# IoT Lab 2 Report

## Team name: The Gradient Ascenders

Kanika Narang
MT2016069

Meghna Srivastava
MT2016082

Shivani Naik
MT2016088

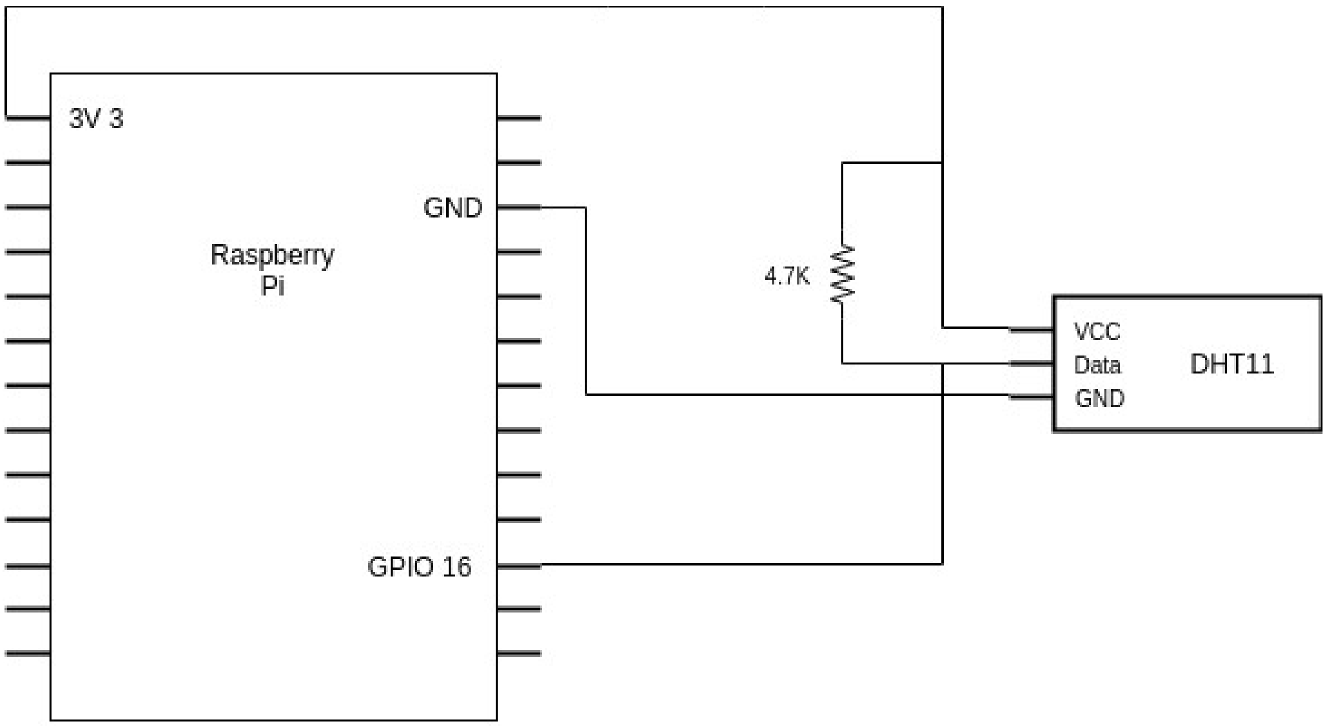
**Problem Statement:**

1. Study the specifications of a temperature/humidity or light sensor available in lab. Create a pin diagram to show the sensor interfacing to Raspberry Pi.
2. Connect the sensor to Pi. Set up the system to read the sensor data from terminal or using Python.
3. Print this sensor data on screen.
4. Write a program to log/ store sensor data.
5. Check the improvement in data accuracy by adding second temperature/humidity or light sensor (if available or share with other project group) to the Pi-based setup.

## DHT11 Sensor Specifications:

- Low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50°C temperature readings ±2°C accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm
- 3 pins ( Data, VCC, GND)

## Pin Diagram:



**Raspberyy Pi to DHT11 Circuit Diagram**

**Code:**

```
#Problem: Reading temperature and humidity sensor data and storing it in log file
import Adafruit_DHT as dht
F = open("log1.txt","w")
i = 0
#Take 10 readings
while i<10:
    #Reading humidity and temperature from DHT11 sensor connected to
    #GPIO 16 pin of Raspberry Pi
    humidity, temperature = dht.read_retry(dht.DHT11, 16)
    #Write data to the log file
    F.write('Temp={0:0.1f}*  Humidity={1:0.1f}%\n'.format(temperature, humidity))
    i += 1
F.close()
```

- Adafruit provides a library to interface with the DHT11 sensor and retrieve its humidity and temperature readings.

