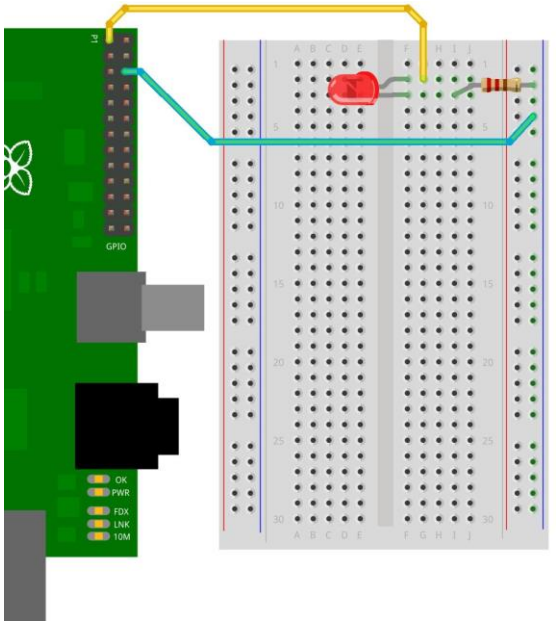
Interfacing sensors

Lab 5

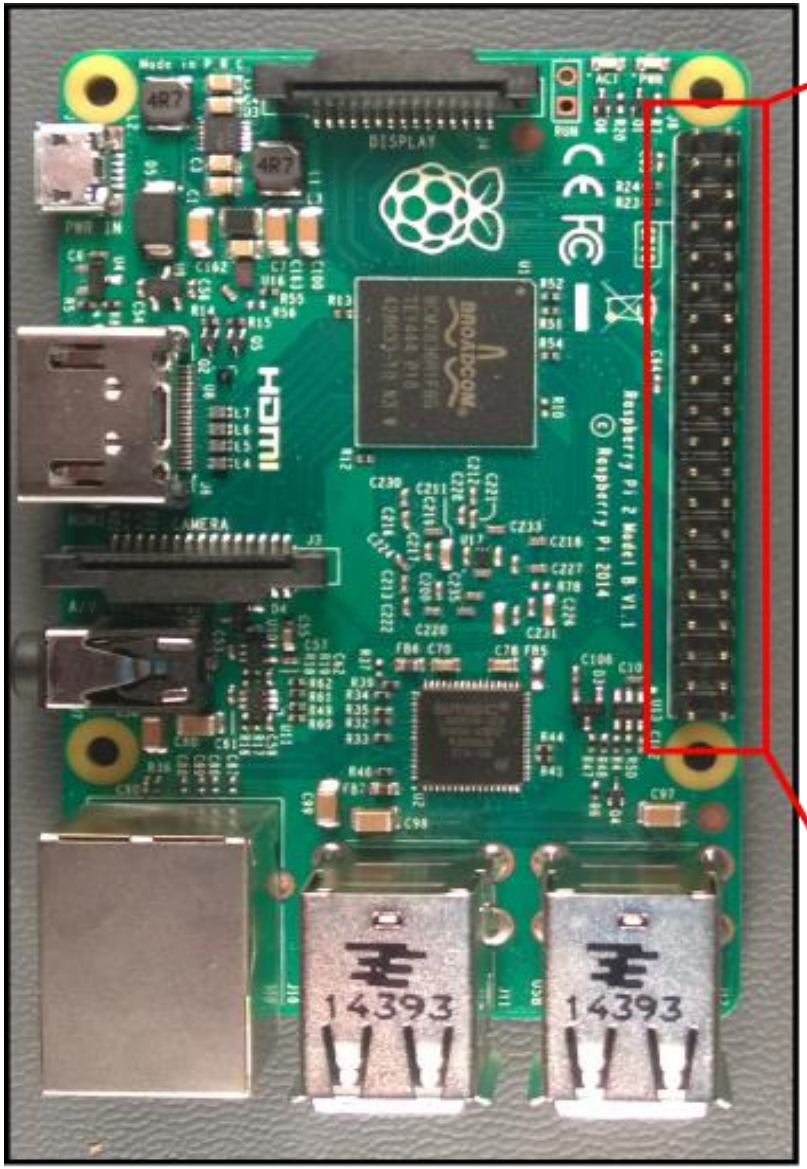


Learning outcomes:

At the end of this Lab 4, you should be able to:

- patch up I/O peripherals
- complete exercises
- patch up ADC IC and sensors
- complete exercises

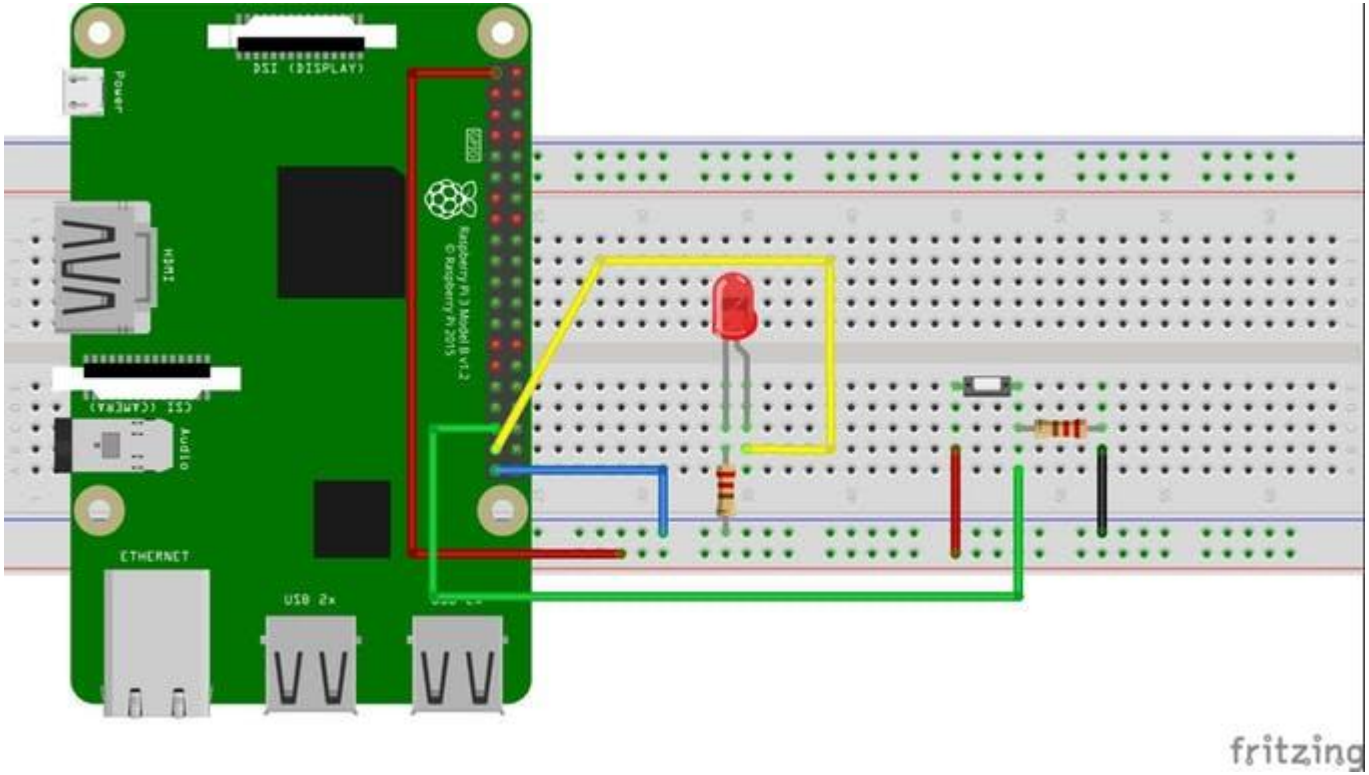
RPi GPIO pin diagram (40 pin)



