G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT (AIOT)
LABCOMPONENT

VII Semester : EEE						Scheme : 2020	
Course Code	Hours/Week			Credits	Maximum Marks		
SCEE04	L	T	P	C	Continuous Internal Assessment	End Exam	TOTAL
	0	0	4	2	40	60	100

End Exam Duration: 3 Hrs

Course Outcomes : At the end of the course students will be able to

CO1: Understand interfacing of sensors and motors with NodeMCU.

CO2: Understand interfacing of sensors and motors with ESP32.

CO3: Understand interfacing of sensors and motors with Raspberry PI.

List of Experiments

1. To interface Ultrasonic sensor with NodeMCU and display the sensor output.
2. To interface temperature and humidity sensor with NodeMCU and display the sensor output.
3. To interface PIR sensor with NodeMCU and display the sensor output.
4. To interface IR sensor with NodeMCU and display the sensor output.
5. To interface Array of LEDs with NodeMCU and
   a)turn on ALL LEDs once b)turn on LEDs one by one from left to right and then from right to left.
6. To interface an Array of LEDs with NodeMCU and turn on LEDs to display a hexadecimalnumber equivalent value from 00 to FF.
7. Interfacing DC motor with NodeMCU.
8. To interface Ultrasonic sensor with ESP32 and display the sensor output.
9. To interface temperature and humidity sensor to ESP32 and display the sensor output.
10. To interface PIR sensor with ESP32 and display the sensor output.
11. To interface IR sensor with ESP32 and display the sensor output.
12. To interface an Array of LEDs with ESP32 and
    a)turn on the ALL LEDs once b)turn on LEDs one by one from left to right and then from right to left.
13. To interface an Array of LEDs with ESP32 and turn on LEDs to display a hexadecimal numberequivalent value from 00 to FF.
14. To interface DC motor with ESP32.
15. To interface Ultrasonic sensor with Raspberry PI and display the sensor output.
16. To interface temperature and humidity sensor with Raspberry PI and display the sensor output.
17. To interface PIR sensor with Raspberry PI and display the sensor output.
18. To interface IR sensor with Raspberry PI and display the sensor output.
19. To interface an Array of LEDs with Raspberry PI and
    a)turn on the ALL LEDs once b)turn on LEDs one by one from left to right and then from right to left.
20. To interface an Array of LEDs with Raspberry PI and turn on the LEDs to display a hexadecimal number equivalent value.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT (AIOT)
LABCOMPONENT

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Introduction to NodeMCU ,ESP32 ,Rasberry Pi and Sensors

GPREC/DEEE/EXPT-AIOT (P)-INTRODUCTION **Date: 17/01/2023**

1.NODEMCU:

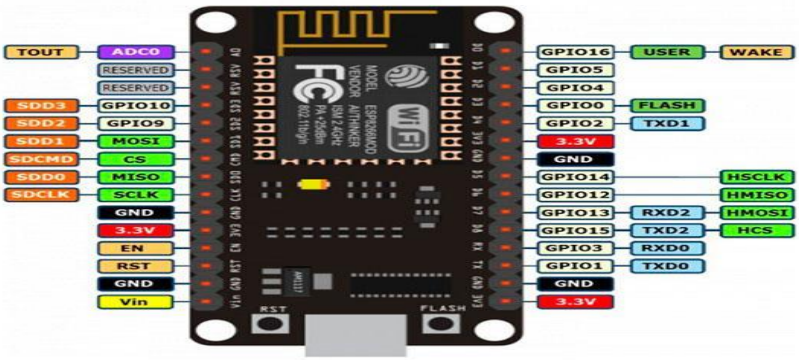
The NodeMCU development board has a total of 30 pins, which can be divided into two main categories: power and input/output (I/O) pins. Here is a brief description of each pin:

i.Power Pins:

- **VIN:** This pin is used to supply power to the board. It can accept voltages between 5V and 12V.
- **GND:** These pins are connected to ground and can be used to complete electrical circuits.

ii.Input/Output (I/O) Pins:

- **D0-D8:** These pins are digital I/O pins that can be used for both input and output. They are also capable of generating PWM signals.
- **A0:** This is an analog input pin that can be used to read analog signals from sensors.
- **TXD0 and RXD0:** These pins are used for serial communication.
- **SDA and SCL:** These pins are used for I2C communication.



