Adapted from Take Home Labs, Oklahoma State University

!

Practical Assignment – 1: Arduino interface with Simulink®

!

1 OBJECTIVE

This assignment is an introduction to the experimental set-up to be used throughout this course. The instructions for installing "MATLAB Support Package for Arduino Hardware" and "Simulink Support Package for Arduino Hardware" are given first. After installing these necessary Add-Ons, you will verify that everything is installed correctly by using the hardware and software to successfully blink an LED on the Arduino board, using two different methods for running models on the Arduino with Simulink: Normal Mode and External Mode.

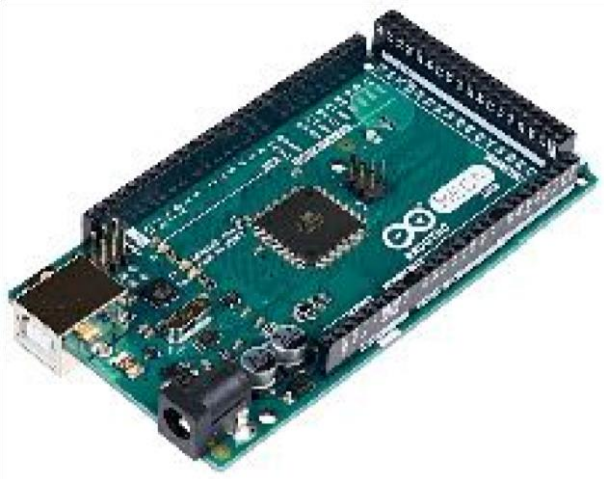
2 SETUP

2.1 REQUIRED MATERIALS

2.1.1 HARDWARE

Your kit will contain either Arduino MEGA 2560 or Arduino UNO. The experiment procedure is ***identical*** for both the boards.

- Arduino Mega 2560 R3 **OR** Arduino Uno R3



We will use original boards and not the clones to avoid any difficulties in the interface with Simulink/MATLAB.

-
- USB cable (Standard A to B plug)

2.1.2 SOFTWARE

- Matlab/Simulink R2013a or later (Institute licences are available for the latest versions)

  The steps and images related to Matlab/Simulink for this experiment are created using Matlab/ Simulink 2018a. Therefore some steps and images may be a little different for other versions of Matlab.

- Personal Computer with administrator access.

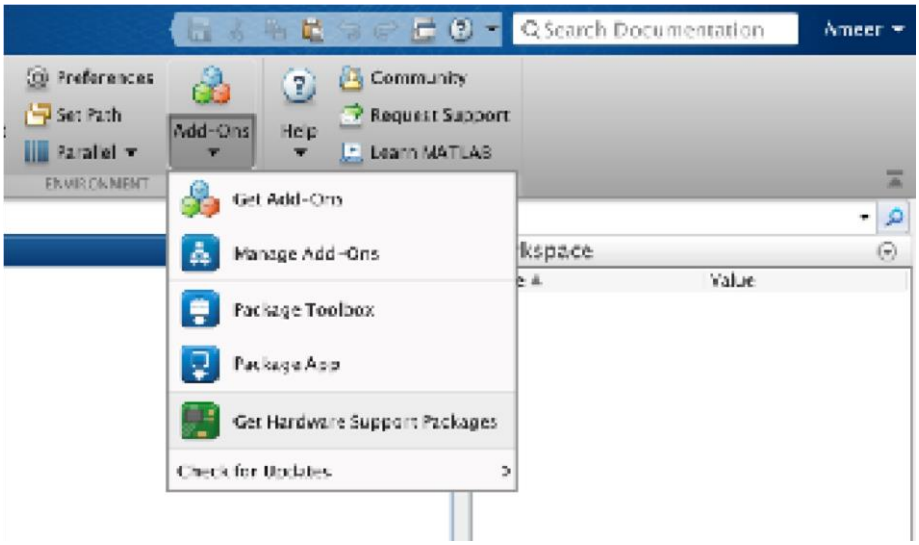
3 EXPERIMENTAL PROCEDURES

The first exercise of this lab demonstrates how to install the Add-Ons: "MATLAB Support Package for Arduino Hardware" and "Simulink Support Package for Arduino Hardware". The next exercise will verify that the first exercise was successful. The exercise consists of creating Simulink models to blink the Arduino's built-in LED using Normal Mode and External Mode.

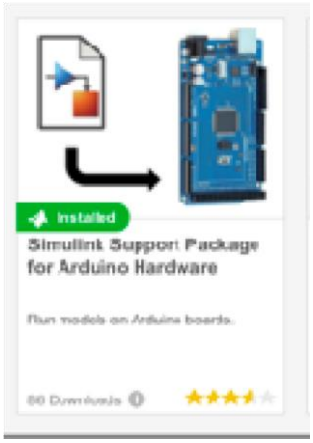
1. Connect the Arduino board to the computer using the USB cable. The Power ON LED on the board will glow as shown below:

3.1 EXERCISE 1: INSTALLING ARDUINO SUPPORT IN SIMULINK

1. On the "Home" toolstrip, navigate to "Resources" and click the "Add-Ons" dropdown menu followed by "Get Hardware Support Packages" as shown in the Figure below.



