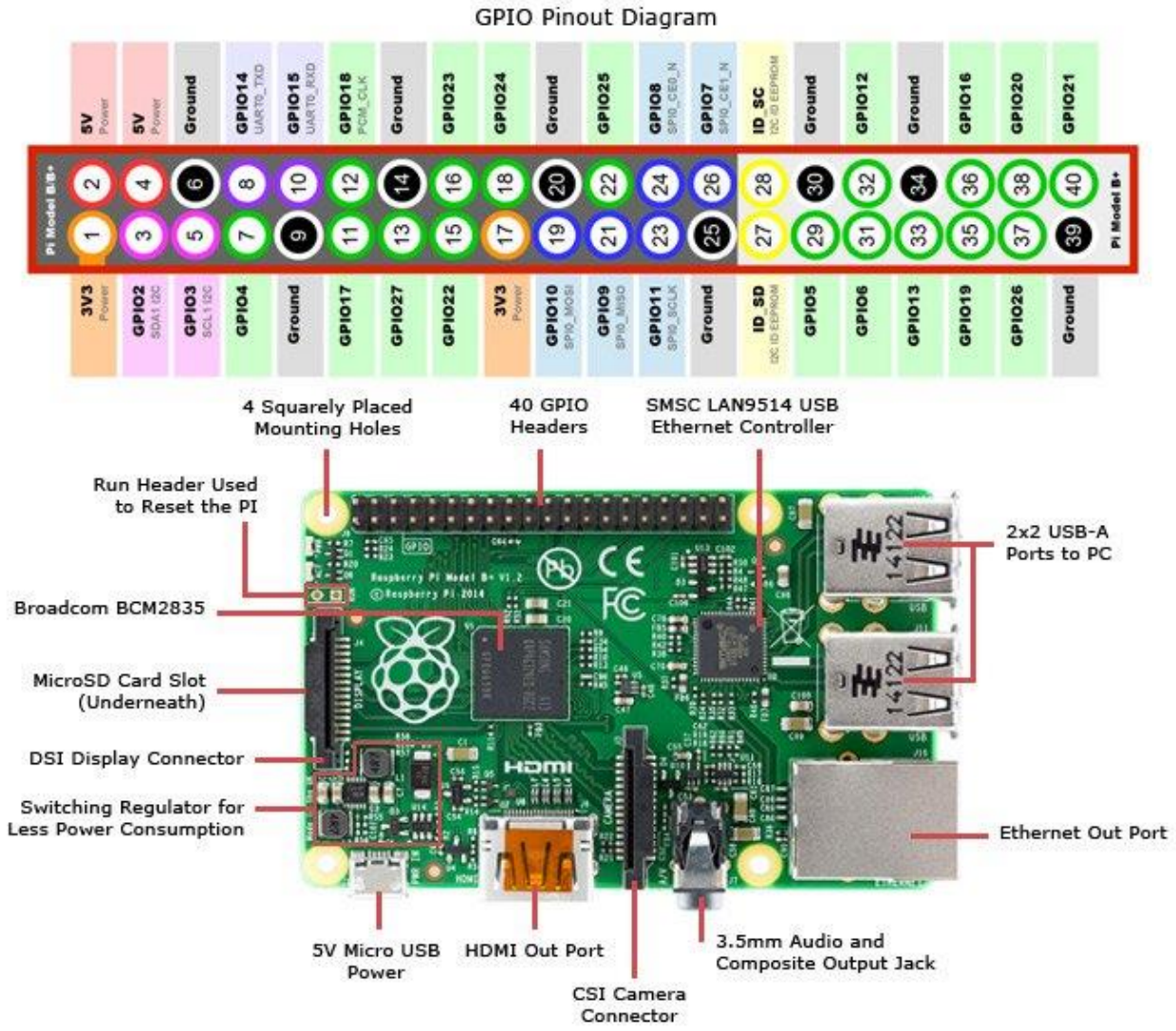
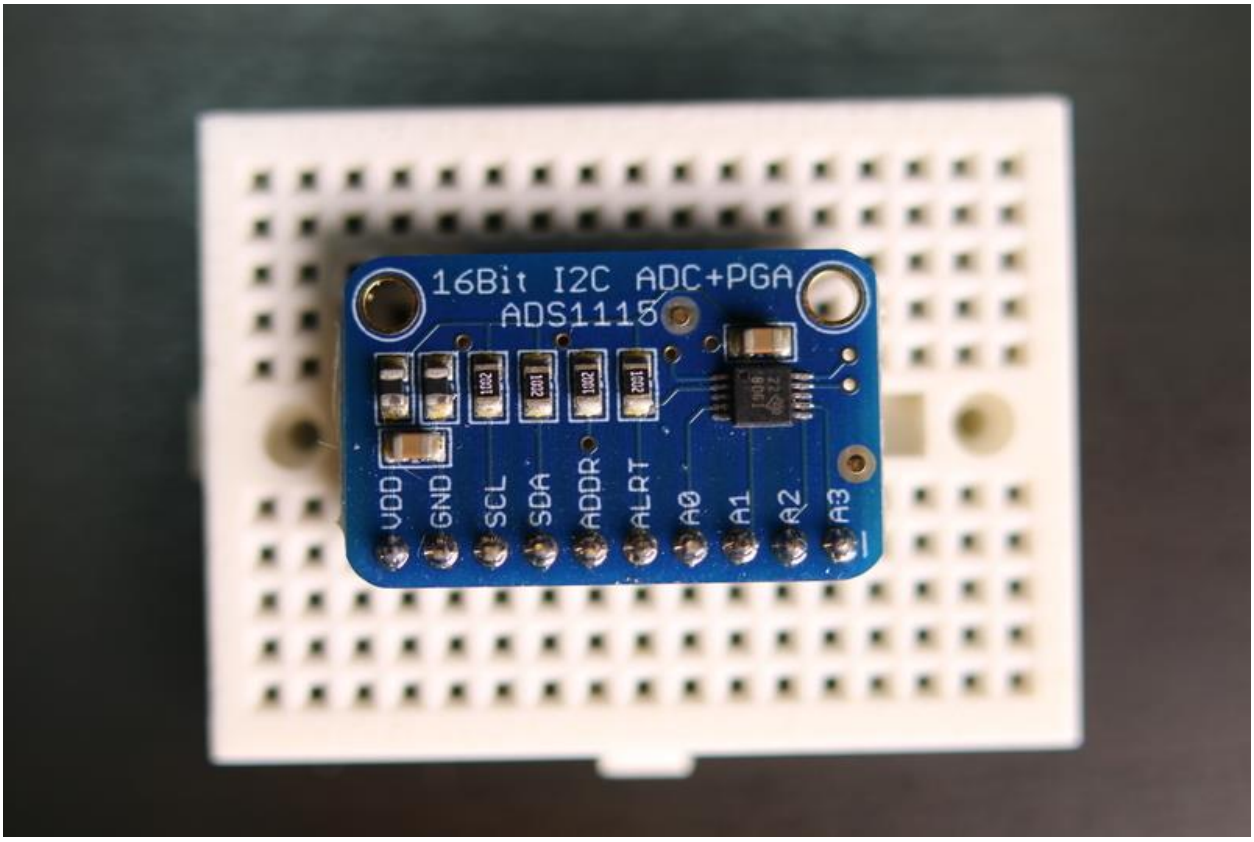
## Analog to Digital converter with Raspberry Pi

