RENSSELAER MECHATRONICS

Blinking the LED

Lab Objectives

- Verify Arduino software package for Simulink using a digital output to light a LED
- Communicate with the target board (Arduino) using external mode by changing the brightness of an LED with PWM
- Use overrun detection to determine how fast you can run your code

Table of Contents

Lab Objectives 1

Part 1: Arduino Toolbox Installation for Simulink 2

Objective: 2

Simulink Arduino Library Installation: 2

How do I find the COM port for my Arduino Board: ..... **Error! Bookmark not defined.**

Part 2: Communicating with External Mode 6

Objective 6

Simulink 6

Deploy to Hardware: 8

External Mode – communicating with the hardware 9

External Mode: 9

External Mode 11

Stopping the Simulation: 12

Debugging – cannot connect to device or cannot download code 12

Checkpoint: 13

Part 3 (Optional): Evaluating code execution speed – how fast can I run my code 13

Objective: 13

Simulink: 13

Part 4: Troubleshooting 15

An error appears when trying to run the file similar to “build procedure aborted”, “make command returned an error…” 15

Error appears when trying to open the Simulink file: 15

Simulink Support for Arduino Hardware does not appear in the Simulink library browser: 16

Device is not recognized in device manager when it is plugged in..... 16

File is not compiling, a long list or errors appears. 16

In addition to the above, make sure that MATLAB working directory is not Windows/system32 or some other system folder that requires admin rights to edit. To be safe, set the working folder to your documents/MATLAB 16

Red block in Simulink diagram: 16

Error about board not connected to computer 17

Error Dialog. Error using open… Simulink diagram file name has parenthesis in it..... 19

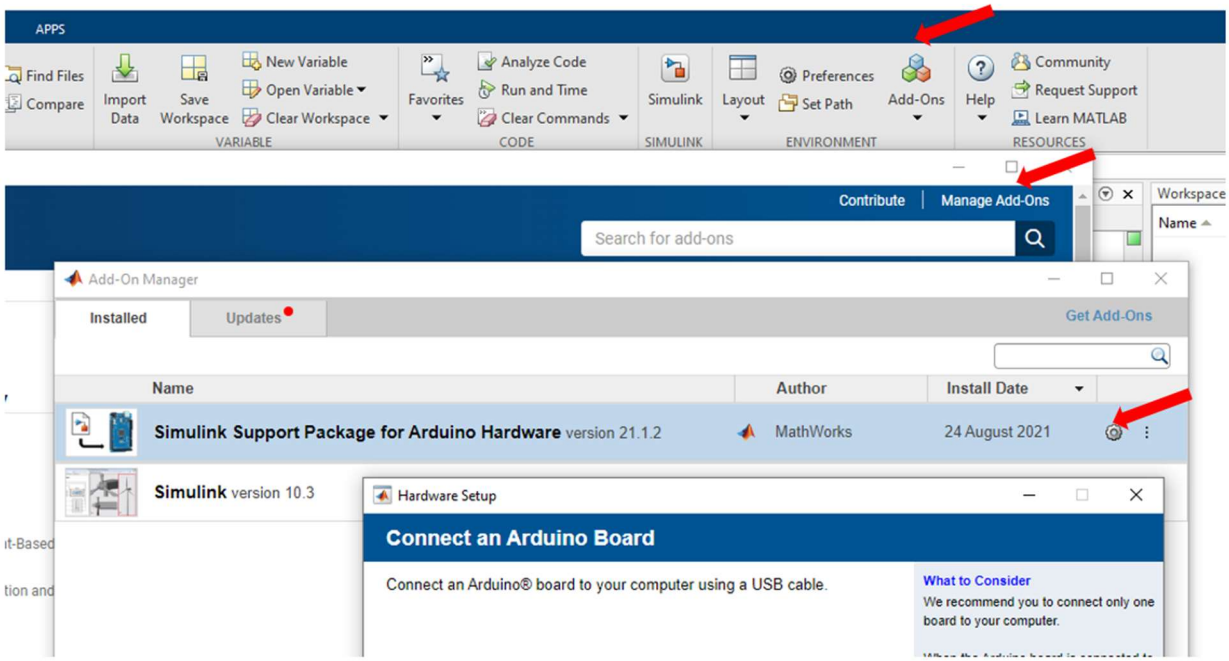
Part 1: Arduino Toolbox Installation for Simulink

Objective:

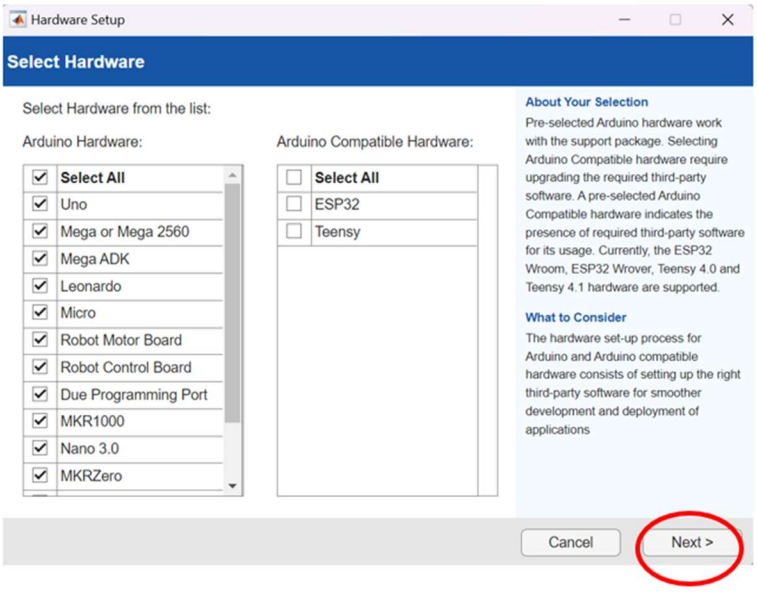
- Install and verify Arduino software package for Simulink using a digital output to light a LED with MATLAB/Simulink 2024a or later.

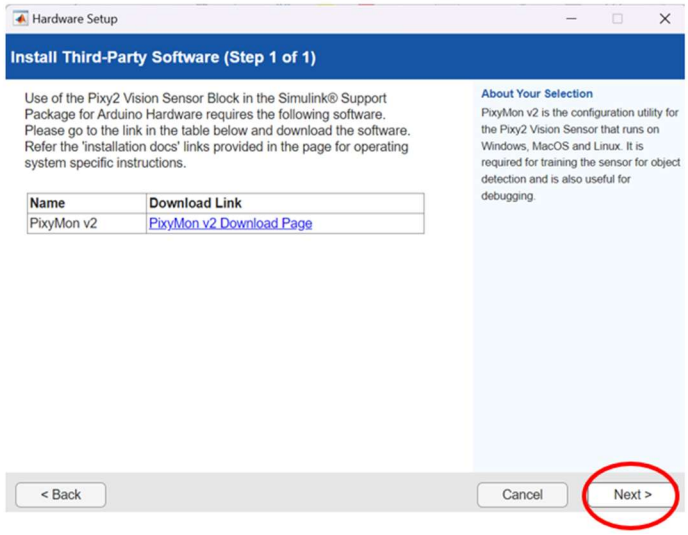
Verify Simulink Arduino Library Installation:

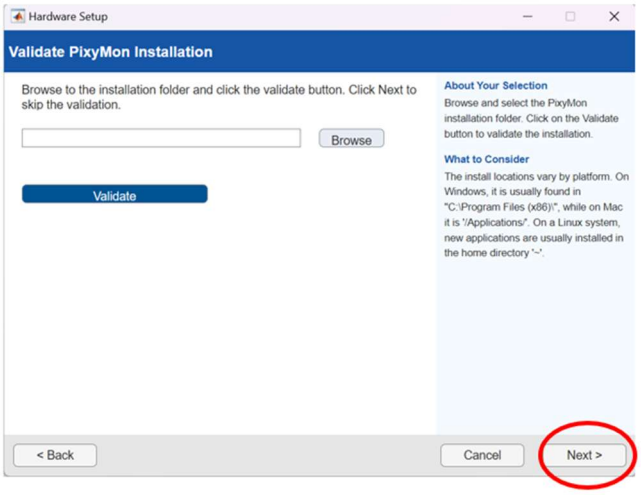
- Open MATLAB 2024a. From the MATLAB window select “Add-Ons”, “Manage Add-Ons”, and click the gear icon for the Simulink Support Package for Arduino Hardware.

