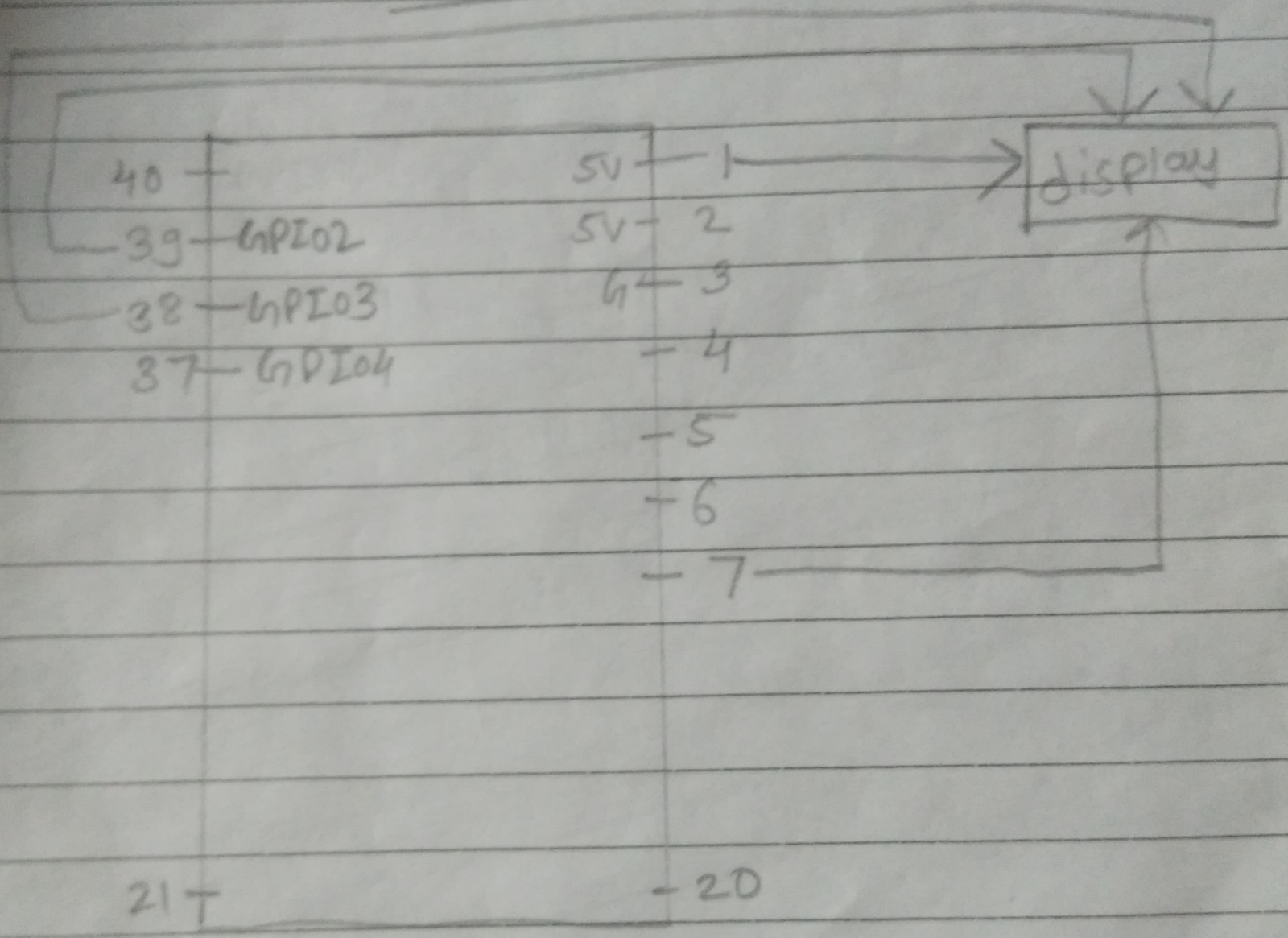
