Microcontroller based Multimeter Design

Supervisor: Dr.Saroj Mondal

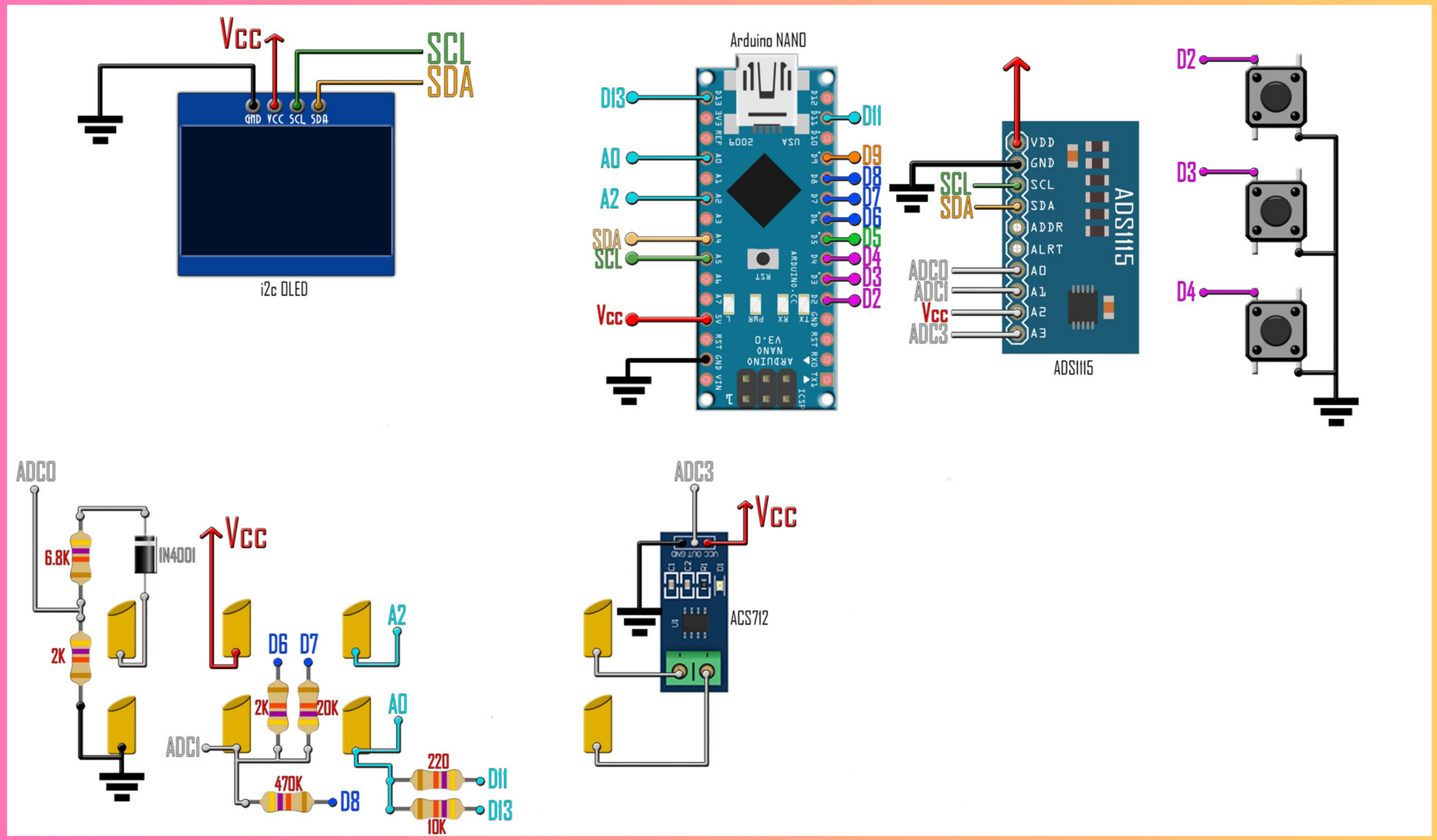
What is our Aim ?

- Designing a Micro controller based multimeter (using Arduino)
- Which can work like a usual multimeter in our lab with components we have in lab (minimum cost).
- Which can measure capacitance which is odd compared to the digital multimeter in Lab.
- Transfer's measured values through HC-05 Blutooth module and displays them on mobile device.