2. This will open a window named “Add-On Explorer”. Choose the following two packages (Exact Name should match):
    a. MATLAB Support Package for Arduino Hardware
    b. Simulink Support Package for Arduino Hardware



For each of the package, select “install”.

3. A MathWorks account will be needed to proceed, so log in with your account linked to your institute email address.

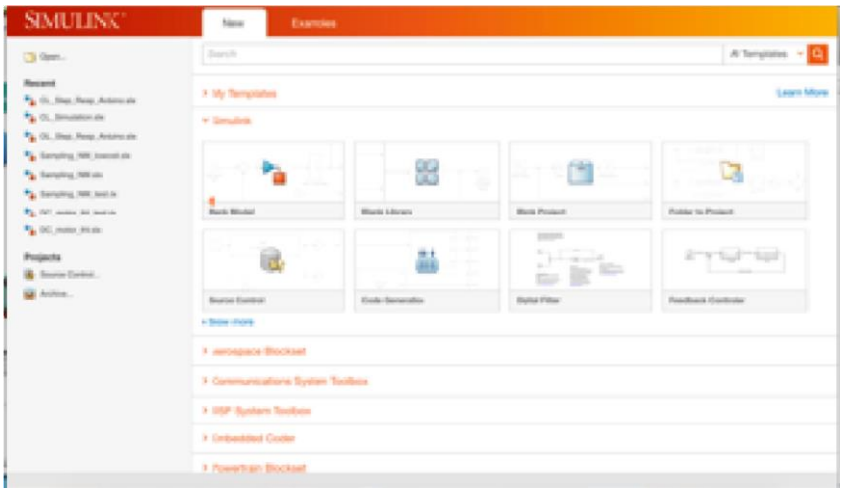
4. Follow the remainder of the on-screen prompts until the installation is finished, and exit out by clicking “Finish".

3.1.1 CHECKPOINT:

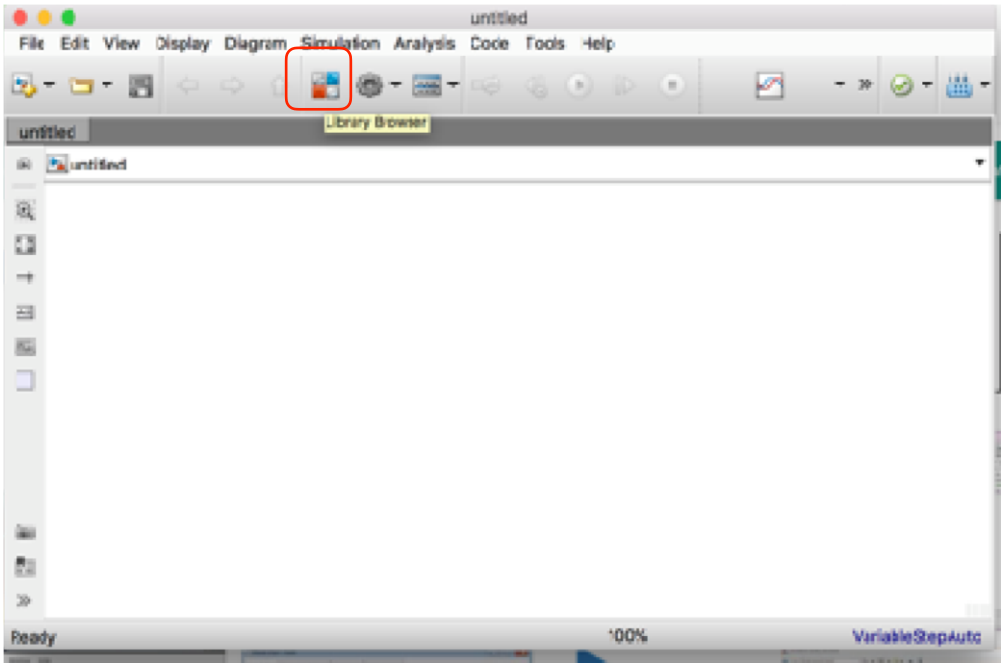
Verify that the installation was successful by performing the following:

5. Type *simulink* in the Matlab Command Window or click on icon.

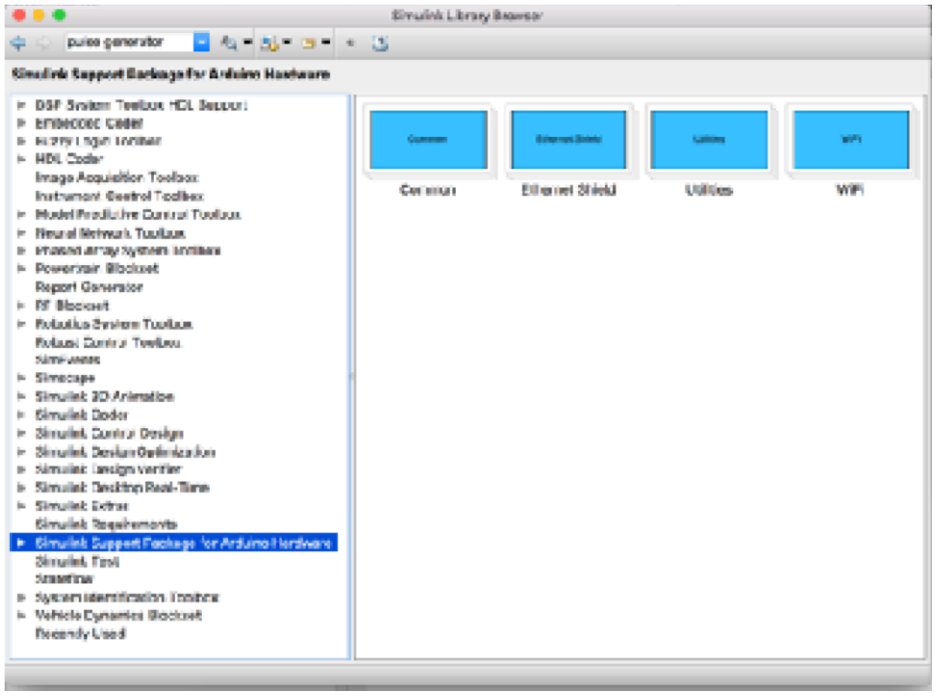
6. A window similar to the following will open.



7. Click on “Blank Model”.
8. It will open a new window similar to the following:



9. Click on the “Library Browser” icon which is highlighted in Red in the figure above.
10. Now, in the “Libraries" section, verify that the “Simulink Support Package for Arduino Hardware” is there, as shown in the figure below:



3.2 Exercise 2: Blinking an LED

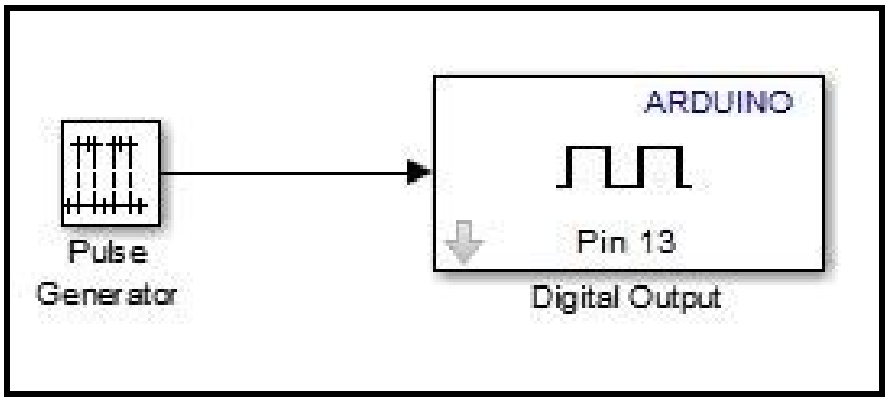
Now that the Arduino support is installed in Simulink, the next step is verifying the PC can communicate with the board properly.

There are two methods that will be used to demonstrate proper communication between the board and the PC: Normal Mode and External Mode. Both methods have their pros and cons, and, depending on the project, one method may prove to be more beneficial than the other. Thus, it is important to introduce both methods now so that the student is familiar with them before starting any of the later labs. A simple experiment to introduce both of these methods is blinking an LED.

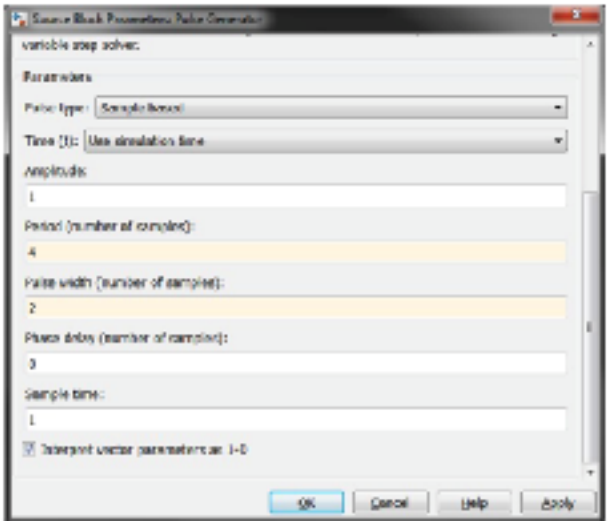
Besides the Arduino and the USB cable, there is no additional hardware needed for this experiment, because the Arduino has a built-in LED.