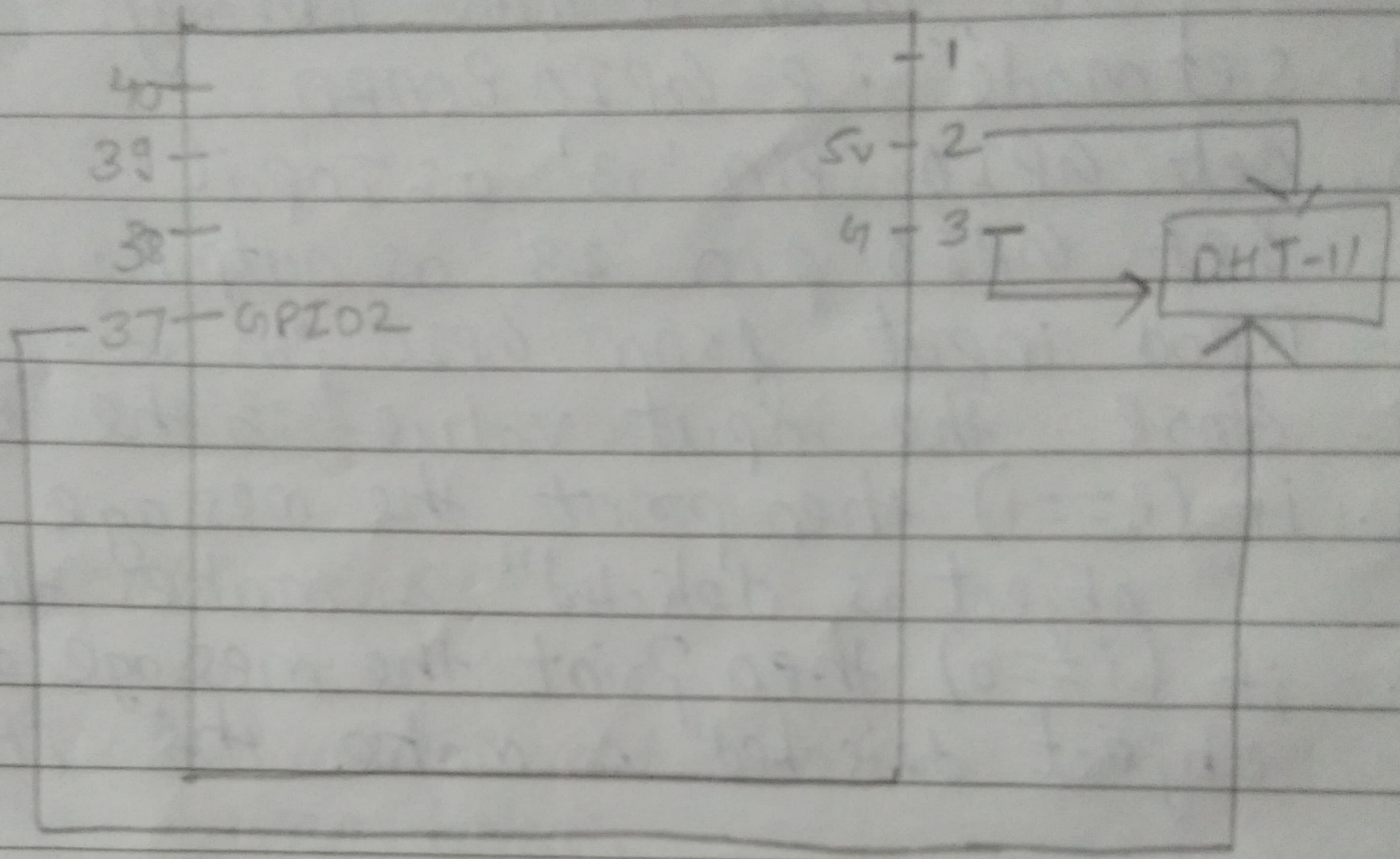
Temp sensor (DHT-11)