Specifications of the Multimeter

- Measures the Voltage in the range of 0 to 20 Volts.
  - Measures the Current in the range of 1mA to 5A
  - Measures the Resistance in the range (1k to 1M) Ω
- Measures the Capacitance in the range of 10pF – 1mF
- Shows the Measured values in Mobile android app

Schematic

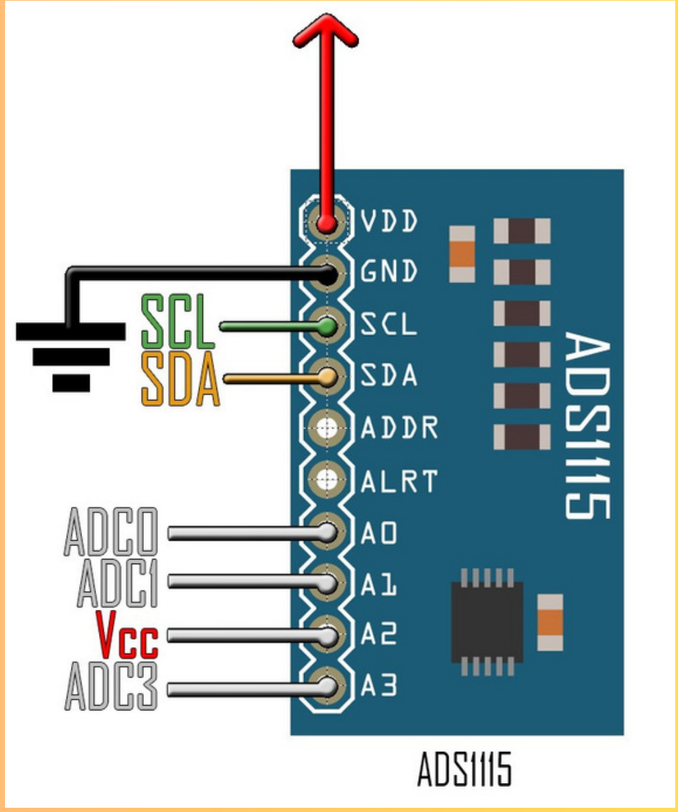

Vcc
SCL
SDA
GND VCC SCL SDA
i2c OLED
Arduino NANO
D13
A0
A2
SDA
SCL
Vcc
D11
D9
D8
D7
D6
D5
D4
D3
D2
VDD
GND
SCL
SDA
ADDR
ALRT
A0
A1
A2
A3
ADS1115
ADC0
ADC1
Vcc
ADC3
D2
D3
D4
ADC0
6.8K
IN4001
Vcc
2K
D6
D7
A2
2K
20K
A0
ADC1
470K
D8
220
D11
D13
10K
ADC3
Vcc
ACS712

1.Measuring Voltage

1. Connect the sensor to Arduino's VCC, GND, SDA, and SCL pins.

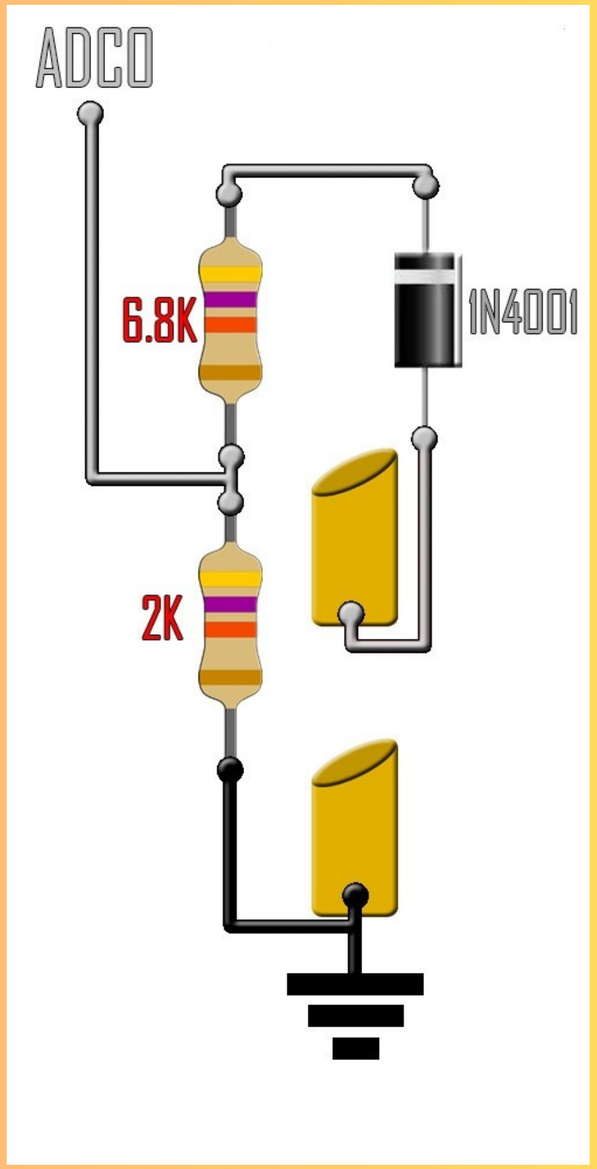
2.ADS1115, the analog signal to be measured is connected to one of the device's four input channels. The device then converts the signal to a digital value, which can be read by a microcontroller over an I2C interface.