3.2.1 Normal Mode

As indicated by its name, Normal Mode is just the normal method used to download programs onto the Arduino. A program downloaded to the board in Normal Mode will stay on the board until another program overwrites it. This means that even when the board is disconnected from power and reconnected later, the program is still loaded onto the board, and will just start executing again. The main disadvantage with Normal Mode is that if any data needs to be recorded, they will be more difficult to acquire than with External Mode. The main advantage with Normal Mode is that it allows programs to run at a much faster rate than External Mode. The following steps show how to use Normal Mode to blink the onboard LED of the Arduino. The final Simulink Model will look like following:



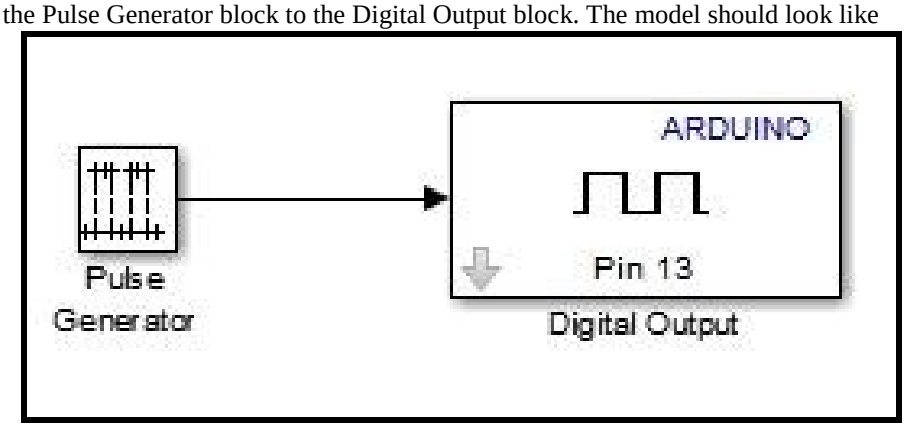
18. Open MATLAB, then open a new Simulink Model using either of the following methods:
    - In the Matlab window's "Home" toolstrip, choose File → New → Simulink Model.
    - Type *simulink* in the Matlab Command Window, then press the New Model !icon.
19. Open the Library Browser by pressing the ! icon, or selecting View→LibraryBrowser.
20. Under the Libraries section add a Pulse Generator block to the model:
    - Simulink → Sources → Pulse Generator
21. Under the Libraries section add an Arduino Digital Output block by selecting:
    Matlab R2013b/R2014a: Simulink → Simulink Support Package for Arduino Hardware → Common → Digital Output
22. Save the Simulink model as any filename.
23. Double-click the Pulse Generator block, make the following changes and then hit OK:
    - Set "Pulse type" to Sample based
    - Set "Period" to 4
    - Set "Pulse width" to 2
    - Set "Sample time" to 1



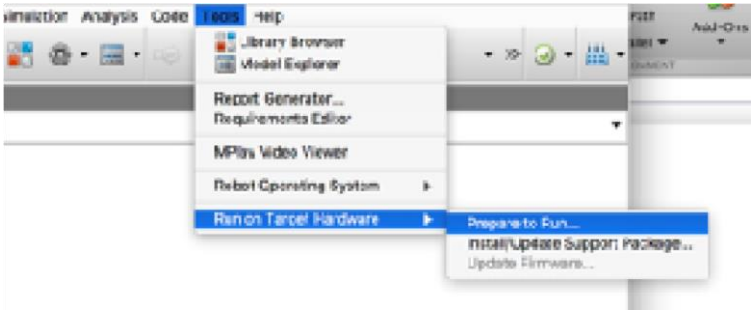
24. Double-click the Arduino Digital Output block, make the following changes and then hit OK:

• Set Pin number to 13, this is the pin the onboard LED is connected to.

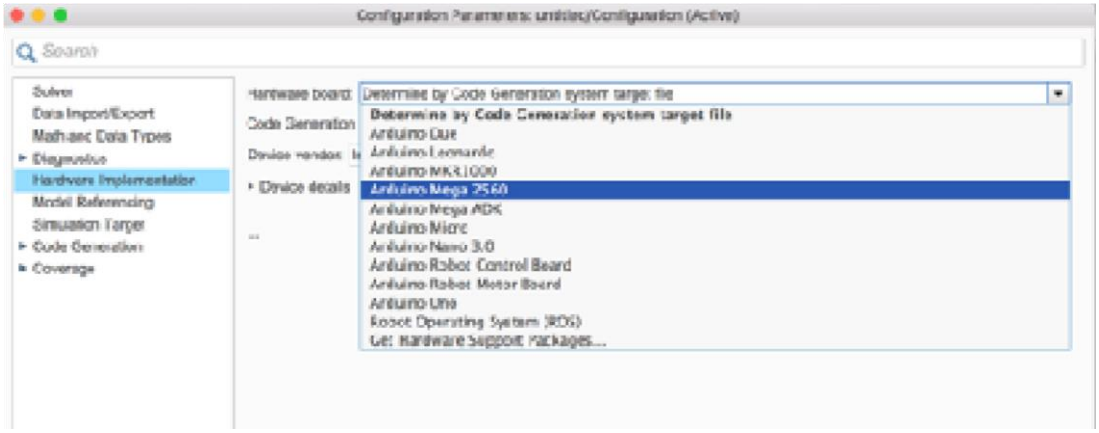
25. Connect the Pulse Generator block to the Digital Output block. The model should look like