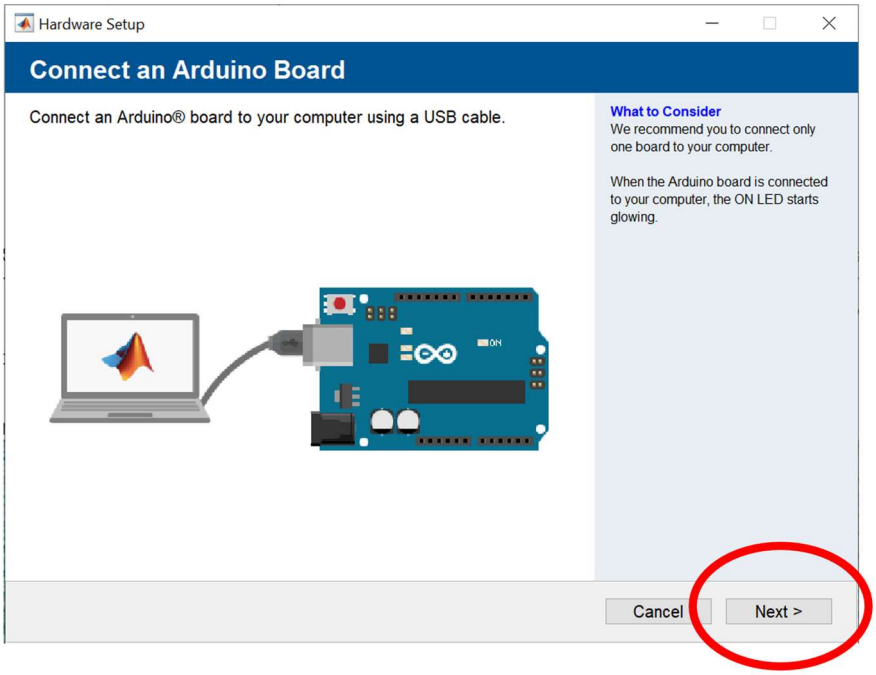


- Leave the default boards selected and select “Next” for the following 3 screens:

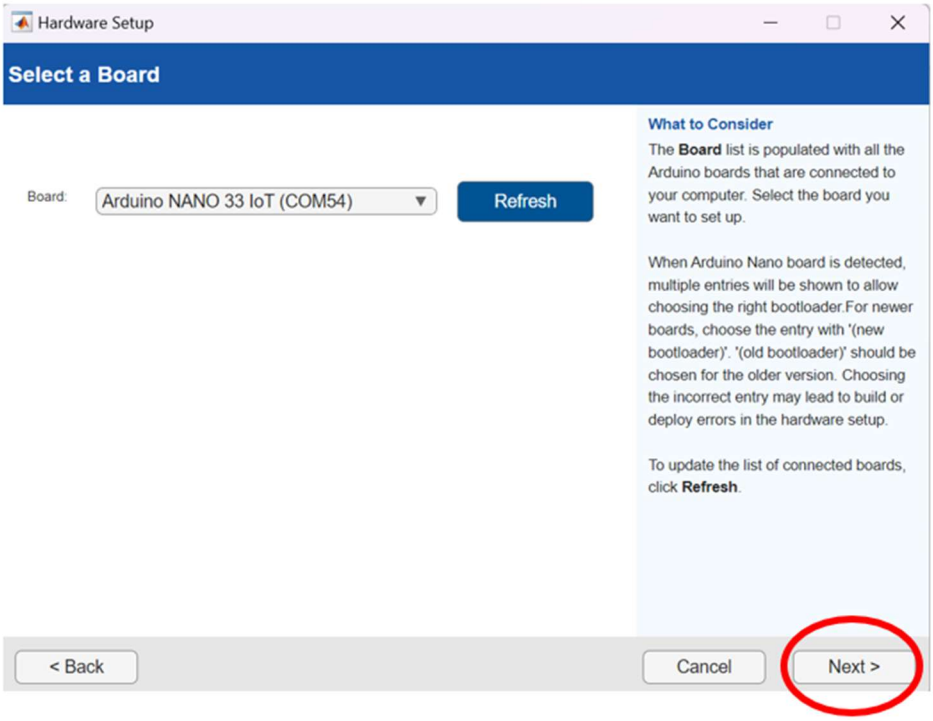




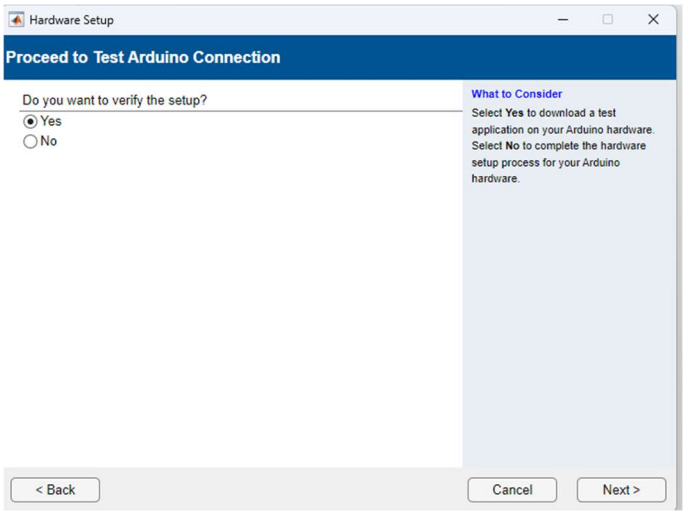




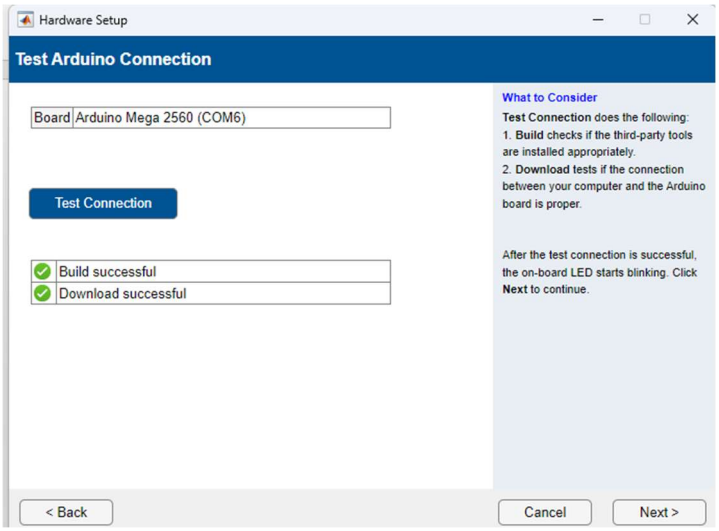
Make sure your board is plugged in and select Next. It will take a couple minutes before getting to the next screen. The next screen should automatically detect your board in which case you can select “Next”.



Click Next to verify the setup.