4. Repeat the measurement steps in a loop or desired interval for continuous voltage monitoring.



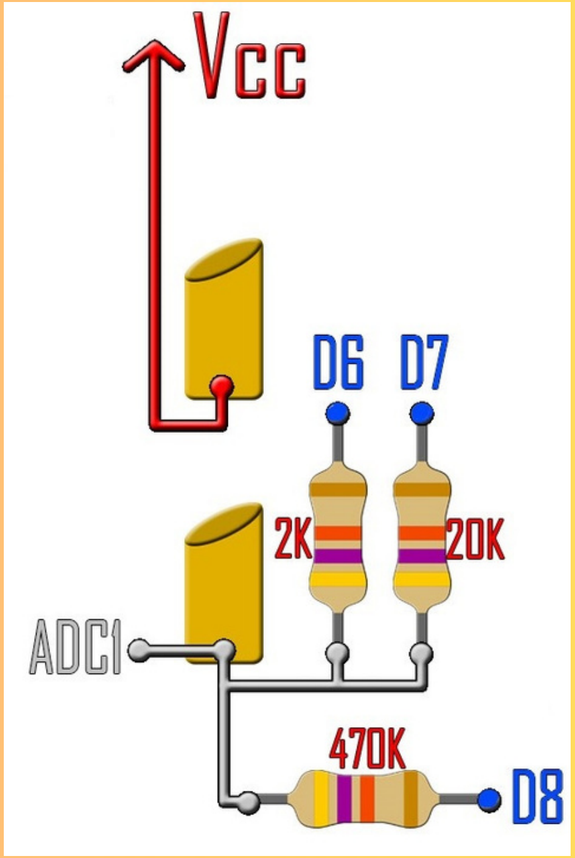
- ADS115 can measure voltage ranging 0-5 Volts
- We are designing it to mesure upto 20 Volts using voltage divider input
- If we apply 20 V at input the drop on 2K resistor will be nearly 5V so max voltage that ADC0 probe can measure indirectly is 5V.
- We use diode to block negative voltage (0.27V).

NOTE: It cannot measure negative voltage. So we came up with another idea to measure negative voltage in later section of the presentation