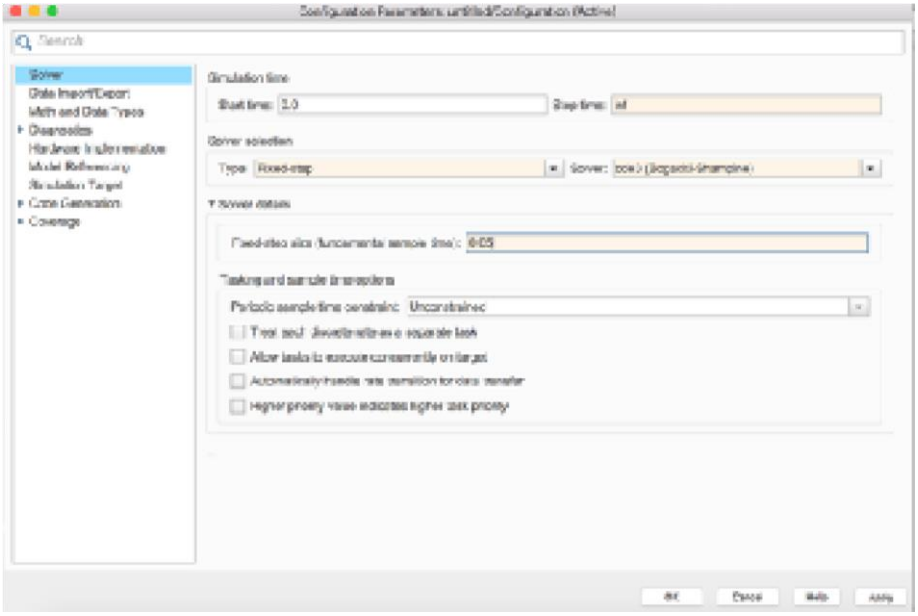
26. Now select Tools → Run on Target Hardware → Prepare to Run...



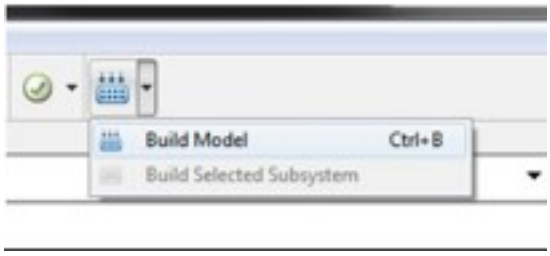
27. A window should appear as that in the figure below. Set Hardware board to Arduino Mega 2560, and wait a moment for the window to update.



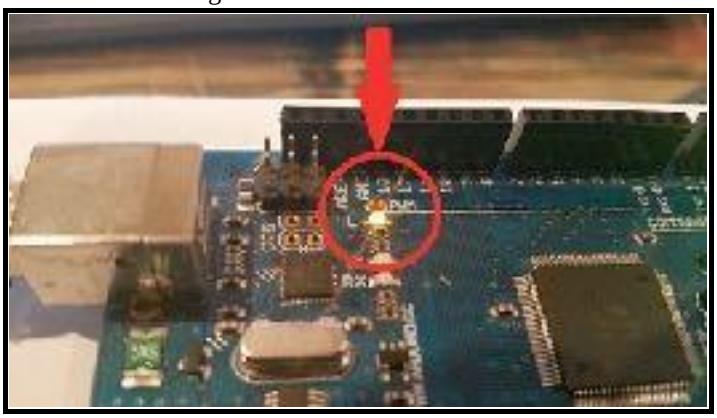
28. Select Solver on the left hand side of the window, then make the following changes and hit OK.
    - Set “Stop time” to *inf*
    - Set “Type" to *Fixed-step*
    - In “Solver details”, set “Fixed-step size" to *0.05*



29. Now run the model by clicking the “Build Model" button



30. Wait until the the model has successfully downloaded to the board, now observe the onboard LED of the Arduino shown in the figure below.