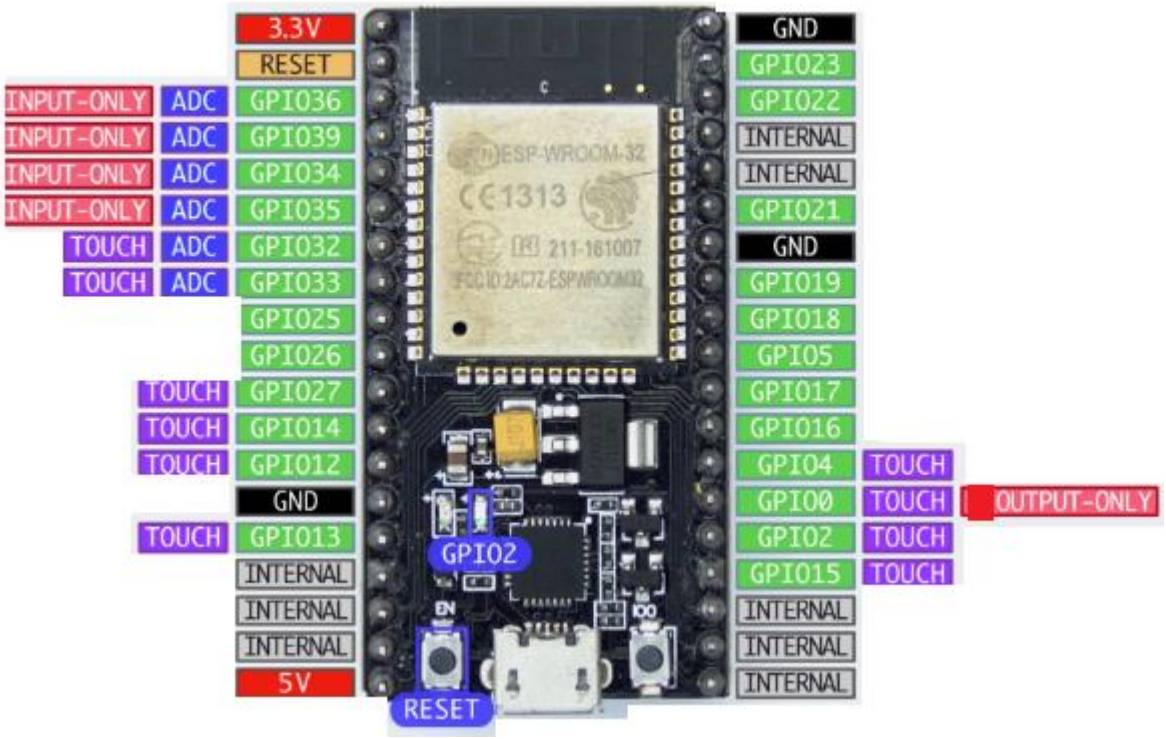
NODEMCU

TITLE: Introduction to NodeMCU ,ESP32 ,Rasberry Pi and Sensors

GPREC/DEEE/EXPT-AIOT (P)-INTRODUCTION **Date: 17/01/2023**

2.ESP32:

The NodeMCU ESP32 board (in some cases also known as ESP32-DevkitC) is fully supported by ESPHome. select ESP32 when the ESPHome wizard asks for type of platform and **nodemcu-32s** as the board type.



1. GPIO0 is used to determine the boot mode on startup. It should therefore not be pulled LOW on startup to avoid booting into flash mode. You can, however, still use this as an output pin.

2. GPIO34-GPIO39 can not be used as outputs (even though GPIO stands for "general purpose input output").
3. GPIO32-GPIO39: These pins can be used with the Analog To Digital Sensor to measure voltages.
4. GPIO2: This pin is connected to the blue LED on the board as seen in the picture above. It also supports the touch pad binary sensor as do the other pins marked touch in the above image.
5. 5V is connected to the 5V rail from the USB bus and can be used to power the board. Note that the UART chip is directly connected to this rail and you therefore cannot supply other voltages into this pin.