Allow the Java. Verify it builds and downloads:



You are now ready to use your hardware.

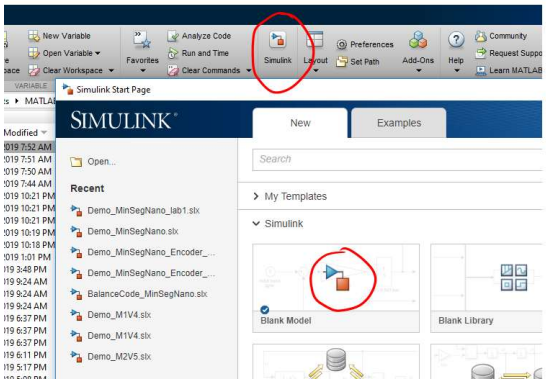
Part 2: Communicating with External Mode

Objective

- Communicate with the target board (Arduino) using external mode to turn LED 13 on and off.

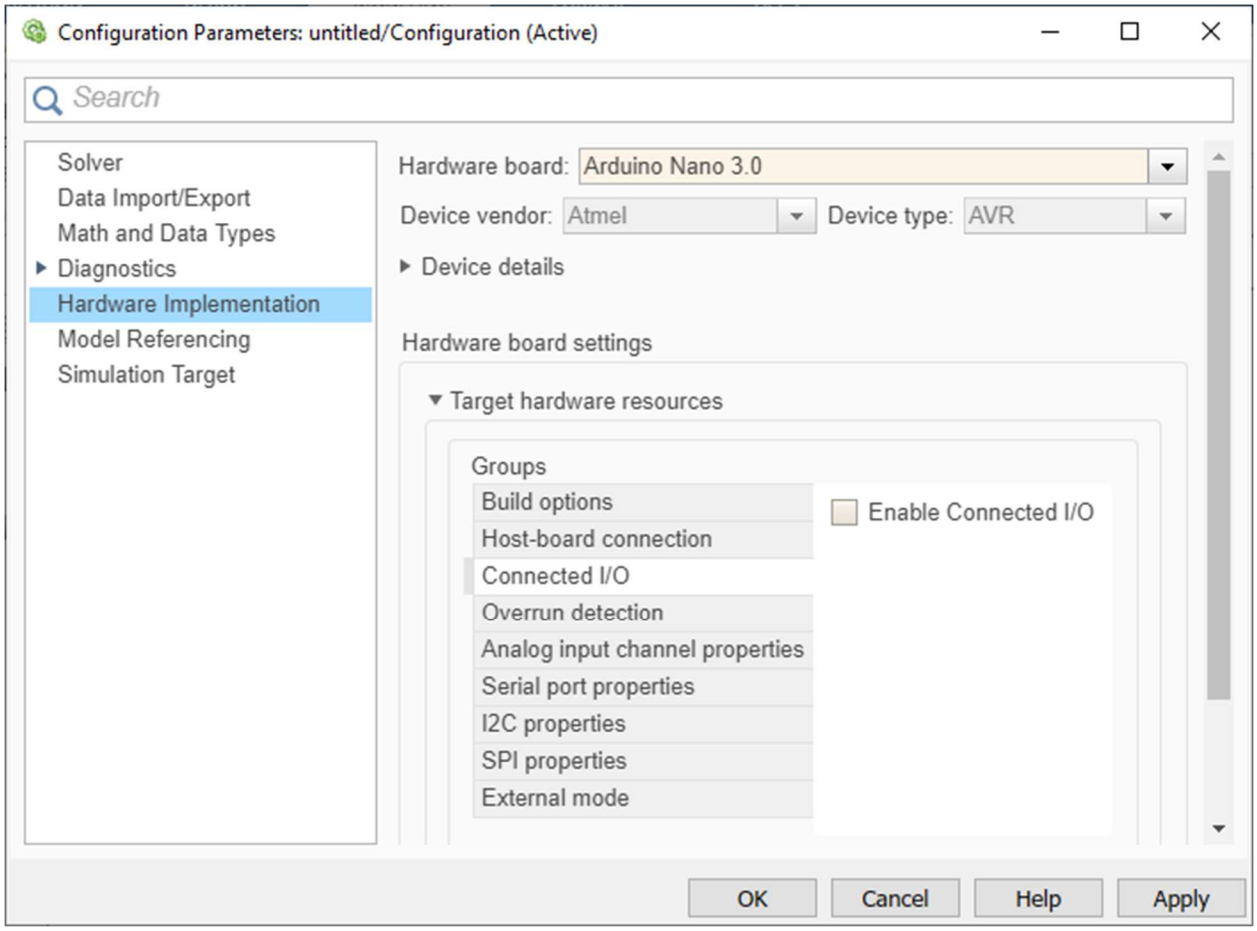
Simulink

1) Open MATLAB then open a new Simulink model

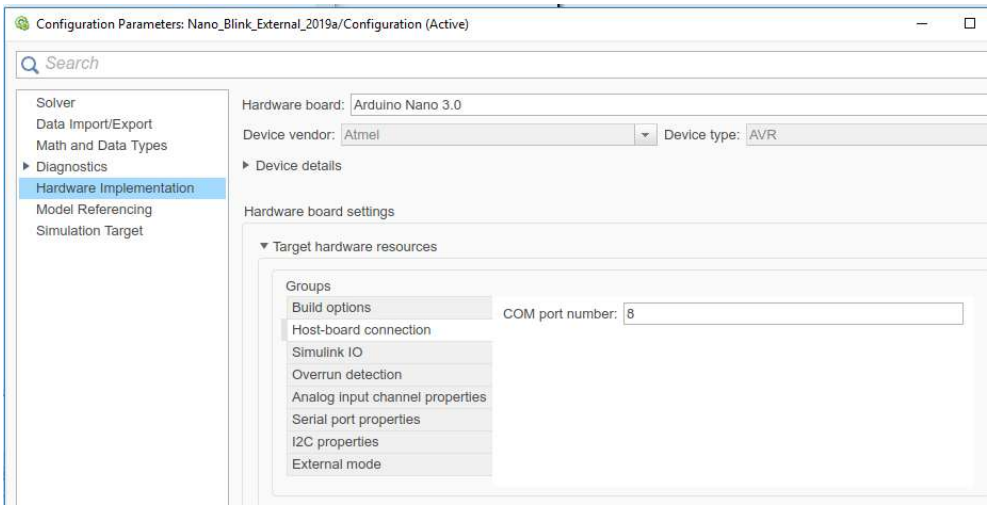


2) Find and assemble the required blocks.
   - Pulse Generator is under View->Library Browser->Simulink->Sources
   - Digital Output is under View->Library Browser->Simulink Support Package for Arduino Hardware->Common
3) Change the output pin to 13, which is connected to the onboard LED. This is done by double clicking on the Digital Output block.
4) Double click on the Pulse Generator to change the amplitude to 1, the Period to 2, and the pulse width to 50%.