31. Describe what the LED is doing. Estimate and record the timing. Is the timing consistent with the settings of the pulse generator you made earlier? Explain.
32. Change the Pulse Generator block sample time from 1 second to 0.1 seconds.
33. Download the modified model onto the board. Describe the changes in the LED operation. How has the timing changed? Did it change in the way that you expected?
34. Change the parameters of the Pulse Generator block so that the LED will be on for 4 seconds and off for 1 second. Record your settings, download the modified model, and describe the resulting

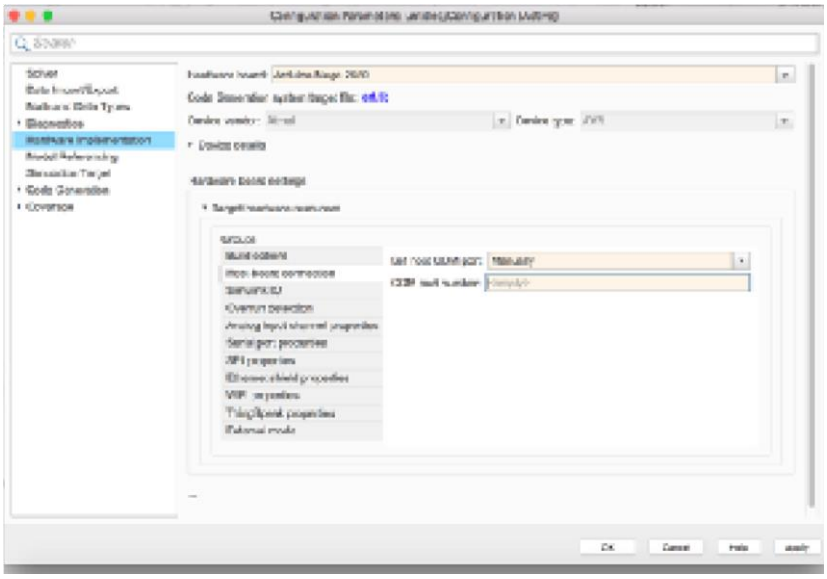
operation. If the timing does not produce a 4 second on period and a 1 second off period, make the necessary changes to the parameters of the Pulse Generator and test the operation again.

35. Record your data in the table of the following format.

Pulse Generator Settings	Description of LED behavior

TROUBLESHOOTING: If there are any errors encountered throughout this section, try the following:

- If Matlab cannot find the board when attempting to run the model, try setting the COM port manually. In Simulink, go to Tools → Run on Target Hardware → Prepare to Run…, then change "Set host COM port:" from Automatically to Manually as shown in the figure below



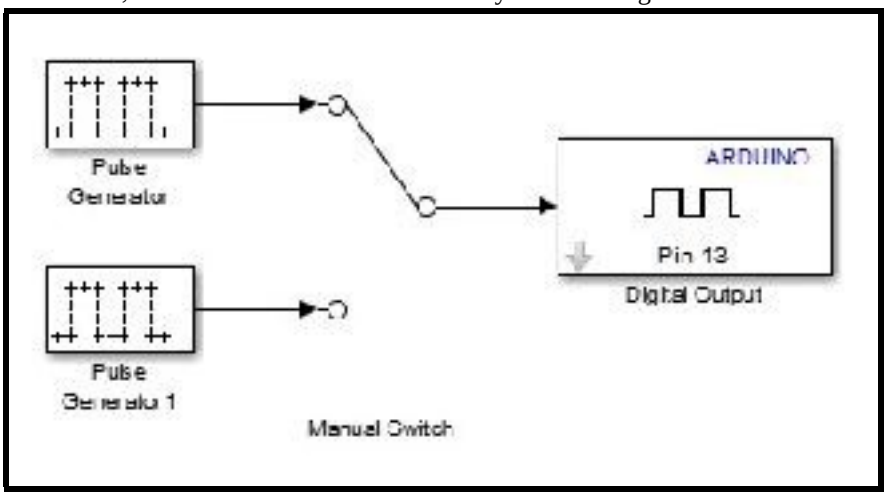
- Find out the COM port number being used for Arduino (based on operating system, there are different ways to find out the COM port number. Google it.) and enter it.
- If Simulink cannot connect to the board either: try disconnecting and reconnecting the USB cable, or closing and reopening Matlab.
- Any other issues encountered may be solved by reading the Arduino troubleshooting page located at http://www.arduino.cc/en/Guide/Troubleshooting .