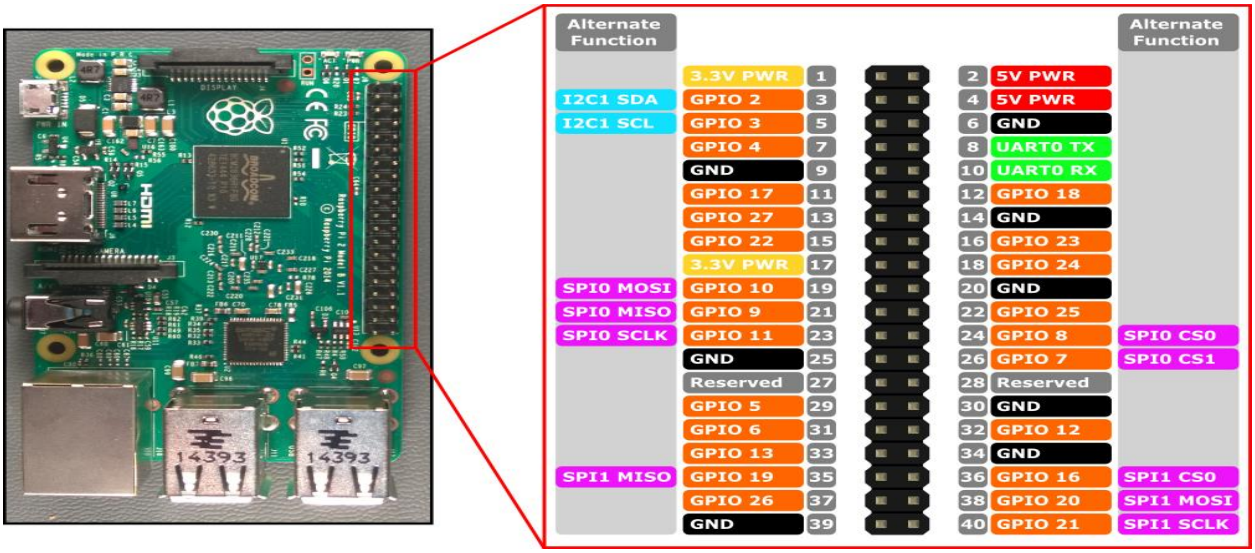
3. RASBERRY PI:

The Raspberry Pi is a small, affordable, single-board computer that can be used for a wide range of projects. It has a number of pins that can be used to connect various sensors, modules, and peripherals to the board. Here is a brief overview of the Raspberry Pi pin descriptions:

1. **Power Pins:** The Raspberry Pi has two power pins: 5V and 3.3V. The 5V pin can be used to power external devices that require 5V, while the 3.3V pin can be used to power sensors and other low-power devices.
2. **Ground Pins:** There are a number of ground pins on the Raspberry Pi board. These pins are used to complete the circuit and provide a common reference point for all connected devices.

TITLE: Introduction to NodeMCU ,ESP32 ,Rasberry Pi and Sensors

GPREC/DEEE/EXPT-AIOT (P)-INTRODUCTION **Date: 17/01/2023**



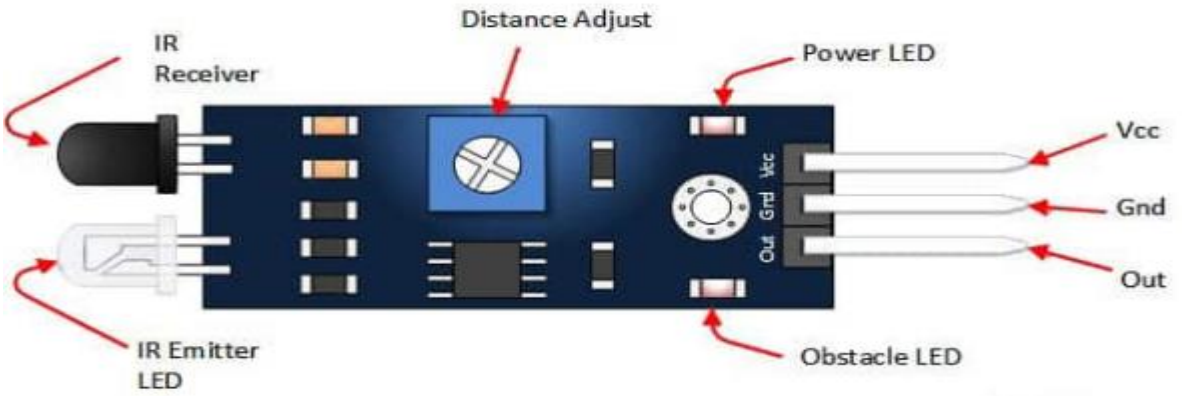
3. **GPIO Pins:** The GPIO (General Purpose Input/Output) pins are used to connect various sensors and modules to the Raspberry Pi. These pins can be configured as either input or output pins, depending on the needs of the project.
4. **UART Pins:** The Raspberry Pi has two UART (Universal Asynchronous Receiver/Transmitter) pins that can be used to communicate with other devices using the serial protocol.
5. **I2C Pins:** The I2C (Inter-Integrated Circuit) pins are used to connect sensors and other devices using the I2C protocol. The Raspberry Pi has two I2C pins: SDA (data) and SCL (clock).
6. **SPI Pins:** The SPI (Serial Peripheral Interface) pins are used to connect sensors and other devices using the SPI protocol. The Raspberry Pi has two SPI pins: MOSI (Master Out Slave In) and MISO (Master In Slave Out).
7. **PWM Pins:** The PWM (Pulse Width Modulation) pins are used to control the brightness of LEDs and the speed of motors.

