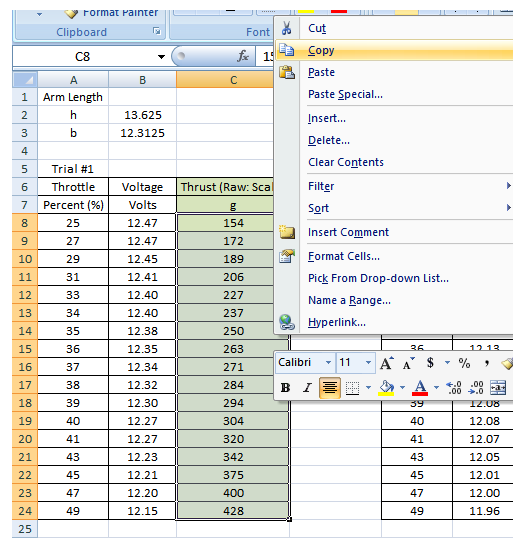


Figure 7. Copying data from Excel sheet

Step 6: This data from Excel can now be pasted into the MATLAB workspace along with the data from the Arduino COM window. This can be seen in Figure 8.

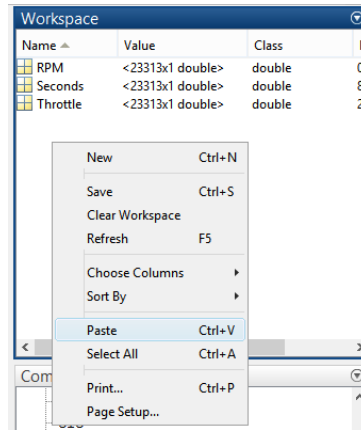


Figure 8. Pasting data into workspace

Step 7: This step is important, as the analysis tools we provide require that the measurements from the scale be stored as a vector with a specific name. After selecting “Paste” into the workspace, MATLAB’s Import window will once again open, except that MATLAB now doesn’t know what to call the column heading. For this case, we must call the column heading “**gramsMeas**”. Once the heading is labeled, you can select “Import Selection” at the top right of the screen and MATLAB will import this data from Excel into a vector called “**gramsMeas**”. This is seen below in Figure 9.

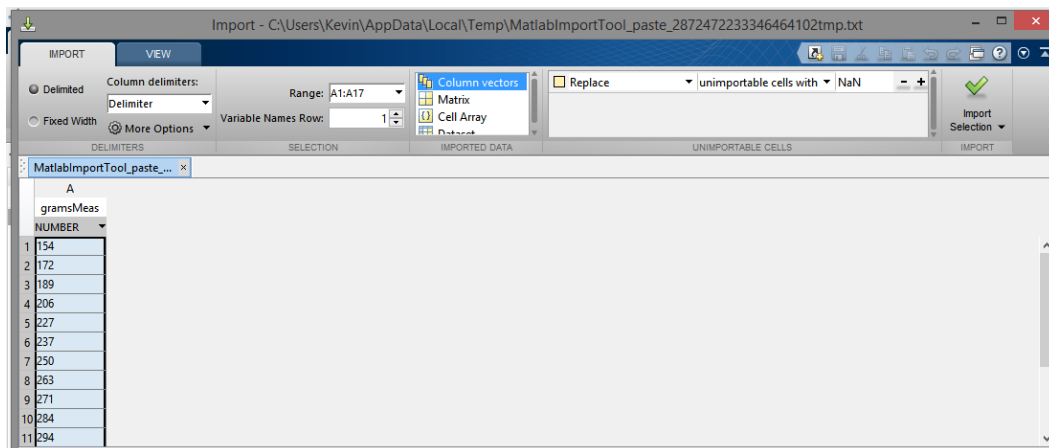


Figure 9. Importing the data into vector called "gramsMeas"

Step 8: Finally, we are able to save all of this workspace data as a .mat file into a location of our choice for later use in the data analysis programs. Select all (4) variables in the workspace window, right click, “Save As...” and Save the data into the desired folder with a meaningful name (date, tests performed, variables of interest, etc.). See Figure 10 below.

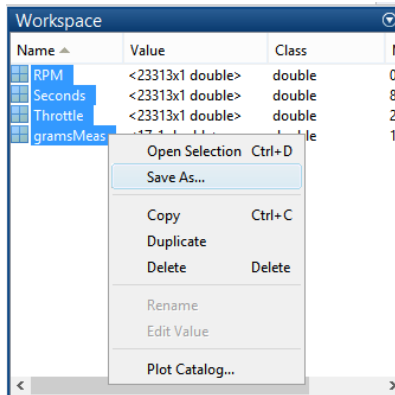


Figure 10. Saving the data