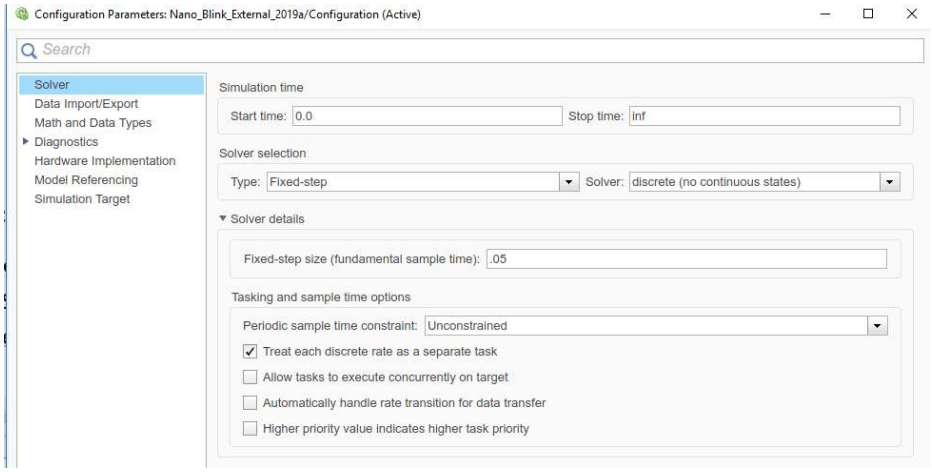
- Right click on the background of the Simulink window and select ‘Model Configuration Parameters’, or  in the top of the Simulink window.
- Under hardware Boad Select Nano 3.0
- Under Connected IO ensure Enable Connected IO is **unchecked**:



- Set COM port to the number you found in the device manager by opening “Target hardware resources” and then “Host-board connection”.



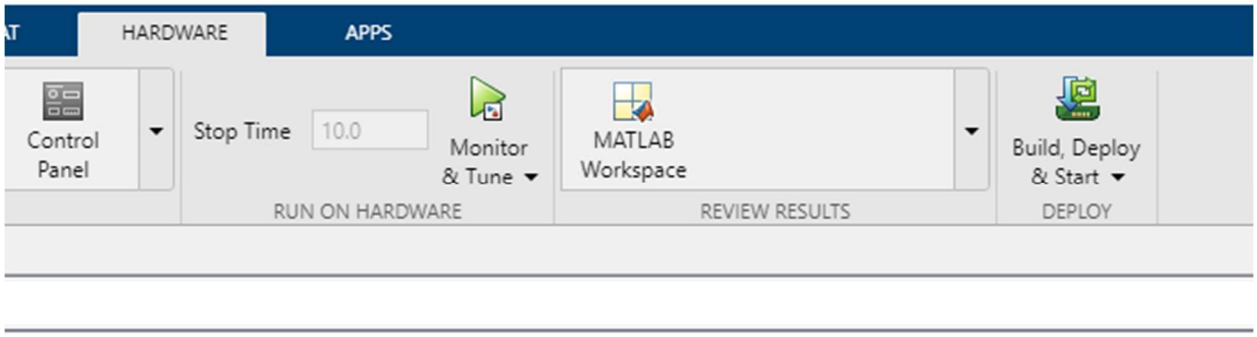
- Right click on the background of the Simulink window and select ‘Model Configuration Parameters’, or  in the top of the Simulink window.
- go to ‘Solver’ on the left hand menu and type ‘inf’ into the ‘Stop time’ spot so that the model will keep running. Expand “Solver details” and make sure the solver options are set to “Fixed-step” and “discrete (no continuous states)”. The “Fixed-step size” should be 0.05. This is how fast in seconds the control loop will execute. In this case, every 50 milliseconds it will execute the Simulink code.



Deploy to Hardware:

Compile and download the code to the board:

- Under the hardware tab select the Build, Deploy and Start button



This will download the code to the board and **will not interact or connect with it.** When "Deploy to Hardware" is used to downloaded to the board no information is passed between it and your computer unless your program specifically instructs the board to do so.

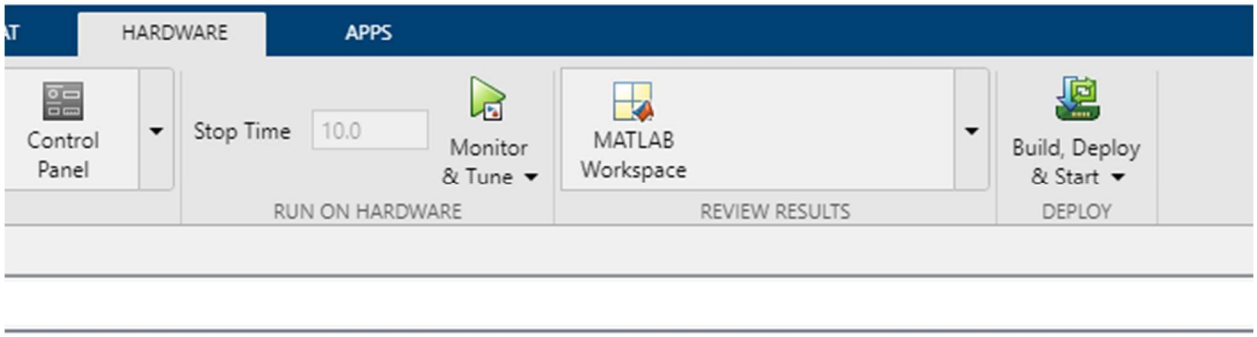
At this point the onboard LED connected to pin 13 should start blinking on and off at every second.

External Mode – communicating with the hardware

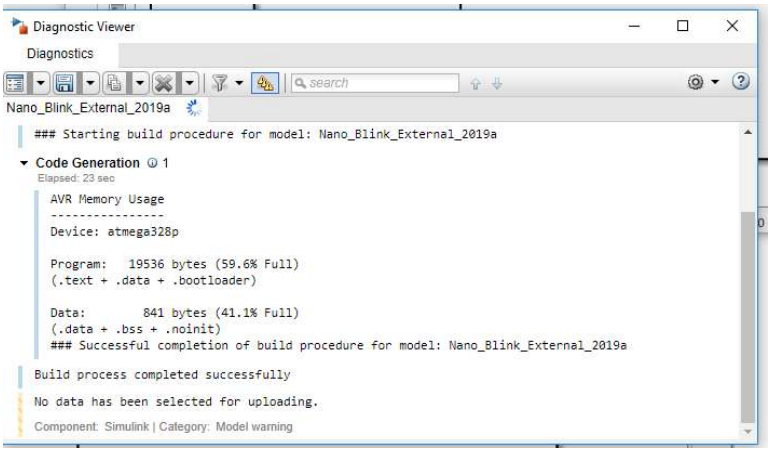
External mode established a data connection with the system so you can view data or interactively change parameters and see the result on the hardware in real time.

External Mode:

1.) External mode enabled by pressing the green "Monitor & Tune" button to download the code. When the green play button is used the hardware will create a connection with your computer to exchange data. This will allow you to modify parameters in your model and see the results on the hardware:

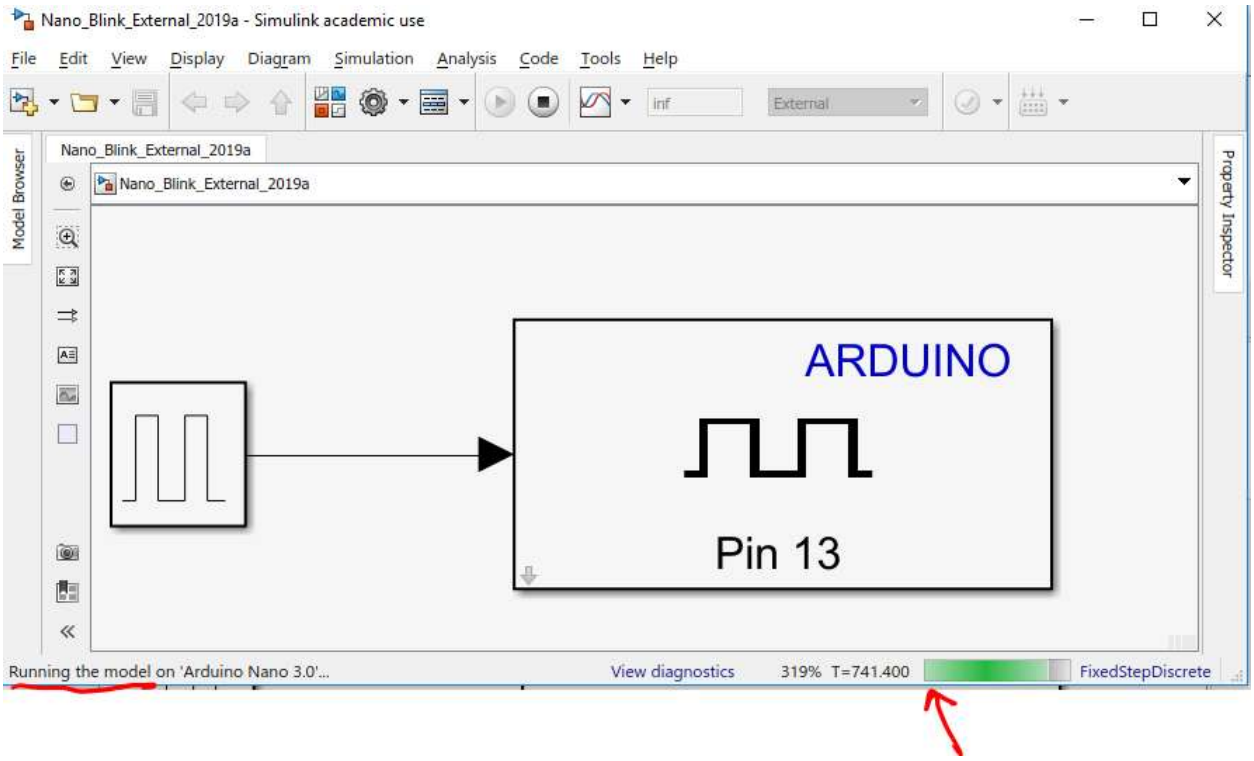


Notice as it is compiles how much larger the program size is:



Notice that to allow communication with the board it uses about 5 times as much program memory (additional 15.7kb) and 3.5 times as much data memory (additional 598 bytes). So the cost over interactive communication will be a challenge for larger programs. This is why an Arduino Mega will be preferable for larger programs.

After the code is downloaded it will connect to the hardware and you will see the time progressing in the bottom right corner:

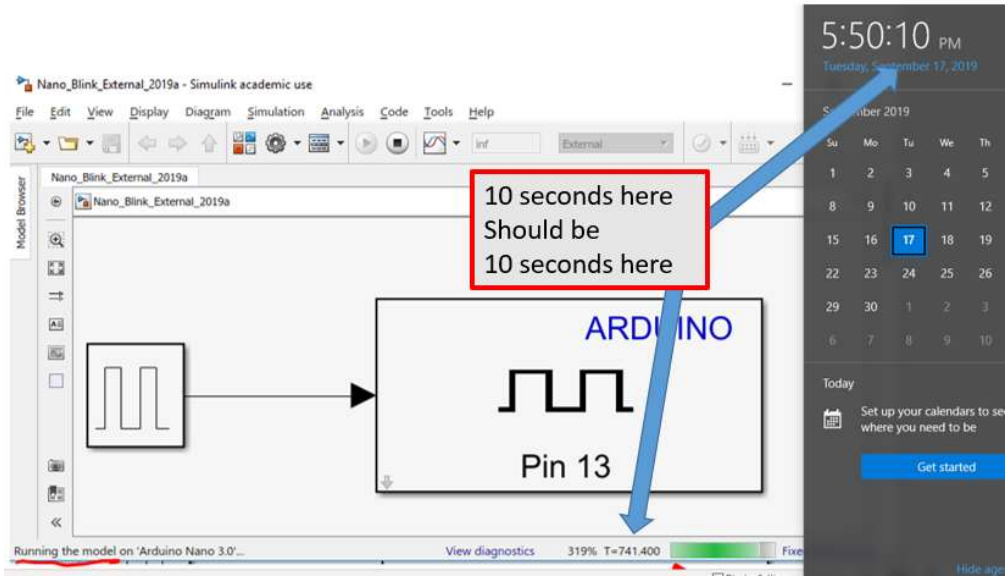


2.) Change parameters in your diagram and observe changes on the hardware. Double Click the Pulse Generator block and change the Period to 0.1 seconds. You should notice that the LED will begin blinking much faster.

3.) Use this file for new projects! For new hardware projects always use this file – it is configured to work with the hardware and you will not need to perform all the configuration steps again.

External Mode

- **External mode** and the green play button **allows bi-directional communication** to/from the application board to the PC, but can limit how fast your code can execute.
    - The "**Deploy to Hardware Button**" downloads the code and **does not connect or interact with it,** it can allow your code to run faster. You can use the **"Deploy to Hardware Button"** to download the code and NOT communicate to it.
- When external mode is selected and the green play button is pressed it downloads additional code to the board to allow this communication. This ***increase*** the program size (requires more memory)
- When you use external mode, you can change parameters while the system is running
    - However, this has an impact on the performance due to the communication bandwidth. For older versions (2015a), the fastest you can run in this mode and still meet your control loop time is approximately 30 milliseconds. Newer USB 3.0 ports and computers may run faster. Later versions of MATLAB can also run faster.
    - You can tell how fast you can run your code by examining the running clock in the Simulink diagram after you press the play button. When the code is running the green bar will be flashing and the time in seconds should be advancing:



If this clock advances as fast as the normal real word clock your code is "keeping up". If, for example, it can take 1 minute for the this timer to reach 10 seconds your code is not keeping up with real time and your requested sample time is to fast – it can not compute everything it needs to do fast enough.

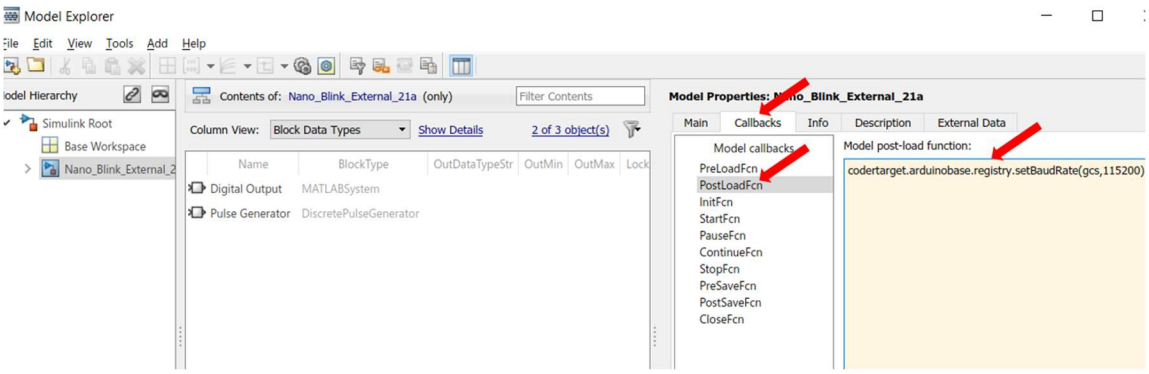
Stopping the Simulation:

When you are running the simulation in external mode you are connected to the device and are sending data and receiving data.

- Use the green “Play” button to download and run the code in external mode
- Use the square “Stop” button to stop. Since the board is no longer connected it will *continue doing the last action before it was disconnected*.