8. **Camera Connector:** The Raspberry Pi has a dedicated camera connector that can be used to connect a Raspberry Pi Camera Module.
9. **Display Connector**: The Raspberry Pi has a dedicated display connector that can be used to connect a Raspberry Pi Display Module.

4.IR SENSOR:

The pin configuration or pin description of an IR sensor may vary depending on the specific type and manufacturer of the sensor. However, here is a general pin description of an IR sensor:

1. VCC or VDD: This pin is used to supply power to the sensor. It is usually connected to a 5V DC power source.
2. GND: This pin is used to connect the sensor to the ground or negative terminal of the power source.
3. OUT or SIGNAL: This pin is the output pin of the sensor. It is used to send a signal to the microcontroller or other electronic devices based on the detection of infrared radiation. The signal can be either analog or digital, depending on the type of sensor.
4. NC (Not Connected): This pin is not used and remains unconnected.



TITLE: Introduction to NodeMCU ,ESP32 ,Rasberry Pi and Sensors

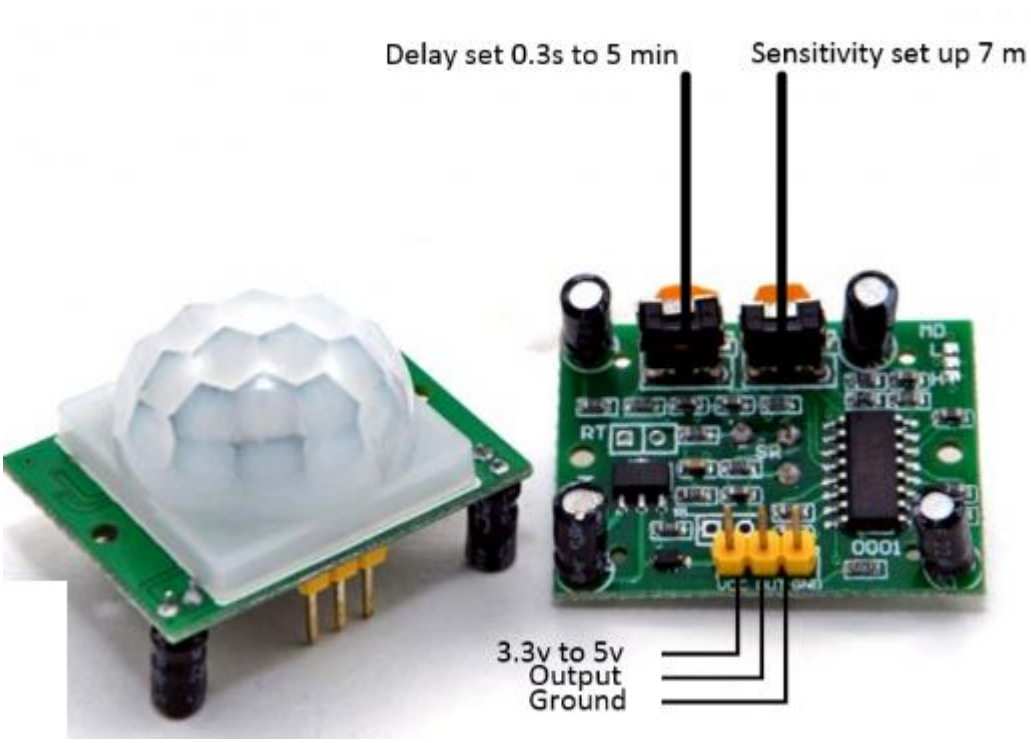
GPREC/DEEE/EXPT-AIOT (P)-INTRODUCTION **Date: 17/01/2023**