GPIO - Display



RHT-11

Experiment No: 4

Aim: understanding the connectivity of Raspberry-Pi / Beagle board circuit with IR sensor. Write an application to detect obstacle & notify user using LEDs.

Theory:

Infrared sensor works by emiting infrared signal / radiation & receiving of the signal when the signal bounces back from any obstacle.

[tx-animate animation="fadeIn" duration="5" delay="4" inline="no"].

[/tx-animate]

IR sensor fig 1

IR Sensor Fig.1

Components : IR sensor.

1. Emitter : This component continuously emits the infrared signal.

2. Receiver : It waits for the signal which is bounced back by obstacle.

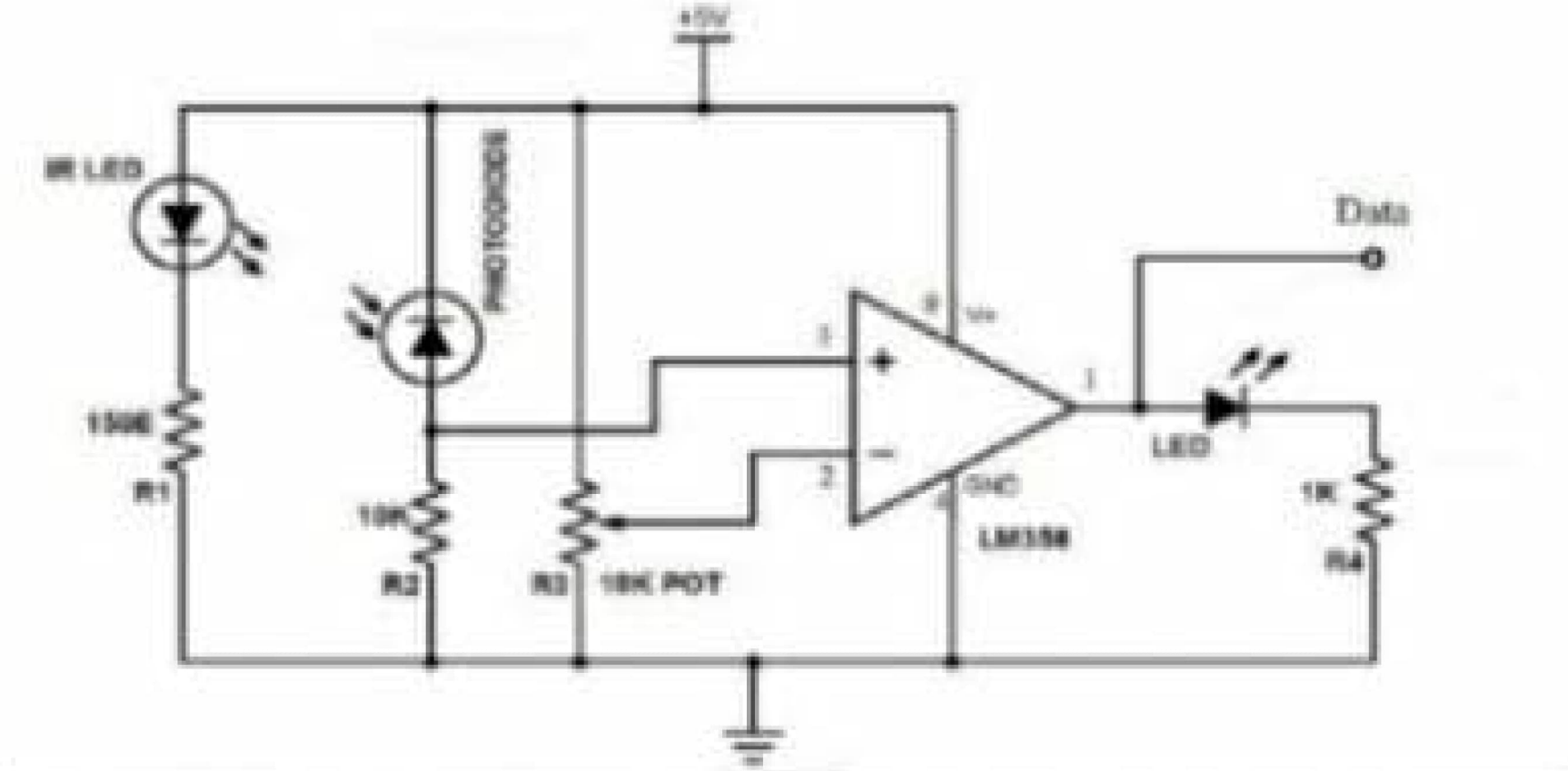
3. Indicator : on board LED to signal if obstacle is deducted by the sensor.