Debugging – cannot connect to device or cannot download code

- If you cannot connect to the device or there is an error when trying to download the code usually this means external mode was enabled and the simulation was stopped with the stop button or the USB cable was unplugged while being connected to the device. In this case you have a couple things to try that may successfully close the serial port
  - Disconnect and reconnect the USB Cable
  - Try the following commands at the Matlab command line (find connected instruments and close them):
    - newobjs=instrfindall
    - fclose(newobjs)
    - delete(newobjs)
  - Reset the device with the reset button or disconnect/connect the serial cable (this sometimes works)
  - If you are using verions of MATLAB before 2021a Make sure the command “codertarget.arduinobase.registry.setBaudRate(gcs,115200)” is located in the Model Explorer PostLoadFcn. If you are using versions before 2021a then you have to put this in every Simulink diagram:



  - Try compiling the code in a new folder that is empty
  - Close and restart Matlab (this usually works)
  - Log off your machine, log back in, then restart Matlab (this almost always works)
- Make sure the COM port is correct in the “Model Configuration Parameters”

Checkpoint:

To complete Lab 1 you will need to show the ability to control the blink of the LED by changing the parameters of the pulse block in external mode.

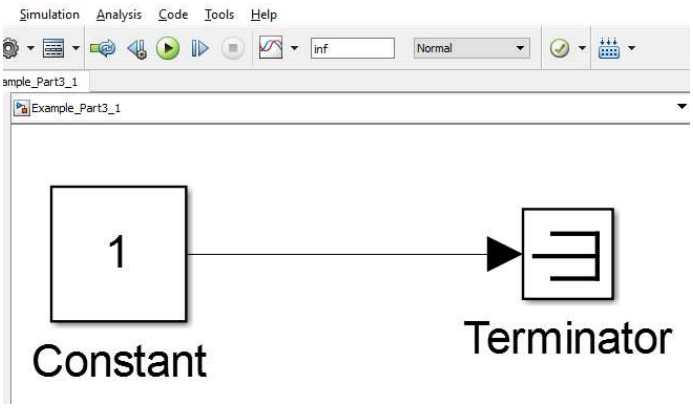
Part 3 (Optional): Evaluating code execution speed – how fast can I run my code

Objective:

- Determine how fast you code can be run on the target hardware using “Enable overrun detection” in Simulink.

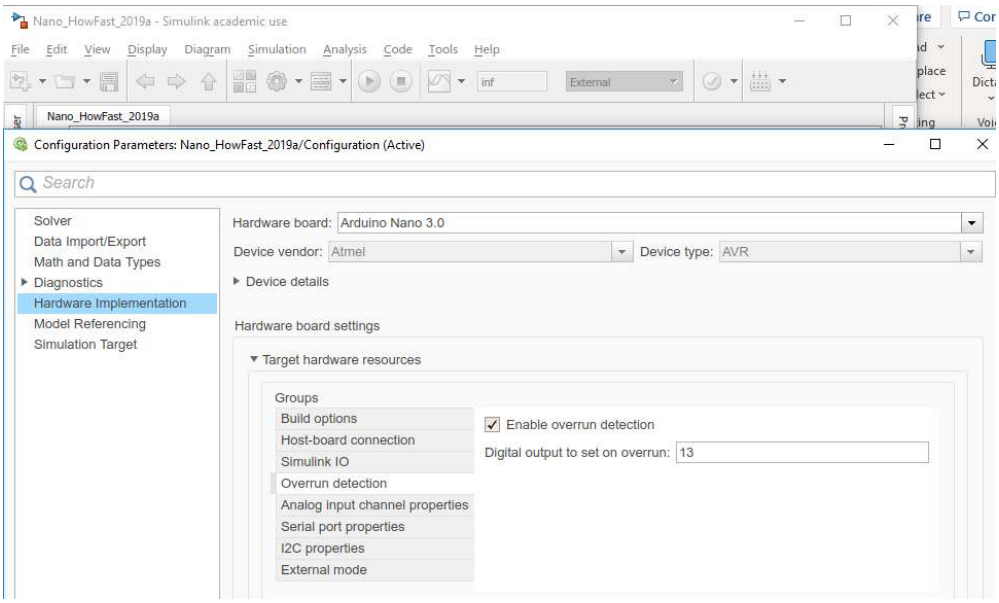
Simulink:

Build the following model:

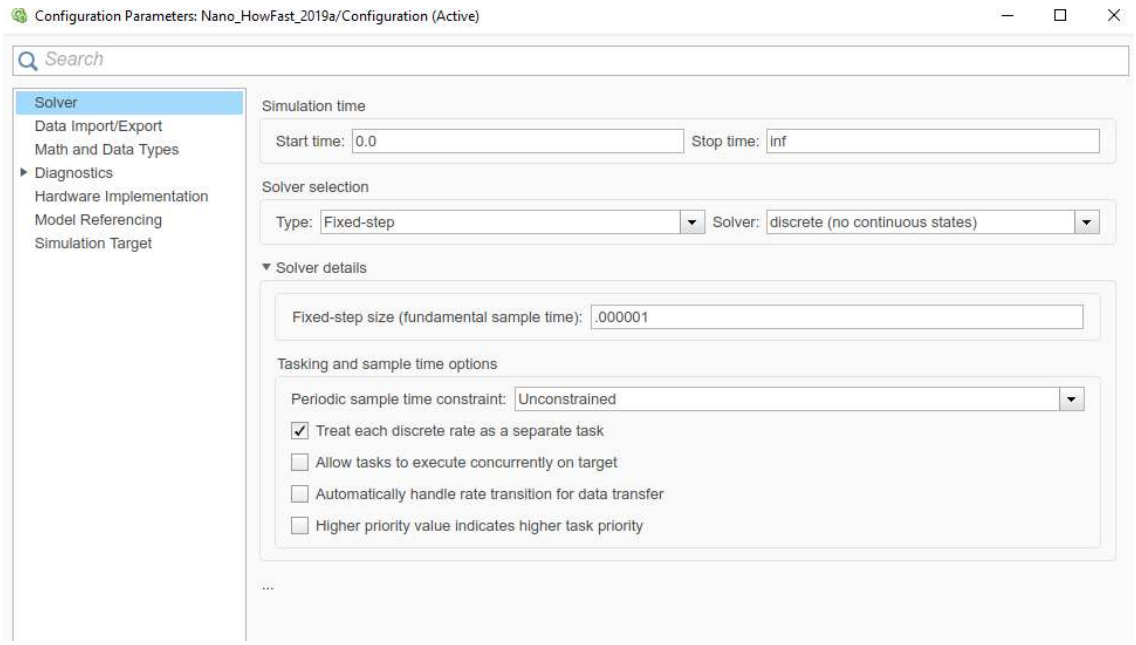


Note that this code essentially does nothing. The terminator is also in the commonly used blocks’. To determine how fast the hardware can execute a simple loop check the box “Enable overrun detection” and set the pin to 13.

In the “Model Configuration Parameters” under “Hardware Implementation”, “Target hardware resources”, “Overrun detection” make sure “enable overrun detection is checked” and the output pin is set to 13.



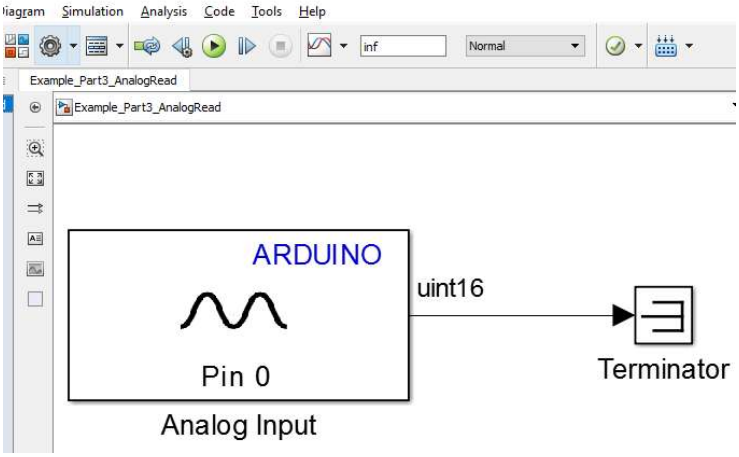
If the hardware cannot execute your code in the same time you specify the digital output, in this case the LED on pin 13 will light up indicating that you code cannot be executed in the time you have specified. You can use any digital output, but pin 13 is conveniently connected to an LED on our board. For the example code above, start with a sample time of 0.001 milliseconds (.000001 seconds). Download the code with the deploy to hardware button (Deploy to hardware mode, NOT external mode play button) and observe if LED 13 on the board lights up after downloading:



You should notice LED 13 will light up. This indicates the microprocessor can not run the code every 0. 000001 second, so you have to specify a larger sample time

Keep increasing the sample time (try .02 milliseconds, etc.) until LED 13 does not light up after downloading. This is the fastest you the board can execute your code.