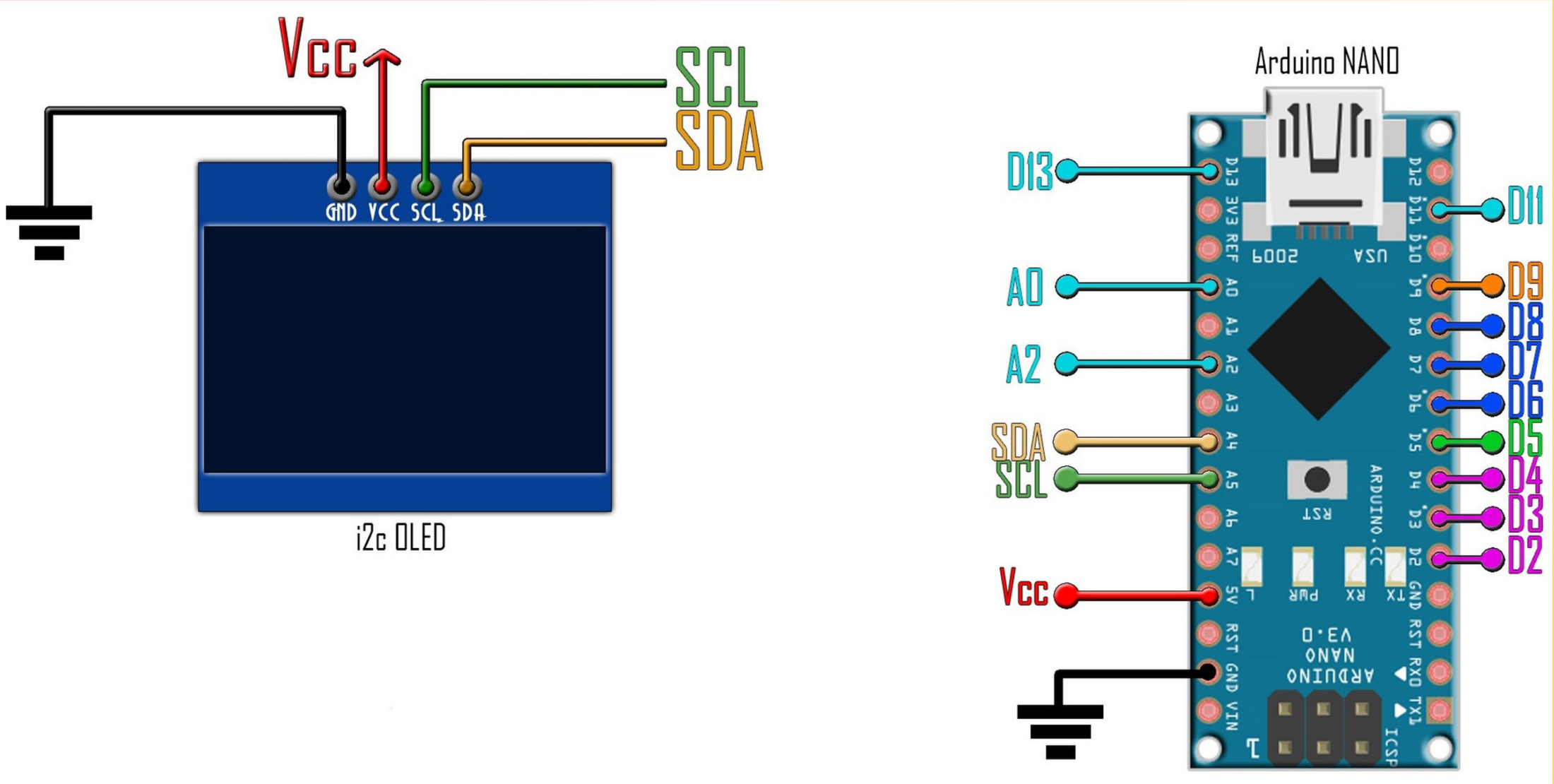


2.Measuring Resistance:

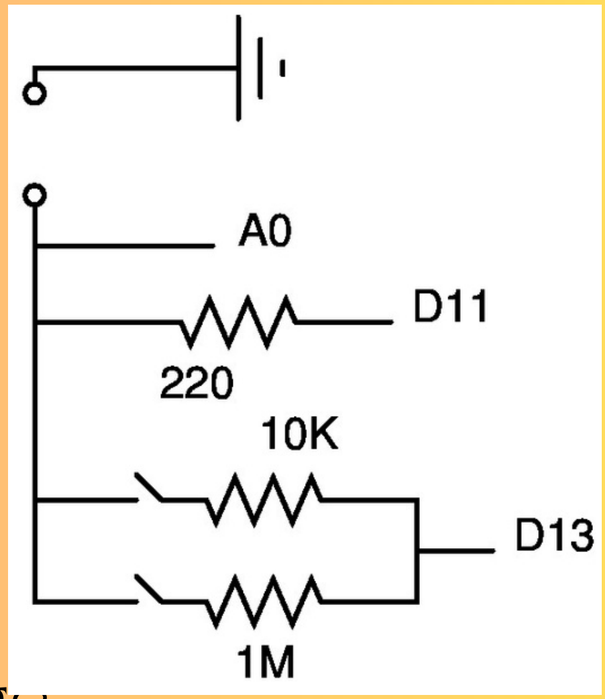
- Use a voltage divider circuit with two resistors (R1 and R2) in series to measure resistance
- The output voltage at the junction of the two resistors is proportional to the ratio of the second resistor (R2) to the sum of the two resistors (R1 + R2).
- Measure the voltage on the ADC1 pin using the analogRead() function.
- Calculate the resistance of the second resistor using the voltage divider formula [R2/(R1+R2)]Vin.
- Repeat the steps for each of the three different scales by switching between the D6, D7, and D8 pins.