4. o/p : Could be used as Input for further processing of the signal.

5. Ground : Ground/Negative point of the Circuit.

6. Voltage : Input 3.3V.

circuit diagram of IR sensor.



Circuit diagram of IR Sensor Fig.2

objective:

1. Raspberry Pi 3.
2. IR (Infrared) Sensor.
3. 1 LED.
4. 1 Resistor (330 Ω).
5. Few jumper cables.
6. 1 Breadboard.

Circuit: To detect obstacles.

We will be creating a circuit which will turn on the LED when an obstacle is detected. And as soon as the obstacle is removed from the way the LED will turn off. In order to achieve that follow steps.

Part 1: Connecting IR sensor.

IR sensor has 3 pins, viz VCC, GND & OUT. We will use GPIO 17 for receiving input from the sensor.

1. Connect GPIO 17 from Raspberry Pi to Breadboard (5a).
2. Connect OUT pin of the sensor with the Breadboard (5c).
3. Connect GND with negative line on left side of the breadboard.
4. Connect GND of the IR sensor to breadboard(10c)
5. Connect GND from step 3 to breadboard (10a).
6. Connect VCC of the IR sensor to bread (15c)
7. Connect 3v3 to positive line on left side of the breadboard.
8. Connect 3v3 to the breadboard (15a).

Now, the circuit is complete & sensor will detect the obstacle. It can be tested by putting anything in front of the IR sensor.

Part 2: Connecting LED

1. Connect GPIO 4 from the board to the breadboard (20 c).
2. Connect positive point of the LED to the breadboard (20 c).
3. Connect negative point of the LED to the breadboard (22 c).
4. Use resistor (220 Ω) to connect negative to the negative point of the LED. (22 a).

Now, we are ready to send signal based on the i/p received from IR sensor to turn on/off the LED.

Part 3: Code to connect IR Sensor I/P with LED status.

```
from GPIO zero import LED
from Signal import Pause
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
LED_PIN = 27
IR_PIN = 17

indicator = LED(LED_PIN)
GPIO.setup(IR_PIN, GPIO.IN)

count = 1
while true:
    get_Something = GPIO.input(IR_PIN)
    if get_Something:
        indicator.on()
        Print("{:733 Got Something"
        .format(count))
    else
```

```
        indicator.off()
        Print("{:>3} Nothing detected".
            format(count))
        count+=1
        time.sleep(0.2)
```

Part 4: Executing the code:

1) open terminal (on Pi itself).
2) Navigate to the directory where the above code is saved.
3) Type : Python3 ir-obstacle.py & press <enter>.

on terminal it will start printing the status based on the condition.

Conclusion:

Thus, we done connectivity of Raspberry-Pi/Beagle board circuit with IR sensor. write an application to detect obstacle & notify using LEDs.

Understanding the connectivity of Raspberry Pi board circuit with IR sensor.

Aim/objectives:

1. To understand the conlept of proximity sensor.
2. To interfale proximity sensor with Raspberry Pi model.
3. To Program the Raspberry Pi model to detect the nearest object using proximity sensor & give indication through led.

Software:

1. Raspbian OS (IDLE).

Hardware Modules:

1. Raspberry Pi board.
2. Proximity Sensor, LED, 330 ohm register.
3. Monitor.

Theory:

1. Proximity IR sensor is a small board containing an IR transmitter, photodiode, IR Receiver & some processing circuitry.
2. This is a discreate sensor that sensed when an object comes near to the sensor face.
3. It works by detecting reflected infrared light & it can glow onboard led when object is directly front of it.
4. by measuring the ammount of reflected infrared light & it can glow onboard led when object is directly front on it.
5. In proximity, it consist of two leds one is the transmitter & another is receiver.
6. The IR led transmits the infrared light signal which reaches till the object & deflects back.
7. The photo diode receives the deflected light.
8. This signal is then amplified & status of this signal is checked by the micro-controller.
9. Proximity sensor is more sensitive but it ~~decides~~ detects only object but cannot measure adistance value.
10. by using a potentiometer we can change sensitivity accordingly.

