

Practical no: 4

Title: Understanding the connectivity of Raspberry-Pi /Beagle board circuit with temperature sensor. Write an application to read the environment temperature. If temperature crosses a threshold value, the application indicated user using LEDSs

Name: Aditi Dinesh Mulay

Class: T.E. Computer

Subject: ES&IOT

Div: A

Roll no: 02

PRN No. 71918146B

ES10T

Practical 4

Aditi Dinesh Mulay
T.E. Comp Div: A
Roll no: 02

Aim

- 1) To understand the concept of Temperature-Humidity Sensor.
- 2) To interface Temperature-Humidity sensor with Raspberry Pi model.
- 3) To program the Raspberry Pi model to measure the real time Temperature & Humidity of the Environment.

Software : Raspbian OS (IDE)

Hardware Modules: 1) Raspberry Pi Board Module
2) Temp-Humidity sensor (DHT11) module.
3) Monitor.



Theory

- 1) Physical quantities like Humidity, temperature, pressure etc. are monitored to get information about the environmental conditions.
- 2) Temperature is basically amount of heat present in environment. Humidity is the presence of water vapors in air. The temperature & amount of water vapor in air can affect human comfort as well as many manufacturing processes in industries. The presence of water vapour also influences various physical, chemical, biological processes.
- 3) In our module we are using "DHT11 Temperature & Humidity Sensor".
- 4) The features of this sensor are calibrated digital signal

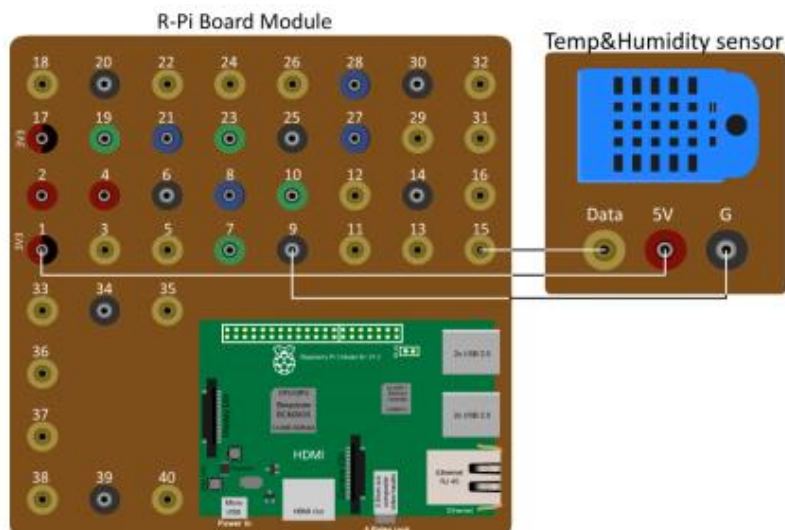
- output, and high reliability of semiconductor material changes as per humanity in environment changes.
- 5) This sensors also includes NTC temp measurement component which detects the change in temp.
 - 6) DHT11 basically provides two output from single data pin semiconductor material.

Steps for assembling circuit:

- 1) Connect the VCC pin of Temperature & Humidity sensor (DHT11) to VCC pin of Raspberry Pi module.
- 2) Connect the DATA pin of Temperature & Humidity sensor (DHT11) to GPIO pin 15 of Raspberry Pi module.
- 3) Connect the GND pin of Temperature & Humidity Sensor (DHT11) to GND pin of Raspberry Pi module.

Procedure to install DHT11 library:

- 1) Open LxTerminal of Raspberry Pi & enter the following commands.
- 2) `sudo apt-get update`
- 3) `sudo apt-get install git-core`
- 4) `cd ~`
- 5) `git clone https://github.com/szazo/DHT11-Python.git`
- 6) `cd DHT11-Python`
- 7) `sudo python setup.py install`



Procedure.

- 1> Write the program as per the algorithm given below.
- 2> Save program in the library folder.
- 3> Run code using Run module.

Algorithm.

- 1> Import GPIO, time and dht11 libraries
- 2> Set all warnings as False.
- 3> Set mode i.e. GPIO.BOARD
- 4> Read data using GPIO pin number 7
- 5> Write 'while loop' for displaying Temperature & Humidity values continuously
- 6> First read the GPIO pin and store the data in dhtValue Variable.
- 7> Print the Humidity value.
- 8> Give delay of 1 second.

Observations.

- 1> Observe the output on python shell as per program.
- 2> Observe the ON & OFF status of buzzer.

Conclusion

Thus, we have studied the concept of Temperature and Humidity sensor with Raspberry Pi model.

```
import lcddriver
import time
import Adafruit_DHT

display = lcddriver.lcd()

try:
    print("Press CTRL + C for stop this script!")

    def long_string(display, text = '', num_line = 1, num_cols = 16):
        if(len(text) > num_cols):
            display.lcd_display_string(text[:num_cols],num_line)
            time.sleep(1)
            for i in range(len(text) - num_cols + 1):
                text_to_print = text[i:i+num_cols]
                display.lcd_display_string(text_to_print,num_line)
                time.sleep(0.5)
            time.sleep(1)
        else:
            display.lcd_display_string(text,num_line)

    long_string(display, "DHT LCD R Pi!",1)
    time.sleep(1)

    long_string(display, "Rohit World ",2)
    time.sleep(1)
    display.lcd_clear()

    while True:
        humidity, temperature = Adafruit_DHT.read_retry(11,4)
        if(temperature != None and humidity != None):
            display.lcd_clear()
            display.lcd_display_string('Temp:{0:0.1f} C
'.format(temperature),1)
            display.lcd_display_string('Humidity:{0:0.1f} %
'.format(humidity),2)
            time.sleep(1)

except KeyboardInterrupt:
    print("Cleaning up!")
    display.lcd_clear()
    display.lcd_display_string ('AsknCapture',1)
    display.lcd_display_string('8237877250',2)
    time.sleep(10)
    display.lcd_clear()
```