3.3.2 EXTERNAL MODE

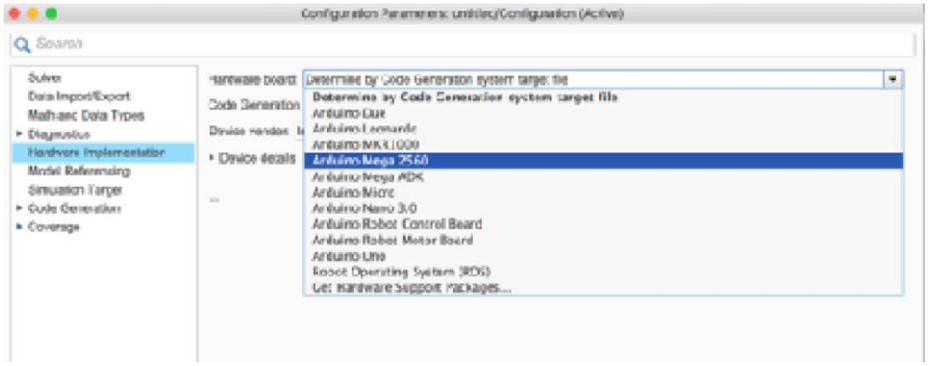
In External Mode you are given the ability to tune parameters, and monitor data in real-time, without having to re-download the model each time parameters are adjusted. Once External mode is started, it uses the serial port to communicate between the board and the PC. Therefore, to run in External mode, the USB should be connected between the board and the PC. This ability to adjust and monitor changes in real-time is what makes External Mode such an enticing option for running these experiments. You

can benefit from being able to see the immediate effects your changes have on the hardware. However, this ability to make changes real-time comes at a cost. To communicate back and forth, the serial port is kept open and has a limited amount of bandwidth. This limitation affects the rate at which the program can run. External Mode's ability to make real-time adjustments will be shown in the following steps, using the Arduino's onboard LED.

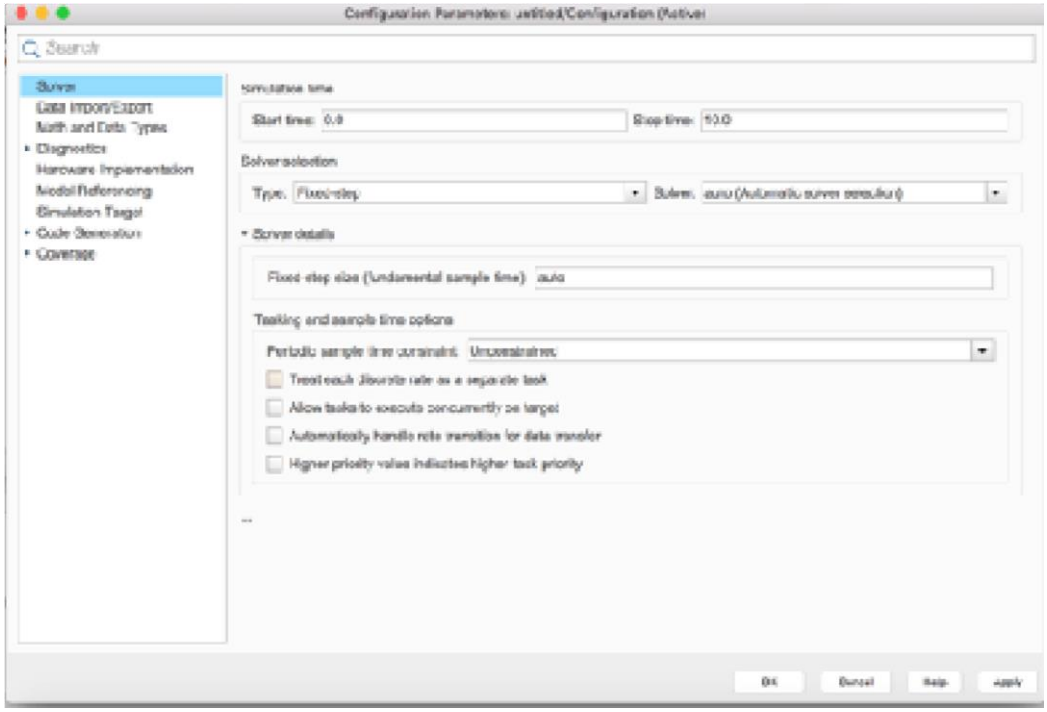
34. Open a new Simulink Model, and name it something different from the previous model.
35. Add the following items from the Simulink Library Browser:
    - Two Pulse Generators: Simulink → Sources → Pulse Generator
    - Manual Switch: Simulink → Signal Routing → Manual Switch
    - Arduino Digital Output Block: Simulink Support Package for Arduino Hardware → Common → Digital Output
36. Save the model, and connect the blocks so that they look like Figure 3.23.