11. when this sensor detects the object, it gives output as a digital value i.e '1' & if not detected then the value is '0'

diag Proximity sensor.

Safety precautions:

1. first, make all the connections as per step given below.
2. power supply.

Interface diagram.

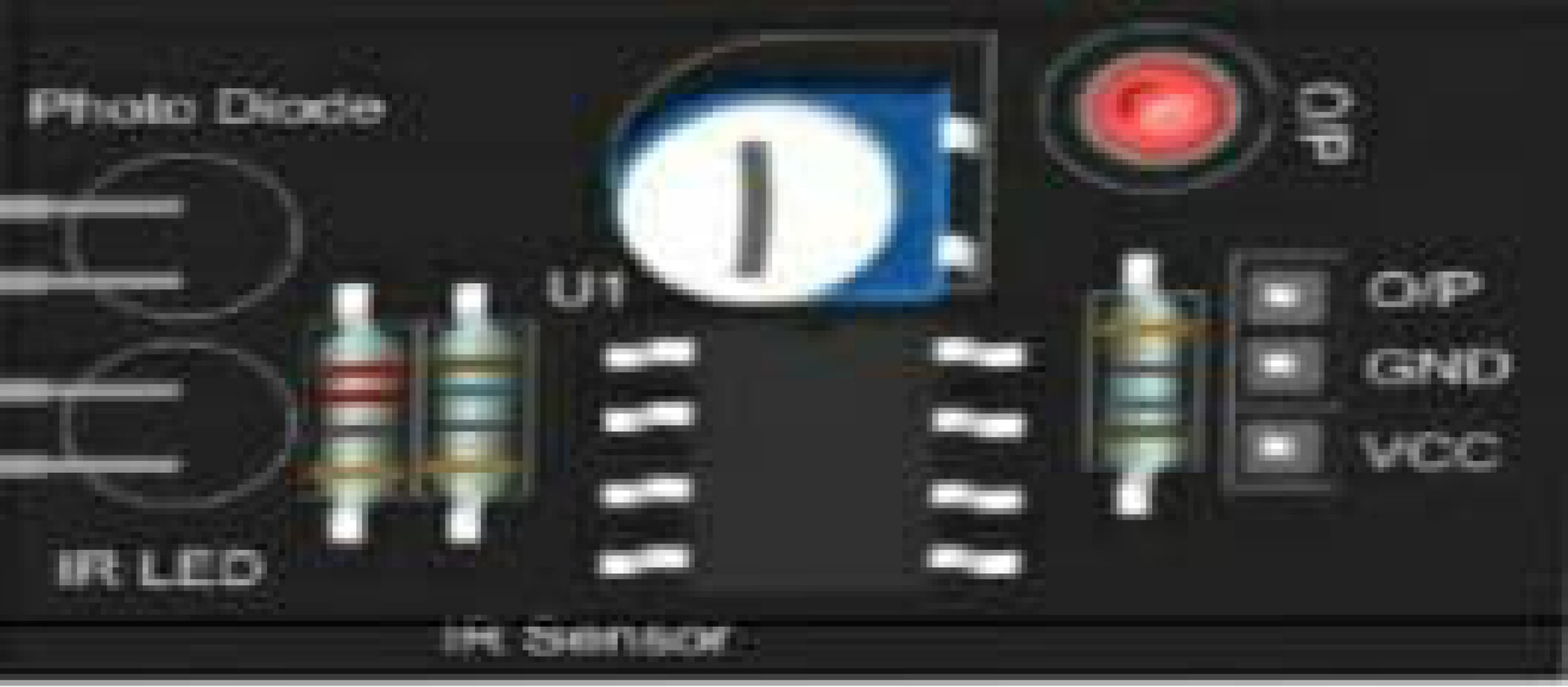
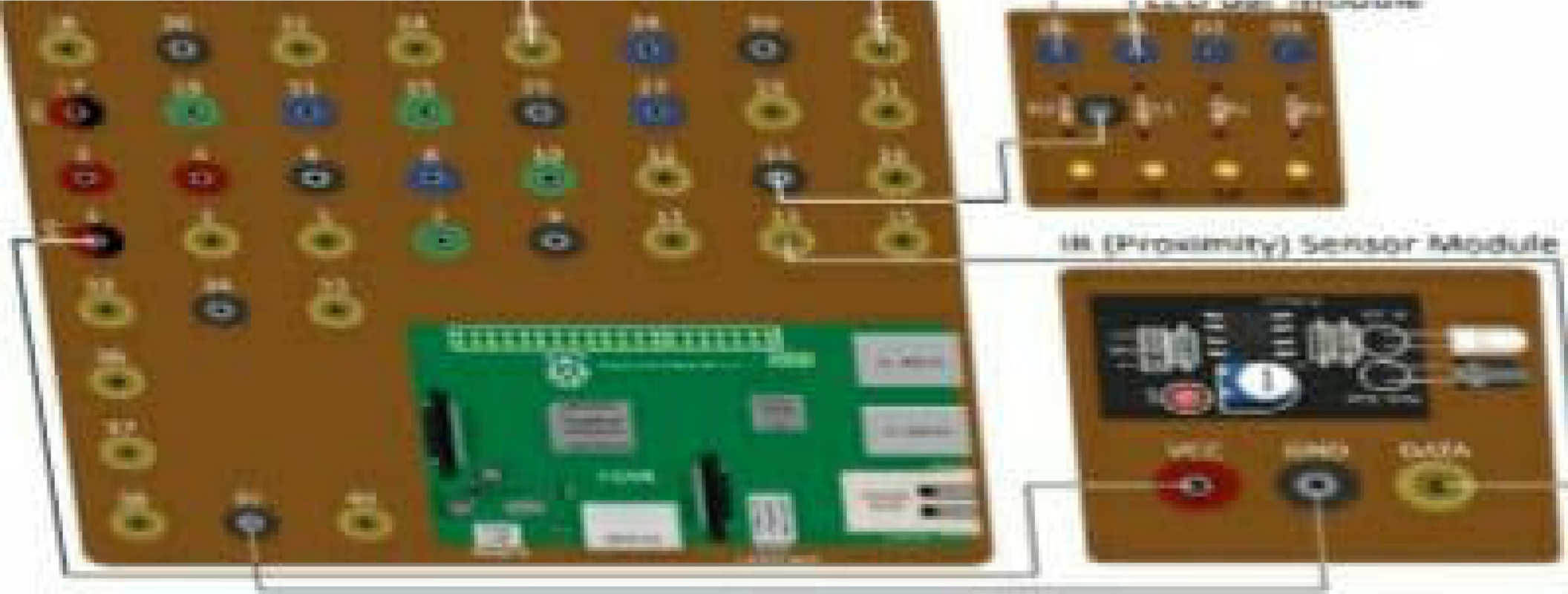