3.3V PWR	1		2	5V PWR
I2C1 SDA	3		4	5V PWR
I2C1 SCL	5		6	GND
GPIO 4	7		8	Reserved
GND	9		10	Reserved
SPI1 CS0	11		12	GPIO 18
GPIO 27	13		14	GND
GPIO 22	15		16	GPIO 23
3.3V PWR	17		18	GPIO 24
SPI0 MOSI	19		20	GND
SPI0 MISO	21		22	GPIO 25
SPI0 SCLK	23		24	SPI0 CS0
GND	25		26	SPI0 CS1
Reserved	27		28	Reserved
GPIO 5	29		30	GND
GPIO 6	31		32	GPIO 12
GPIO 13	33		34	GND
SPI1 MISO	35		36	GPIO 16
GPIO 26	37		38	SPI1 MOSI
GND	39		40	SPI1 SCLK

Ex 1:LEDs:

- Connect Red and Green LEDs to pin **GPIO 22 and GPIO 27**. Blink both the LEDs continuously.

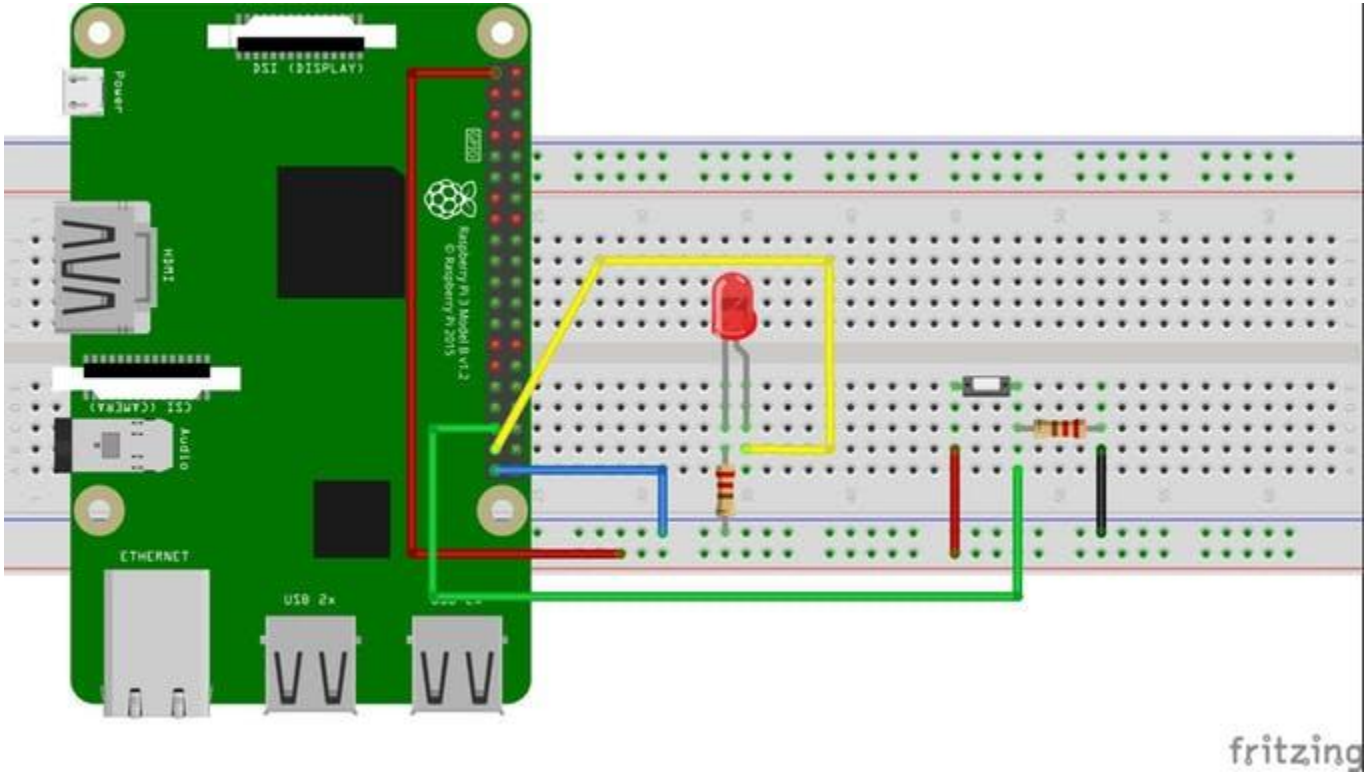


Ex 1:LEDs

```
import RPi.GPIO as G
G.setmode(G.BCM)  # set board mode to Broadcom
```

Ex 2:Button

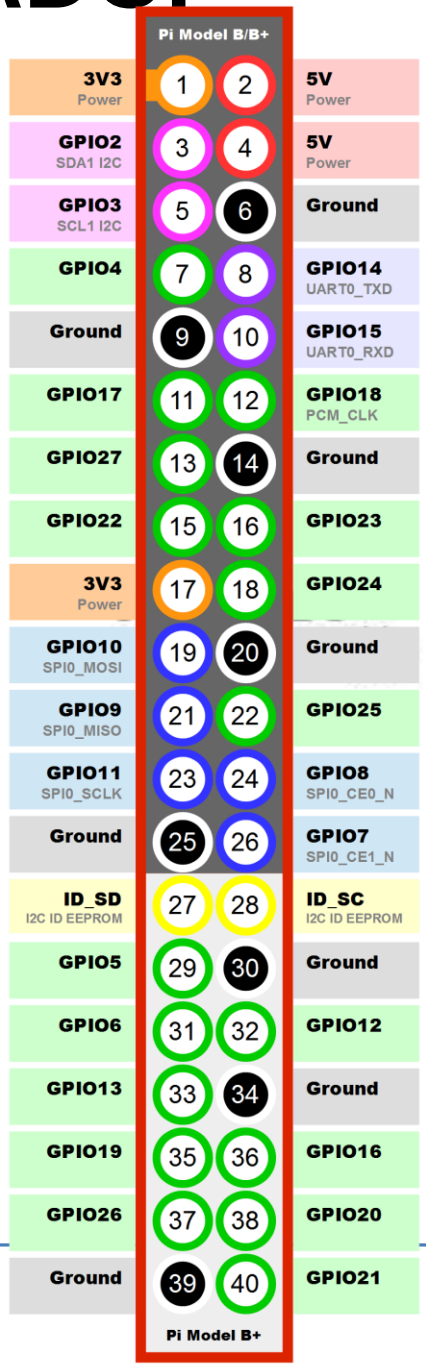
- Now connect button to **GPIO 4**. Turn ON both the LEDs whenever button is pressed. Otherwise, LEDs will be OFF.