- We created a file log1.txt in write mode to store the readings taken from the sensor. We have taken 10 readings using the read_retry() function provided by Adafruit. We have to provide the sensor type (DHT11) and the GPIO pin as the parameters to this function.

- Then we have printed these readings to the log file with the required precision of 1 digit after the decimal point.

- We have taken the readings using 2 sensors as mentioned in the problem statement. Log1.txt stores output of sensor 1 and log2.txt stores output of sensor 2.

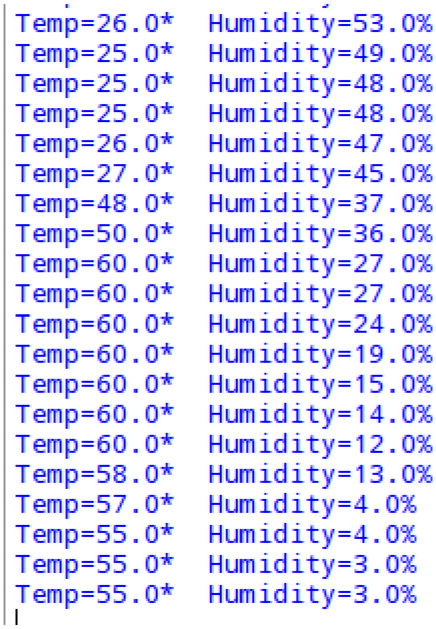
**OUTPUT:**

| Log 1 | Log 2 |
|---|---|
| Temp=23.0*  Humidity=33.0% | Temp=25.0*  Humidity=39.0% |
| Temp=23.0*  Humidity=33.0% | Temp=24.0*  Humidity=43.0% |
| Temp=23.0*  Humidity=33.0% | Temp=24.0*  Humidity=42.0% |
| Temp=23.0*  Humidity=33.0% | Temp=24.0*  Humidity=42.0% |
| Temp=23.0*  Humidity=33.0% | Temp=24.0*  Humidity=42.0% |
| Temp=23.0*  Humidity=32.0% | Temp=24.0*  Humidity=42.0% |
| Temp=23.0*  Humidity=33.0% | Temp=24.0*  Humidity=42.0% |
| Temp=23.0*  Humidity=33.0% | Temp=24.0*  Humidity=42.0% |
| Temp=23.0*  Humidity=32.0% | Temp=24.0*  Humidity=41.0% |
| Temp=23.0*  Humidity=33.0% | Temp=24.0*  Humidity=43.0% |

**Observations:**

- As we can observe, the temperature and humidity readings in log 1 and log 2 are almost same. The sensors are quite precise as they are giving same value for same input over 10 readings.

- We performed another experiment by using a blow dryer to increase the temperature and decrease the humidity of the surroundings. Following were the readings for this. As we can see, the temperature has raised to 60 degrees and humidity has reduced to around 3%.



```
Temp=26.0*  Humidity=53.0%
Temp=25.0*  Humidity=49.0%
Temp=25.0*  Humidity=48.0%
Temp=25.0*  Humidity=48.0%
Temp=26.0*  Humidity=47.0%
Temp=27.0*  Humidity=45.0%
Temp=48.0*  Humidity=37.0%
Temp=50.0*  Humidity=36.0%
Temp=60.0*  Humidity=27.0%
Temp=60.0*  Humidity=27.0%
Temp=60.0*  Humidity=24.0%
Temp=60.0*  Humidity=19.0%
Temp=60.0*  Humidity=15.0%
Temp=60.0*  Humidity=14.0%
Temp=60.0*  Humidity=12.0%
Temp=58.0*  Humidity=13.0%
Temp=57.0*  Humidity=4.0%
Temp=55.0*  Humidity=4.0%
Temp=55.0*  Humidity=3.0%
Temp=55.0*  Humidity=3.0%
```