Using this procedure try and see how fast you can run the following, start at a sample time of 0.001 milliseconds.



- Double click the analog input block and make sure the sample time is set to -1. This means that the analog input will read every time the code loop

In this fashion, you can determine how fast your code, or any part of your code takes to run on your hardware.

Part 4: Troubleshooting

An error appears when trying to run the file similar to "build procedure aborted", "make command returned an error…"

- Verify **Simulink** Support for Arduino Hardware is installed and appears in the Simulink Library browser
- Ensure that the Matlab directory in **your documents**/MATLAB folder or similar location (NOT the programs folder)
- Open MATLAB Before opening the Simulink file (sometimes if you open MATLAB directly with the file it will open in the programs folder instead of your documents folder)

Error appears when trying to open the Simulink file:

- Ensure your downloaded file does not have parenthesis in the file name for example:
  - Bad file name "Lab_0_Step_Response (1).slx" This happens when you download the file several times, it will append (1) to the filename and Matlab files cannot have parenthesis in the names.

Simulink Support for Arduino Hardware does not appear in the Simulink library browser:

- Restart Matlab and Reinstall **Simulink** Support for Arduino Hardware

Device is not recognized in device manager when it is plugged in

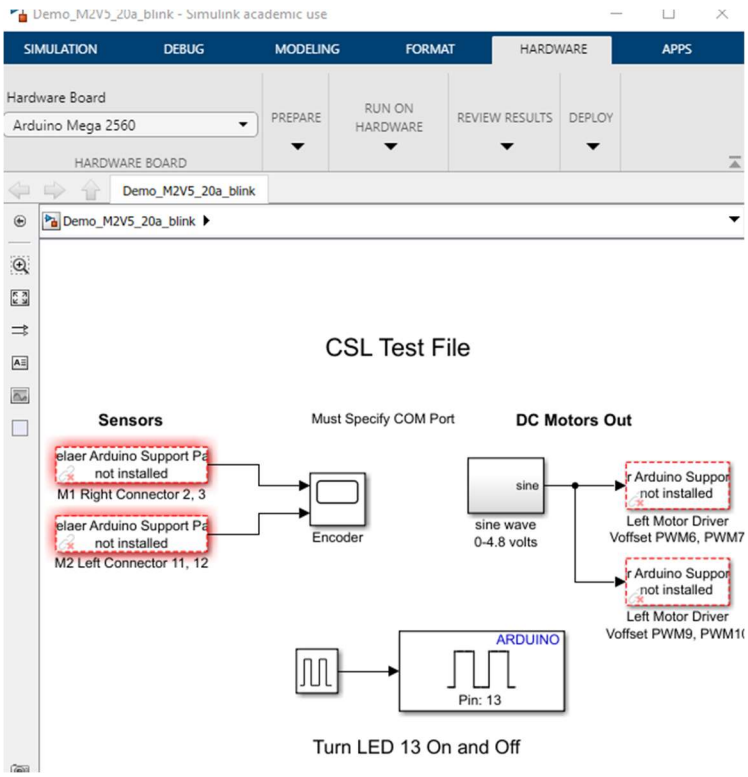
- Open device manager in Windows, find the device, right click and choose Update device drivers or install the drivers via “manage add ons menu”

File is not compiling, a long list or errors appears.

- Verify that RASPLib is installed and that verify Rensselaer Arduino Support Package appears in the file browser.
- Verify that RASPLib folder and startup.m are located in
  - Documents\MATLAB\RASPlib
  - Documents\MATLAB\startup.m

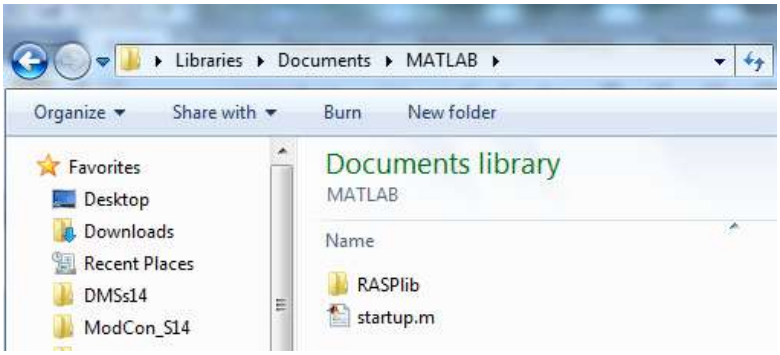
In addition to the above, make sure that MATLAB working directory is not Windows/system32 or some other system folder that requires admin rights to edit. To be safe, set the working folder to your documents/MATLAB

Red block in Simulink diagram:



The Arduino Support package or RASPLib are not properly installed. Follow the pre-lab instruction carefully. The most common error is not placing startup.m and RASPLib in the correct location:

Exactly as shown below:

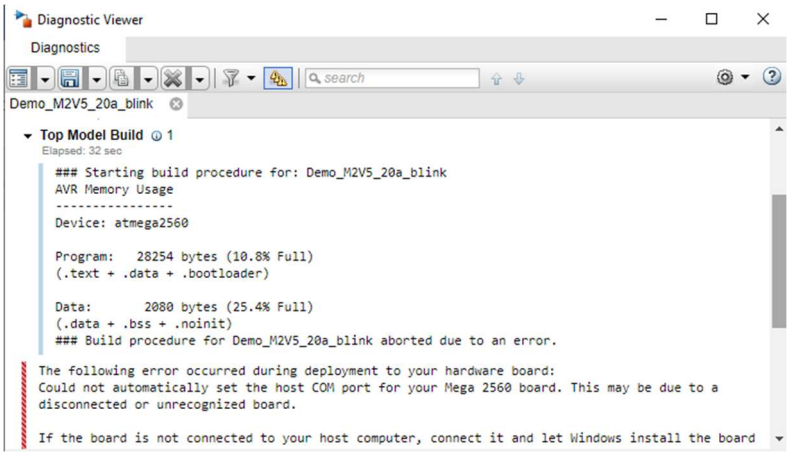


- YES C:\Users\hurstj\Documents\MATLAB\RASPlib
- YES C:\Users\hurstj\Documents\MATLAB\startup.m

- NOT C:\Users\hurstj\Documents\MATLAB\RASPlib\RASPlib
- NOT C:\Users\hurstj\Documents\MATLAB\RASPlib\startup.m
- NOT C:\Users\hurstj\Documents\MATLAB\hurstj01-RASplib-366dd1e\RASPLib

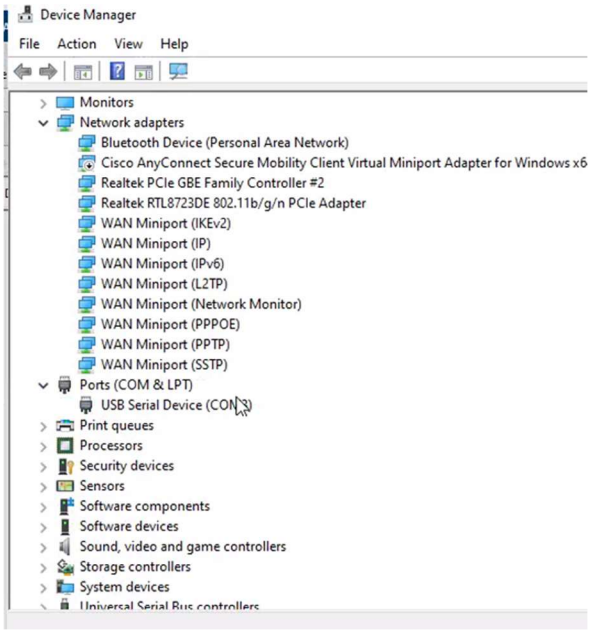
Fix and restart matlab.

