

## Steps for Arduino Installation

**Attention:** Please note that this installation 'readme' file is valid and applicable only for 32-bit Operating systems running MATLAB 2010a or above with Instrument Control Toolbox, Real-Time Workshop and Real-Time Workshop with Embedded Coder.

- 1) Create a folder in C:\ called Arduino and Download 3 folders from internet. Place/Download into the folder on C:\Arduino. (Need to log-in into your mathworks account. If you don't have, create one for free).

- a. Arduino Target for SIMULINK from Mathworks:

<http://www.mathworks.com/academia/arduino-software/arduino-simulink.html>

Extract the zipped folder into a folder named 'Arduino\_Target'.

- b. Arduino Interface for MATLAB:

<http://www.mathworks.com/academia/arduino-software/arduino-matlab.html#>

Extract the zipped folder into a folder named 'arduino\_ml'.

- c. Arduino Integrated development environment: (Many versions are available. Recommended would be the version 'arduino-0018'. This downloaded folder includes the USB (FTDI) virtual COM port drivers required for Windows XP).

<http://arduino.cc/en/Main/Software>

- 2) Start MATLAB and change the current working directory to C:\Arduino.
- 3) Connect the Arduino board. The operating system (Windows Vista and Windows 7) automatically configures the FTDI drivers and successfully assigns a COM port. You would be notified of the COM port number assigned. Note down the COM port #.
- 4) Run the executable file 'Arduino' from 'arduino-0018' folder and under tools select your COM port. At this juncture, your Arduino board is ready to be programmed in command line. You can run your program and compile into the board. For further instructions, please open the 'Readme' in the 'arduino\_ml' folder.

- 5) Arduino Simulink Installatoin:

- a. Check the current directory. Change it to C:\Arduino\Arduino\_Target.

- b. In the Matlab command window, type:

```
>>addpath(fullfile(pwd, 'arduino'),fullfile(pwd, 'blocks'),fullfile(pwd, 'demos'))
```

```
>>savepath
```

If you encounter an error after savepath, then your operating system is not allowing any changes to be made to a Matlab file 'pathdef.m'. To overcome this problem, right click on the Matlab R2010a folder, click on properties, select security, click edit and then check the box 'Full Control'. Try 'savepath' again. If no success, contact the administrator.

c. Type:

```
>> sl_refresh_customizations
>> arduino.Prefs.setArduinoPath('c:\Arduino\arduino-0018')
>> arduino.Prefs.setMcu('atmega328p') % or atmega168
>> comPorts=arduino.Prefs.searchForComPort;
>> arduino.Prefs.setComPort(comPorts{1});
```

d. Select the compiler for Matlab. Type:

```
>> mex('-setup');
```

The compilers would be listed, choose the Matlab Lcc win32 compiler.

e. Type:

```
>> demo_arduino_blink
```

Once the file opens, build the file. Use Ctrl+B and observe the output. During the run, the TX and RX led's blink when the hex file is downloading into the target.

Instructions for 64-bit operating systems will be updated when a suitable compiler is found for MATLAB.