ADS1115 is and 16bit Analog  to Digital convertor with four input pins over I2C (I2C have to be enabled in Raspberry Pi through raspi-config->Interfacing options). Analog measurements such as Voltage and currents have to converted to digital signals in order to be able to read by RaspberryPi. Thus, we use ADS1115 in order to connect Voltage, current and soil moisture sensors to the RaspberryPi.




In this respect, we should connect ADS1115 pins as follows in order to be able to read measurements:

VDD-> 3.3V pin in RaspberryPi
GND-> Ground pin in RaspberryPi
SCL -> SCL(Serial clock line) pin in RaspberryPi
SDA -> SDA(Serial Data line) pin in RaspberryPi
ADDR-> ground pin
ALRT -> unconnected
A0-3 -> analog lines from sensors




After wiring ADS1115 to the Raspberry Pi, we can use python library to read data from ADS1115. Hence, we can install Adafruit python library as follows:

```
sudo apt-get update
sudo apt-get install build-essential python-dev python-smbus python-pip
sudo pip install adafruit-ads1x15
```

We can also install Adafruit Python library by pip installation as an alternative **(Optional):**

```
sudo apt-get update
sudo apt-get -y install build-essential python-dev python-smbus git
cd ~
git clone https://github.com/adafruit/Adafruit_Python_ADS1x15.git
cd Adafruit_Python_ADS1x15
sudo python setup.py install
```

After installation of Adafruit Python library from Github then we can change directory to the adafruit_Python_ADS1x15 in order to use python example codes.

```
cd Adafruit_Python_ADS1x15/examples
```

Then we can use simpletest.py file in order to read four inputs of ADS1115 by only running the following command:

```
cd Adafruit_Python_ADS1x15/examples
```

In the simpletest.py file we can set the GAIN value in order to limit the voltage range reading in order to read precise value. Indeed, we should tune the GAIN value based on the voltage variance in the ADS1115 input pin. Therefore, if the voltage on the ADS1115 input will change 0-4V then we should use GAIN=1.

```
#  - 2/3 = +/-6.144V
#  -   1 = +/-4.096V
#  -   2 = +/-2.048V
#  -   4 = +/-1.024V
#  -   8 = +/-0.512V
#  -  16 = +/-0.256V
```

The value for each channel is the ADC value for that channel. This is a number that ranges from -32768 to 32767 on the 16-bit ADS1115. A value of 0 means the signal is at a ground (reference) level, 32767 means it's at or higher than the maximum voltage value for the current gain (4.096V by default), and -32768 means it's a negative voltage below the reference voltage. Therefore, we can use the following formula in order to calculate converted digital measurement.

(ADS1115 value*Gain voltage)/32768
Voltage=(ADS1115 value *4.096)/32768

## Soil Moisture reading by ADS1115

```
import time
#import Adafruit library for ADS1115 analog to digital convertor
import Adafruit_ADS1x15
#Create an ADS1115 instance
adc=Adafruit_ADS1x15.ADS1115()
#

# Choose a gain of 1 for reading voltages from 0 to 4.09V.
# Or pick a different gain to change the range of voltages that are read:
#  - 2/3 = +/-6.144V
#  -   1 = +/-4.096V
#  -   2 = +/-2.048V
#  -   4 = +/-1.024V
#  -   8 = +/-0.512V
#  -  16 = +/-0.256V
# See table 3 in the ADS1015/ADS1115 datasheet for more info on gain.
#maximum digital output of ADC is 32767
#Define ADC input line and the gain for Voltage sensor module
soilInput=0
soilGain=2/3  #when soil sensor connected to 5V

#Please enter voltage when soil sensor is 100% in water
soilSensor_maxVoltage=1.6
#Please enter voltage when soil sensor is dry
soilSensor_DryVoltage=4.8
```

```
        #This section convert voltage read by  ADC to percentage (0-100%)
        #for more information read Calculating Scale and Offset in  https://racelogic.support [1]
        #calculate voltage scale or gradient of line
        #dy/dx=(100-0)/(soilSensor_maxVoltage-soilSensor_DryVoltage)
        scale=100/(soilSensor_maxVoltage-soilSensor_DryVoltage)
        #calculate ofsset: Y=mX+c -> percent=(scale*Voltage)+offset
        offset=(-scale*soilSensor_DryVoltage)
        while True:
          # Read ADC converted value from Analog Input for Voltage
          soil_adc=adc.read_adc(soilInput,soilGain)
          #Calculate actual voltage from ADC data
          soilMoistVoltage=((soil_adc*6.144)/32767)
          #Calculate soil moisture percentage
          soilMoist_percentage=(scale*soilMoistVoltage)+offset
          print('| {moisture} | {soil}| {soilPercent:.2f}%|
'.format(moisture=soil_adc,soil=soilMoistVoltage,soilPercent=soilMoist_percentage))
        #   time.sleep(60)
          time.sleep(1)
```

## References


[1] Racelogic, "Calculating Scale and Offset," 13 March 2018. [Online]. Available: https://racelogic.support/02VBOX_Motorsport/Video_Data_Loggers/Video_VBOX_Range/Video_VBOX_-_User_manual/24_-_Calculating_Scale_and_Offset. [Accessed 2018].