Error about board not connected to computer

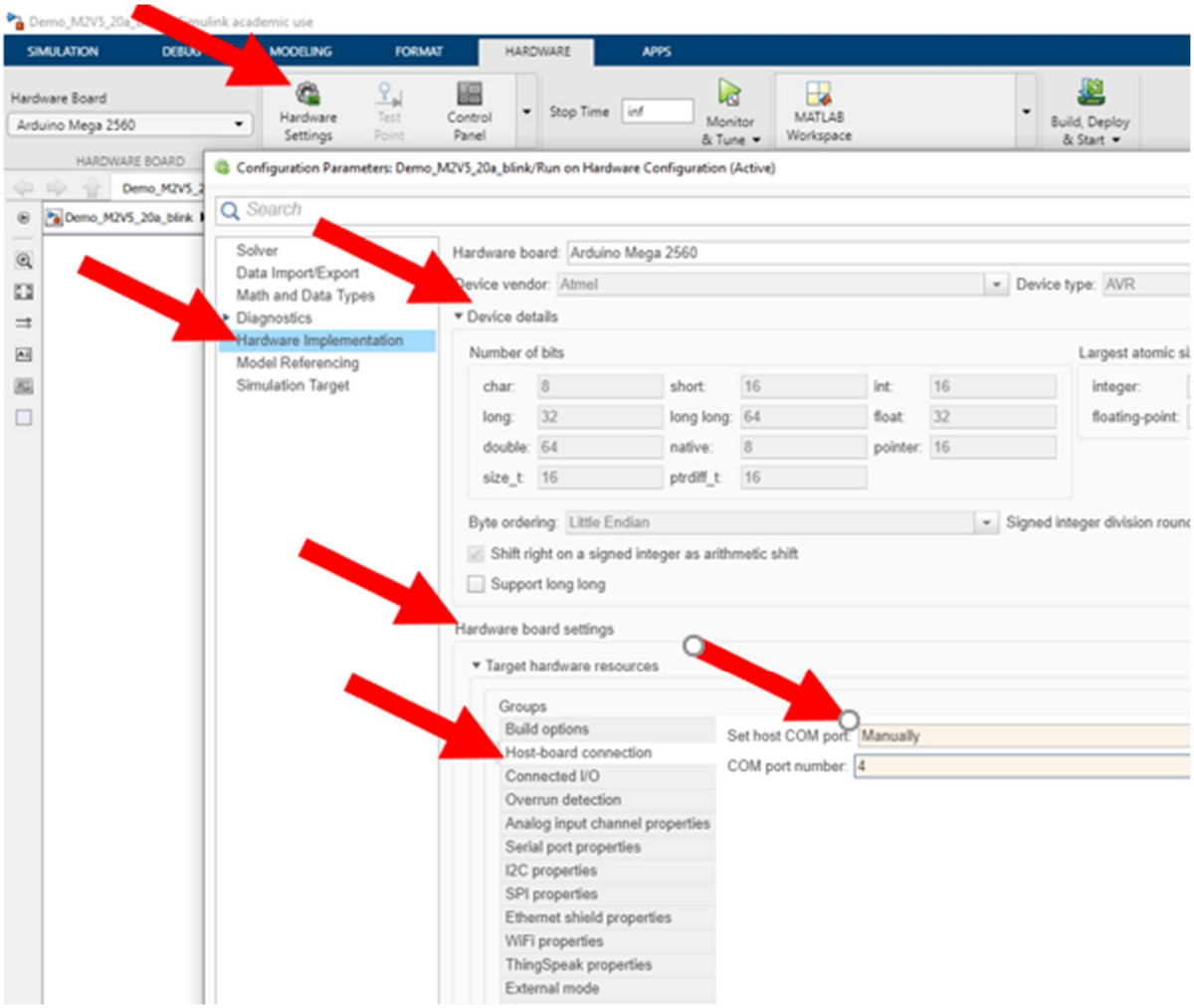


First Check that board is recognized by your computer:

Press windows key, type “device manager”, scroll down to Ports, and expand. Connect the USB cable to the computer and see what device it shows. It should read “USB-SERIAL CH340”. If it does not right click on the device and select “update driver” and search automatically. After this is done disconnect and reconnect the USB cable. Or install the drivers in the CH341SERexe.zip.

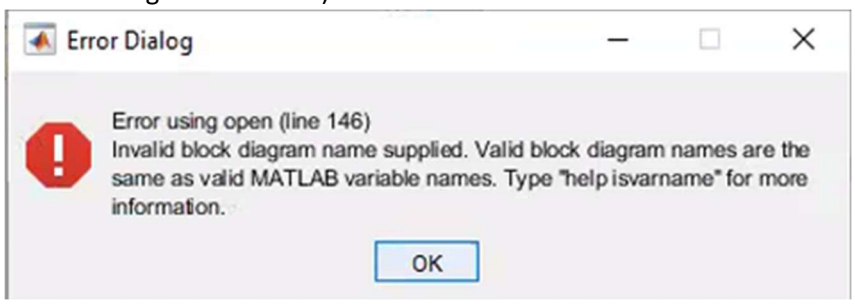


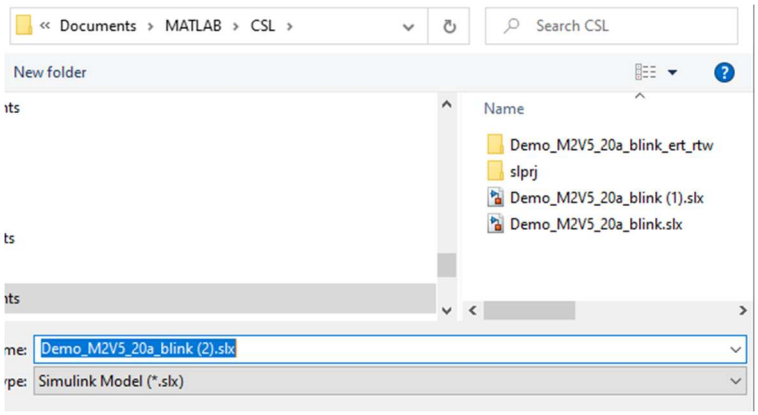
Verify the Simulink diagram is using the correct COM port number. Then in the Simulink diagram click “Hardware Settings”, “Hardware Implementation”, expand the “Device Details”, “Hardware board settings”, “Target hardware Resources”, “Host Board connection”, then “set host Com port” to “Manually” and specify the com port number.



Error Dialog. Error using open… Simulink diagram file name has parenthesis in it

- Make sure the Simulink diagram does not have parenthesis in it (which can be caused by downloading the file twice).





Questions:

How can you determine how fast you are able run code in External mode?

How can you determine how fast you are able to run run in Deploy to hardware (normal) mode?

What is the difference between Deploy to hardware and External mode?

What can happen if the path to your compile folder is too long?

What can happen is there are parenthesis in the file name?