Photo Diode
U1
O/P
O/P
GND
VCC
IR LED
IR Sensor


IR (Proximity) Sensor Module
VCC
GND
DATA

Steps for assembling Circuit:

1. Connect the VCC pin of proximity sensor to 3.3v of Raspberry Pi Module.
2. Connect the GND pin of proximity sensor to GND pin of Raspberry Pi Module.
3. Connect the DATA pin of proximity sensor to pin '13' of Raspberry Pi module & connect the Do pin of LED bar to pin '28' of Raspberry Pi Module.
4. Connect the GND pin of LED bar to GND pin of Raspberry Pi Module.

Procedure:

5. Write the Program as the algorithm.
6. Save program.
7. Run Code using Run Module.

Algorithm:

8. import GPIO & time library.
9. setmode i.e GPIO BOARD.
10. set GPIO Pin '13' as Input.
11. Set GPIO pin '28' as output.
12. Read input from GPIO pin '13'.
13. Store the input value in the variable 'i'.
14. if (i==1) then print the message as "object is detected" & make the LED on
15. if (i==0) then print the message as "No object detected" & make the LED off.

Observation:

I. See output on command prompt or Python Shell & also check LED status.

```
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 RPi!", 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 & humidity !=
                None):
            display.lcd_clear()
```

```
        display.lcd_display_string('Temp:
                {0:d} If 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('Ask n
            Capture', 1)
    display.lcd_display_string('823--', 2)
    time.sleep(10)
    display.lcd_clear()
```