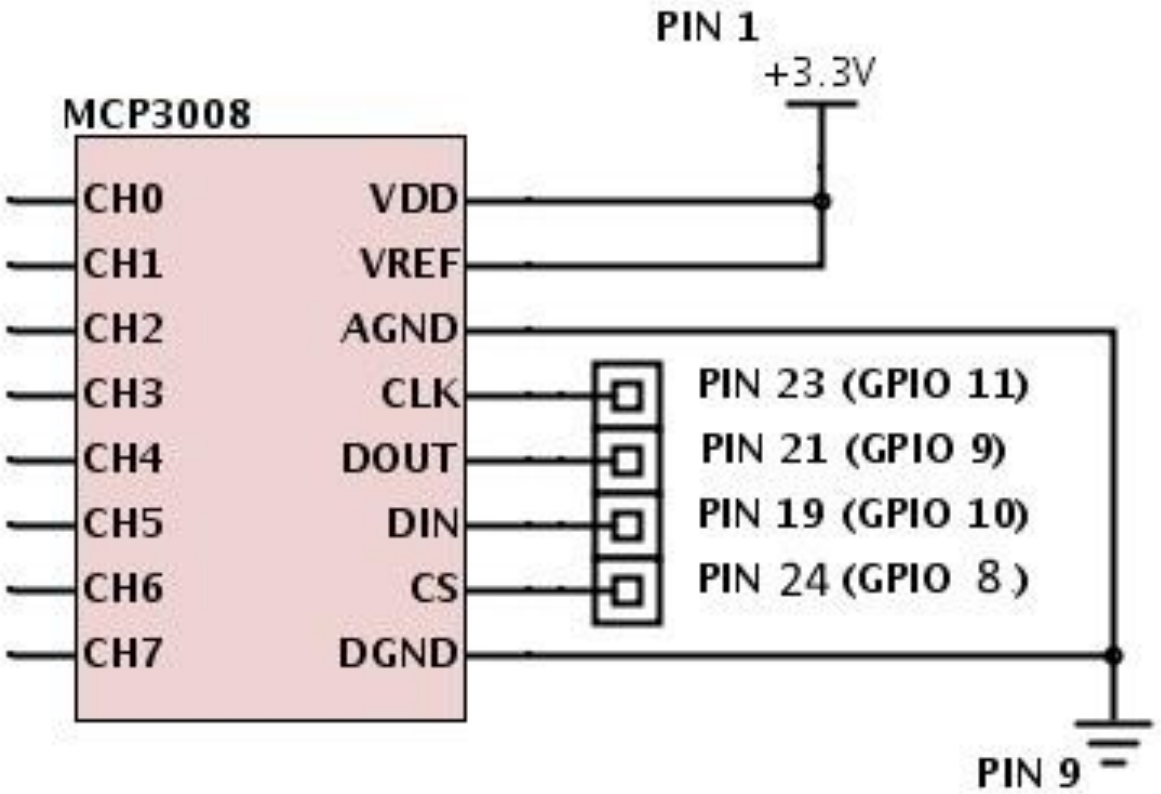
Ex 2:Button

```
import RPi.GPIO as G
import time
```

ADC:

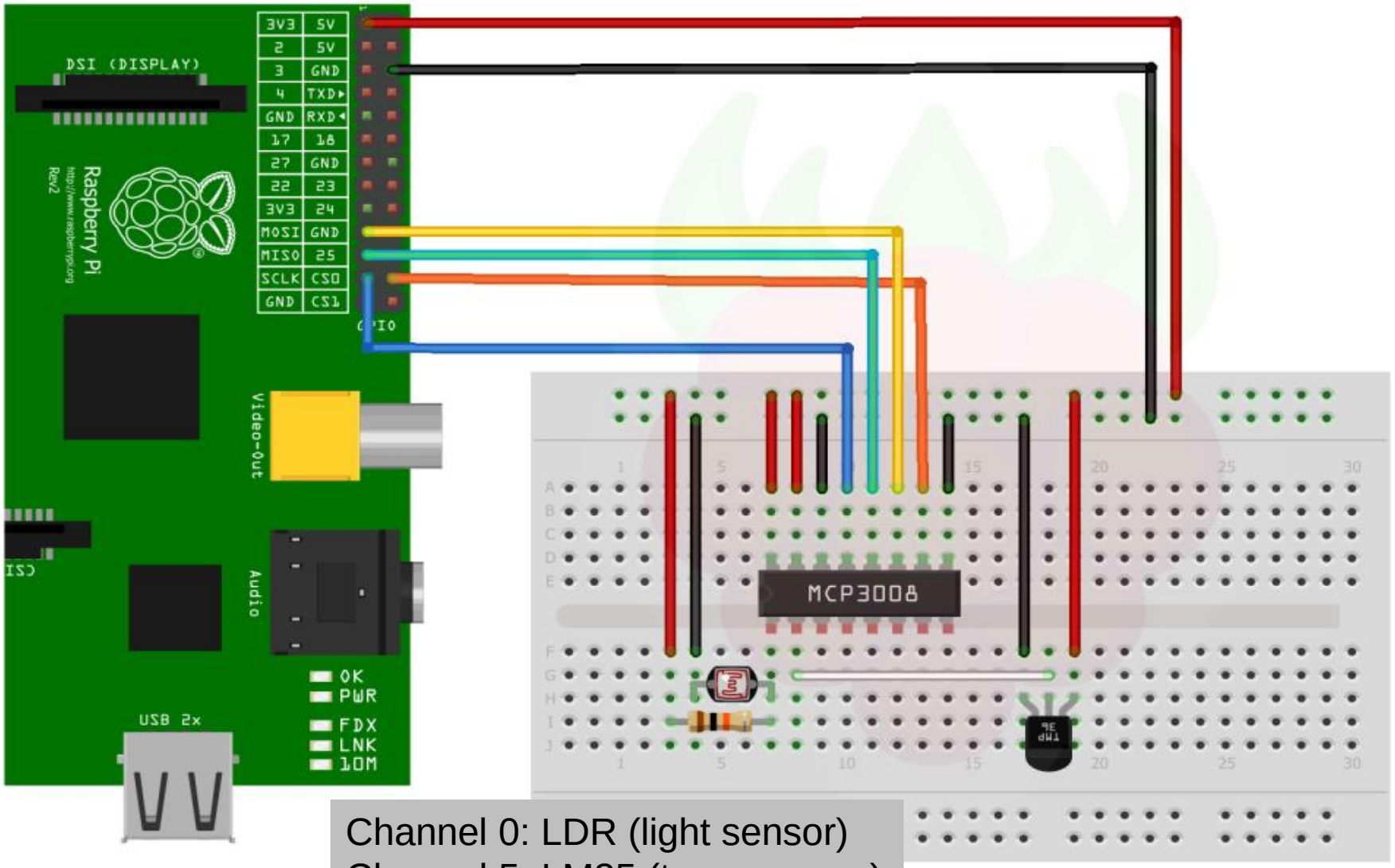


RPi is not capable of reading analog inputs, it requires an ADC.



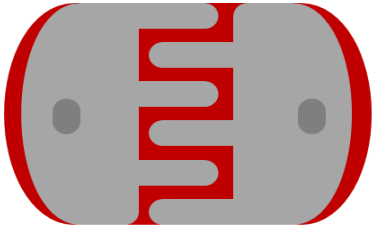
Channel 0: LDR: LDR (light)
Channel 5: LM35 (temp)

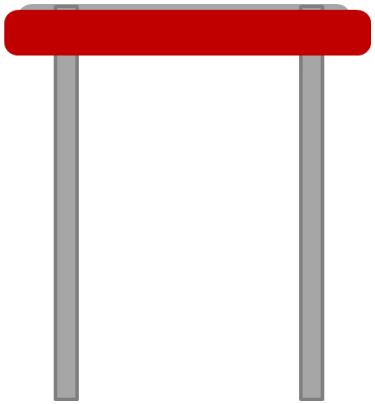
ADC MCP3008 Connection



Channel 0: LDR (light sensor)
Channel 5: LM35 (temp sensor)

Photoresistor (LDR)

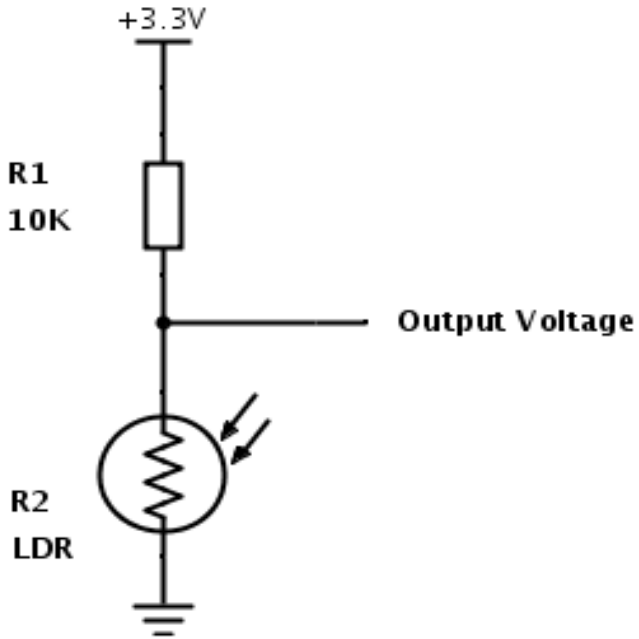




A photoresistor or **LDR** (**l**ight **d**ependent **r**esistor) is a resistor whose resistance depends on light intensity