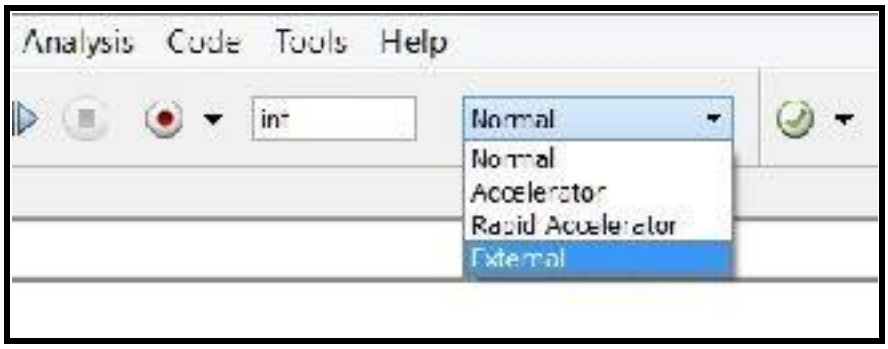
37. Double-click the Digital Output block and set the pin number to be 13.
38. Double-click the top Pulse Generator block, make the following changes and hit OK:
    - Set "Pulse type" to Sample based
    - Set "Period" to 4
    - Set "Pulse width" to 2
    - Set "Sample time" to 1 second
39. Double-click the bottom Pulse Generator block, make the following changes and hit OK:
    - Set "Pulse type" to Sample based
    - Set "Period" to 4
    - Set "Pulse width" to 2
    - Set "Sample time" to 0.1 second
40. Set up the Arduino to run in External Mode. Select Tools → Run on Target Hardware → Prepare to Run…
41. Set Target hardware to Arduino Mega 2560, and wait a moment for the window to update.



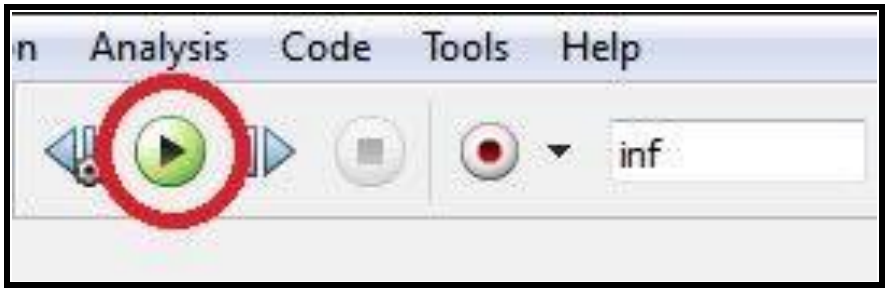
42. Select Solver on the left hand side of the window, then make the following changes and hit OK.
    - Set “Stop time" to *inf*
    - Set “Type" to *Fixed-step*
    - Set “Fixed-step size" to *.05*
    - Uncheck “Treat each discrete rate as a separate task” Refer to the figure on the next page.



43. Make sure the drop-down menu shown in the figure below is set to External.



44. Download the model to the Arduino using the same method used for Normal Mode.
45. Press the Run button , as shown in the figure below:



46. Record what the Arduino's LED is doing. Estimate timing. Is the timing consistent with the parameter settings?
47. Double click the Manual Switch in the Simulink Model. The switch should now be connecting the bottom Pulse Generator to the Digital Output.
48. Record what the Arduino's LED is doing now. Describe any timing changes. Explain why the timing changed. How is the process different than Normal Mode? Which Mode is better? Why?
49. Press the Stop button , as shown in Figure 3.29.

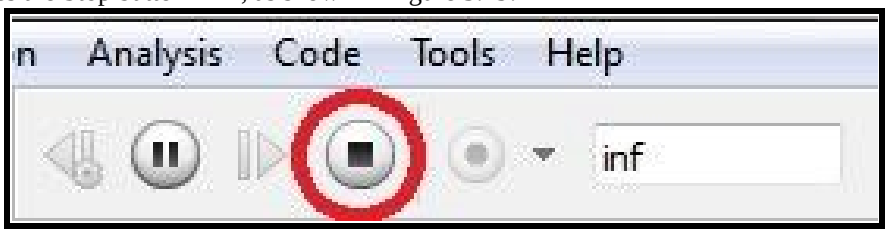


Figure 3.29: External Mode Disconnect: R2013b or later

50. Disconnect the USB from the Arduino and PC.

TROUBLESHOOTING:

• The most common error that may be encountered using External Mode is Simulink being unable to connect to the Arduino. This can happen when running in External mode if a previous simulation was not stopped correctly. The serial port may have been left open creating issues when trying to reconnect to the board. Another possible cause is that the model was not rebuilt after making changes. Possible fixes for this error are:

– Try rebuilding the model.

– Disconnect the USB from the Arduino and reconnect. (This usually fixes most of the errors).

– Try hitting the reset button on the Arduino, shown below:

– Close Matlab and re-open.

– Any other issues encountered may be solved by reading the Arduino troubleshooting page located at http://www.arduino.cc/en/Guide/Troubleshooting.

4 Conclusion

This introductory lab was constructed to show how to install the necessary support for the Arduino in Simulink, and on the PC. Control of the Arduino's onboard LED was used to verify that everything was installed correctly. There were two methods used to test this – Normal Mode and External Mode. This experiment only provided a glimpse of these two modes. A more in depth comparison of the two modes is provided in the subsequent experiments.

5 Deliverables

The assignment is **to be submitted on i-learn.**. Capture a Screen shot of your final model in Simulink and submit it.