Connecting i2C OLED (128x32)



3.Measuring Capacitance:



1. We charge the capacitor using Arduino pin 13 with known resistor in series.

2. When the capacitor charges we will start a time counter in micro seconds and we will measure the voltage through the Arduino pin A0

3. Use readADC_SingleEnded() function to read voltage values and process the data as needed.

4. We will now stop the time counter Voltage measured by Arduino A0 reads 63.2% of the Input Voltage

5. As we are using Arduino ADC for Measuring Voltage we will map all the voltage ranges to Arduino ADC 1024 bits.

6. The Arduino ADC has 10 bits so 0 volts would be 0 and 5 volts would be 1024. So 63.2% of 1024 is 648

7. So we will write a loop in the code in such away that timer stops after A0 value reaches 648.

8. But here the problem is Range for the capacitance, if capcitance is much smaller timer could not be accurate and results will be wrong.

9. So we will define ranges for mF to uF range we use 10K resistor and for nF to pF Range we use 1M Resistor

5. Now we got time constant value i.e 63.2% of Inout Voltage from the Formula C = R/T we calculate the Capacitance (Time constant = RC)

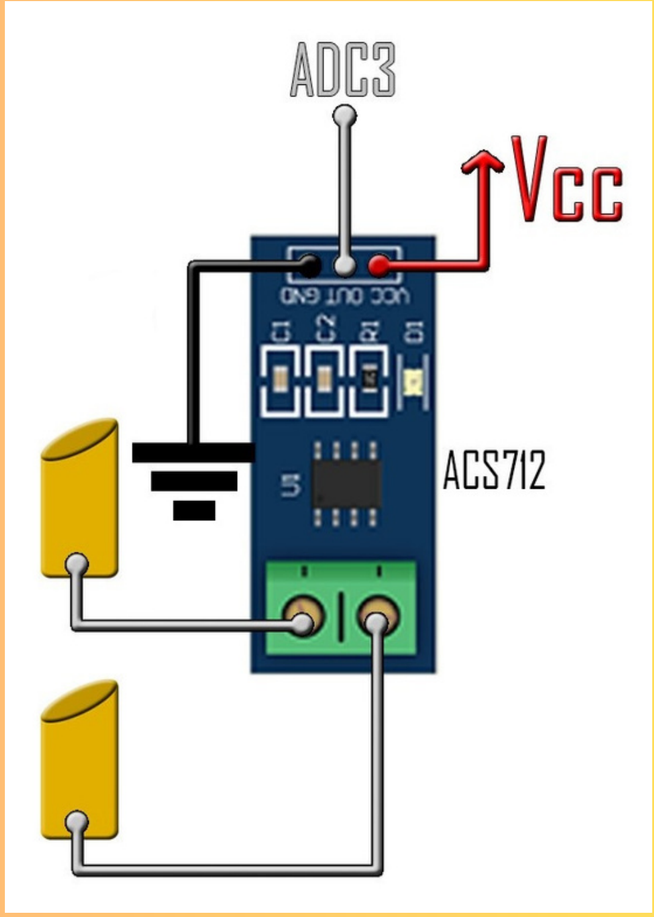
3.Measuring Current

1. Connect the sensor to Arduino's VCC, GND, and OUT to ADC3 of ADS115

2. Use readADC_SingleEnded() function to read voltage value of ADC3 and process the data as needed.

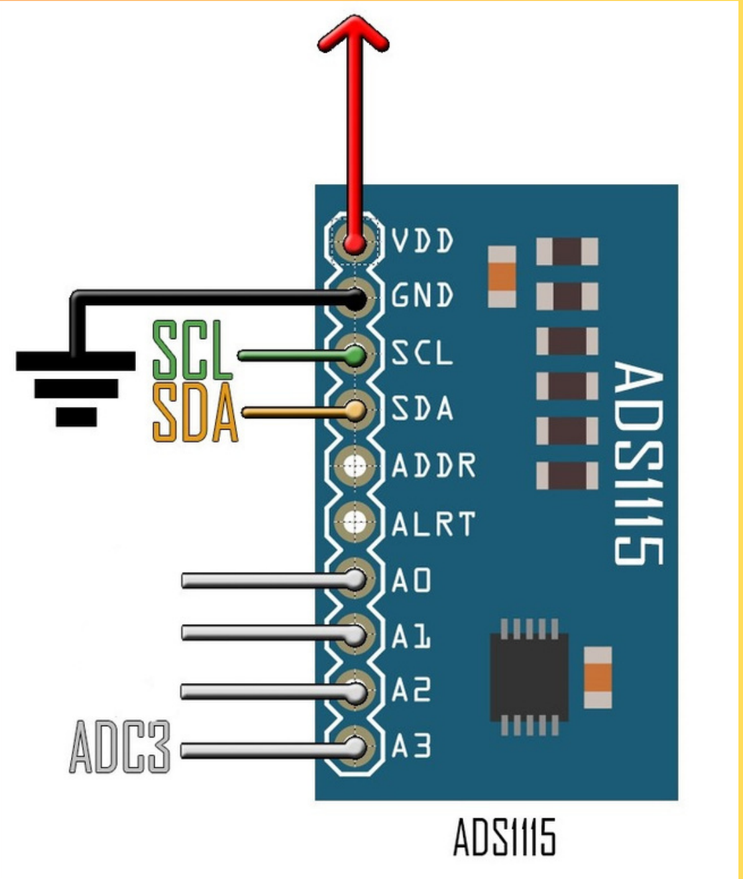
