|  |  |
| --- | --- |
|  | **ADS1115 and Raspberry Pi Connections:**  VDD – 3.3v GND – GND SDA – SDA SCL – SCL  **Step 1: Enable Raspberry Pi I2C interface**  To enable the I2C, from the terminal, run; |
| **sudo raspi-config** |
| **Step 2: Update the Raspberry pi** |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  |  |
| **sudo apt-get update sudo apt-get upgrade** |
| **Step 3: Install the Adafruit ADS1115 library for ADC** |
| **cd ~** |
|  |
| **sudo apt-get install build-essential python-dev python-smbus git** |
|  |
| **git clone** [**https://github.com/adafruit/Adafruit\_Python\_ADS1x15.git**](https://github.com/adafruit/Adafruit_Python_ADS1x15.git) |
|  |
| **cd Adafruit\_Python\_ADS1x15 sudo python setup.py install** |
| . |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | **Step 4: Test the library and 12C communication.** |
| **cd examples** |
|  |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | **python simpletest.py** |
| Test the library and 12C communication  If an error occurs, check to ensure the ADC is well connected to the PI and I2C communication is enabled on the Pi.  **Step 5: Install *Matplotlib*** |
| **sudo apt-get install python-matplotlib** |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | Install Matplotlib  **Step6: Install the *Drawnow* python module** |
| **sudo apt-get install python-pip12** |
|  |
| **sudo pip install drawnow** |
|  |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | Install the Drawnow python module  With all the dependencies installed, we are now ready to write the code.  With the monitor as the interface **open a new python file**. You can call it any name you want, but I will call it scope.py. |
| **sudo nano scope.py** |
|  |
| **import time**  **import matplotlib.pyplot as plt from drawnow import \***  **import Adafruit\_ADS1x15** |
|  |
| **adc = Adafruit\_ADS1x15.ADS1115()** |
| **GAIN = 1** |
|  |
| **Val = [ ] cnt = 0** |
|  |
| **plt.ion()** |
| . |
| **adc.start\_adc(0, gain=GAIN)** |
|  |
| **plt.ylim(-5000,5000) plt.title('Osciloscope') plt.grid(True)**  **plt.ylabel('ADC outputs')** |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | **plt.plot(val, 'ro-', label='lux')**  **plt.legend(loc='lower right')** |
|  |
| **value = adc.get\_last\_result()** |
|  |
| **print('Channel 0: {0}'.format(value)) time.sleep(0.5) val.append(int(value))** |
|  |
| **drawnow(makeFig)** |
|  |
| **cnt = cnt+1**  **if(cnt>50): val.pop(0)** |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | Save the code and run using; |
| **sudo python scope.py** |
| If you used a different name other than scope.py, don’t forget to change this to match. |
| **Import warnings**  **import matplotlib.cbook**  **warnings.filterwarnings(“ignore”, category=matplotlib.cbook.mplDeprecation)** |
| ADC data printed on terminal |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | Oscilloscope window on Raspberry pi  **Code:**  import time  import matplotlib.pyplot as plt #import numpy  from drawnow import \*  # Import the ADS1x15 module. import Adafruit\_ADS1x15  # Create an ADS1115 ADC (16-bit) instance. adc = Adafruit\_ADS1x15.ADS1115()  GAIN = 1  val = [ ] cnt = 0 plt.ion()  # Start continuous ADC conversions on channel 0 using the previous gain value. adc.start\_adc(0, gain=GAIN)  print('Reading ADS1x15 channel 0') #create the figure function  def makeFig(): plt.ylim(-5000,5000)  plt.title('Osciloscope') plt.grid(True) plt.ylabel('ADC outputs')  plt.plot(val, 'ro-', label='Channel 0') plt.legend(loc='lower right') |

T.Y.B.Sc. (I.T.) 2019

|  |  |
| --- | --- |
|  | while (True):  # Read the last ADC conversion value and print it out. value = adc.get\_last\_result()  print('Channel 0: {0}'.format(value)) # Sleep for half a second. time.sleep(0.5) val.append(int(value)) drawnow(makeFig) plt.pause(.000001)  cnt = cnt+1 if(cnt>50):  val.pop(0) |