An LDR can be used as a simple, **analog sensor.**

The orientation of an LDR does not matter



Ex 3:LDR

```
#!/usr/bin/python

import spidev
import time
import os

# Open SPI bus
spi = spidev.SpiDev()
spi.open(0,0)

# Function to read SPI data from MCP3008 chip
# Channel must be an integer 0-7
def ReadChannel(channel):
  adc = spi.xfer2([1,(8+channel)<<4,0])
  data = ((adc[1]&3) << 8) + adc[2]
  return data
```

Ex 3: ….continued

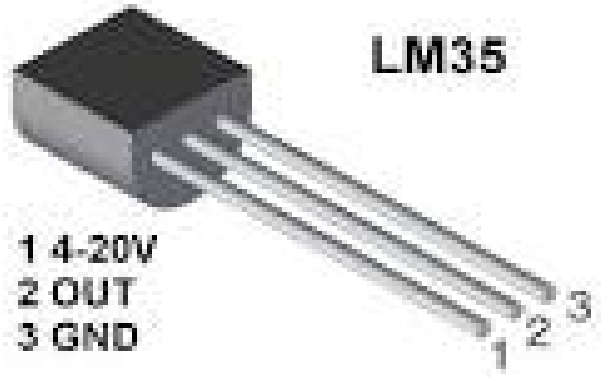
```
# Function to convert data to voltage level,
# rounded to specified number of decimal places.
def ConvertVolts(data,places):
  volts = (data * 3.3) / float(1023)
  volts = round(volts,places)
  return volts

# Define sensor channels
light_channel = 0

# Define delay between readings
delay = 5

while True:

  # Read the light sensor data
  light_level = ReadChannel(light_channel)
  light_volts = ConvertVolts(light_level,2)
```

Ex 3: ….continued

```
# Print out results
  print ("----------------------------------------“)
  print("Light: {} ({}V)".format(light_level,light_volts))

  # Wait before repeating loop
  time.sleep(delay)
```

Ex 4: LM35(Temp)

```
#!/usr/bin/python

import spidev
import time
import os

# Open SPI bus
spi = spidev.SpiDev()
spi.open(0,0)

# Function to read SPI data from MCP3008 chip
# Channel must be an integer 0-7
def ReadChannel(channel):
  adc = spi.xfer2([1,(8+channel)<<4,0])
  data = ((adc[1]&3) << 8) + adc[2]
  return data
```