3. Repeat the measurement steps in a loop or desired interval for continuous voltage monitoring.

NOTE ;We used ADS115 Sensor of 5A range

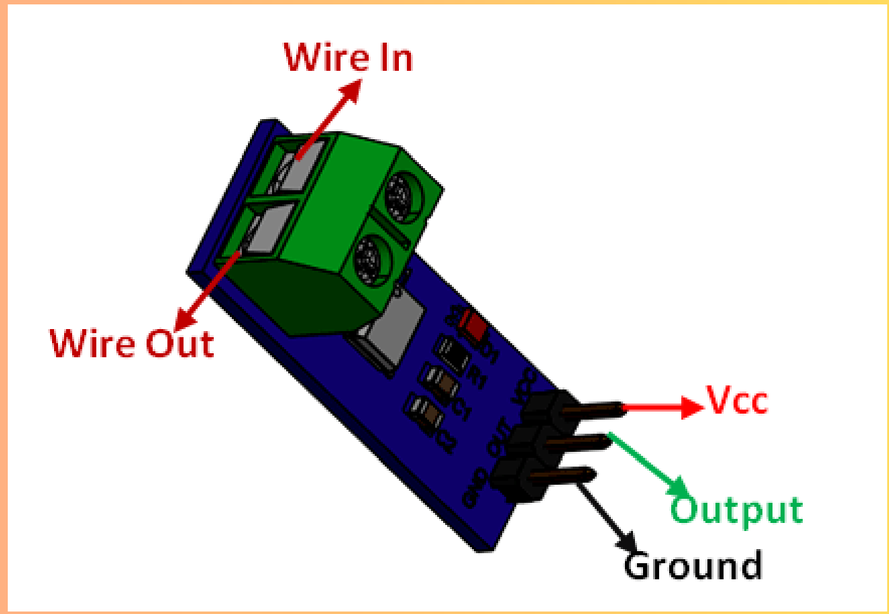


Arduino Code

```
void setup(void)
{
 Serial.begin(9600);
ads.begin();
}
void loop(void)
{
int16_t adc0;
adc0 = ads.readADC_SingleEnded(3);
float SensorVoltage = (adc3 * 0.1875)/1000;
float I =    (SensorVoltage-2.5)/Current_sensor_Resolution;
Serial.println(I);
delay(1000);
}
```



$$V_{out} = 2{,}5 + 185I_{in}$$



- V = Vcc/2 + 185I, represents the output voltage (V) of the ACS712 sensor.
- Where Vcc is the supply voltage, I is the current being measured, and 185 is the sensitivity or sensitivity factor of the ACS712 sensor in millivolts per ampere (mV/A).
- Overall, the ACS712 sensor provides a linear relationship between the current being measured and the output voltage, making it suitable for precise current measurements

SCL and SDA Pins

- A4 and A5 are used as the default SCL and SDA pins in Arduino boards that are based on the ATmega328P microcontroller, such as Arduino Uno, Arduino Nano, and Arduino Pro Mini.
- These pins are labeled as "A4" and "A5" on the Arduino boards, but they can also be used as general-purpose digital pins, labeled as "SDA" and "SCL" respectively, in the Arduino IDE.
- The I2C library in the Arduino IDE provides functions to simplify the communication with I2C devices using these pins.

Connecting Blutooth Module Hc-05