Ex 4: ….continued

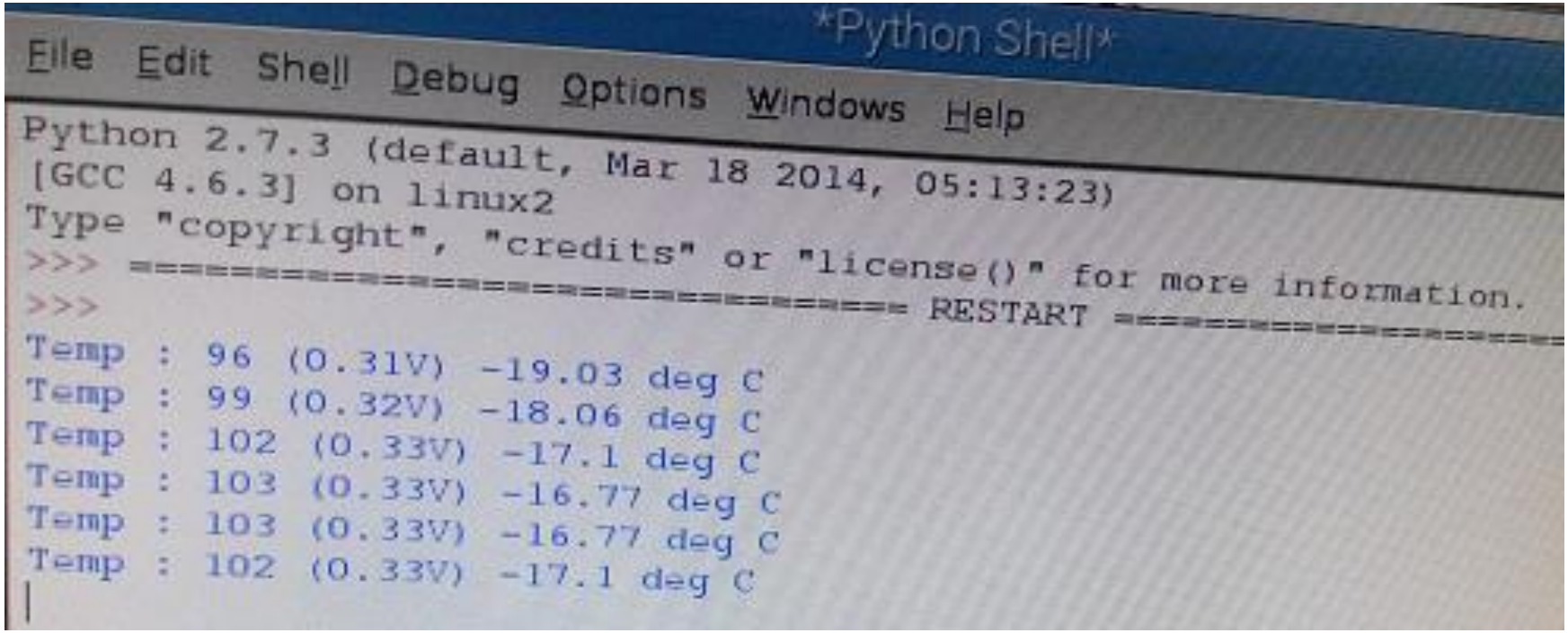
```
# Function to convert data to voltage level,
# rounded to specified number of decimal places.
def ConvertVolts(data,places):
  volts = (data * 3.3) / float(1023)
  volts = round(volts,places)
  return volts

# Function to calculate temperature from, LM35 data, rounded to specified
# number of decimal places.
def ConvertTemp(data,places):
   # ADC Value
  # (approx)  Temp  Volts
  #    0      -50    0.00
  #   78      -25    0.25
  #  155        0    0.50
  #  233       25    0.75
  #  310       50    1.00
```

Ex 4:….continued

```
  temp = ((data * 3.3)/float(1023)) + 25
   temp = round(temp,places)
   return temp

temp_channel  = 5

delay = 5

while True:
   # Read the temperature sensor data
   temp_level = ReadChannel(temp_channel)
   temp_volts = ConvertVolts(temp_level,2)
   temp       = ConvertTemp(temp_level,2)

   #Print out results
    print("Temp : {} ({}V) {} deg C".format(temp_level,temp_volts,temp))

   time.sleep(delay)
```

Ex 5: Combine LDR & LM35

- Write a program to read and display light and temp values. The expected output is shown below.



Ex 5: Combine LDR & LM35

Ex 6: Writing data into a text file

```
#Print out results
 print("Light: {} ({}V)".format(light_level,light_volts))
 print("Temp : {} ({}V) {} deg C".format(temp_level,temp_volts,temp))

# Add the following lines (highlighted) to write the data into a text file
called Output.txt

text_file= open("Output.txt", "a+")
text_file.write("Light: {} ({}V)".format(light_level,light_volts))
text_file.write("Temp : {} ({}V) {} deg
        C".format(temp_level,temp_volts,temp))
text_file.close()


#wait before repeating loop
time.sleep (delay)
```