5.PIR SENSOR:

A PIR (Passive Infrared) sensor typically has three pins:

1. VCC (or +): This pin is used to supply power to the sensor. It is usually connected to a 5V power source.
2. GND (or -): This pin is the ground or negative terminal of the sensor. It should be connected to the ground of the circuit.
3. OUT (or Signal): This pin is used to output the detection signal. When the sensor detects motion or changes in the infrared radiation in its field of view, it sends a signal to this pin. The signal can be either a digital signal (high or low) or an analog signal (varying voltage level), depending on the sensor's configuration.

It is important to check the datasheet of the specific PIR sensor you are using to ensure the correct pinout and proper usage.



TITLE: Introduction to NodeMCU ,ESP32 ,Rasberry Pi and Sensors

GPREC/DEEE/EXPT-AIOT (P)-INTRODUCTION **Date: 17/01/2023**

6.Temperature and Humidity Sensor:

The DHT sensor is a popular type of digital temperature and humidity sensor used in various applications. The sensor usually has four pins:

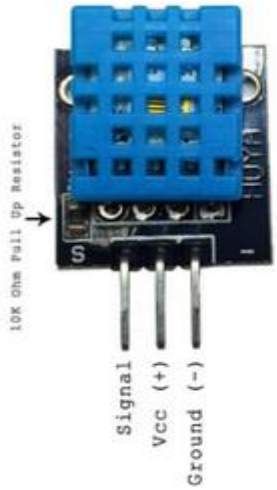
1. VCC (or +): This pin is used to supply power to the sensor. It is usually connected to a 3.3V or 5V power source.
2. GND (or -): This pin is the ground or negative terminal of the sensor. It should be connected to the ground of the circuit.
3. DATA: This pin is used to transfer digital data from the sensor to the microcontroller. It is usually connected to a digital input/output (I/O) pin of the microcontroller.
4. NC (or Not Connected): This pin is not used and can be left unconnected.

The DHT sensor outputs digital data that includes the temperature and humidity values in a specific format. The data is transmitted through the data pin in a serial format that can be read by the microcontroller using a specific protocol.

It is important to note that different versions of the DHT sensor may have slightly different pinouts or communication protocols. Therefore, it is important to check the datasheet of the specific DHT sensor you are using to ensure the correct pinout and proper usage.

TITLE: Introduction to NodeMCU ,ESP32 ,Rasberry Pi and Sensors

GPREC/DEEE/EXPT-AIOT (P)-INTRODUCTION **Date: 17/01/2023**



7.ULTRASONIC SENSOR:

An ultrasonic sensor is a device that can measure the distance to an object by emitting high-frequency sound waves and detecting the time it takes for the sound waves to bounce back. The sensor typically has four pins:

1. VCC (or +): This pin is used to supply power to the sensor. It is usually connected to a 5V power source.
2. GND (or -): This pin is the ground or negative terminal of the sensor. It should be connected to the ground of the circuit.
3. TRIG: This pin is used to trigger the ultrasonic pulse. When this pin receives a high signal, the sensor emits a pulse.
4. ECHO: This pin is used to receive the echo signal that is generated when the ultrasonic pulse bounces back from an object. The time it takes for the pulse to return can be used to calculate the distance to the object.

The TRIG and ECHO pins usually operate on 5V logic levels, so it is important to use appropriate level-shifting circuitry if connecting the sensor to a microcontroller that operates on lower voltage levels.

TITLE: Introduction to NodeMCU ,ESP32 ,Rasberry Pi and Sensors

GPREC/DEEE/EXPT-AIOT (P)-INTRODUCTION **Date: 17/01/2023**

It is important to check the datasheet of the specific ultrasonic sensor you are using to ensure the correct pinout and proper usage.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE: Interface Ultrasonic sensor with NodeMCU and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-1 **Date: 17/01/2023**

OBJECTIVE:

To interface Ultrasonic sensor with NodeMCU and display the sensor output.

APPARATUS:

1. NodeMCU board
2. Ultrasonic sensor (HC-SR04)
3. Jumper wires

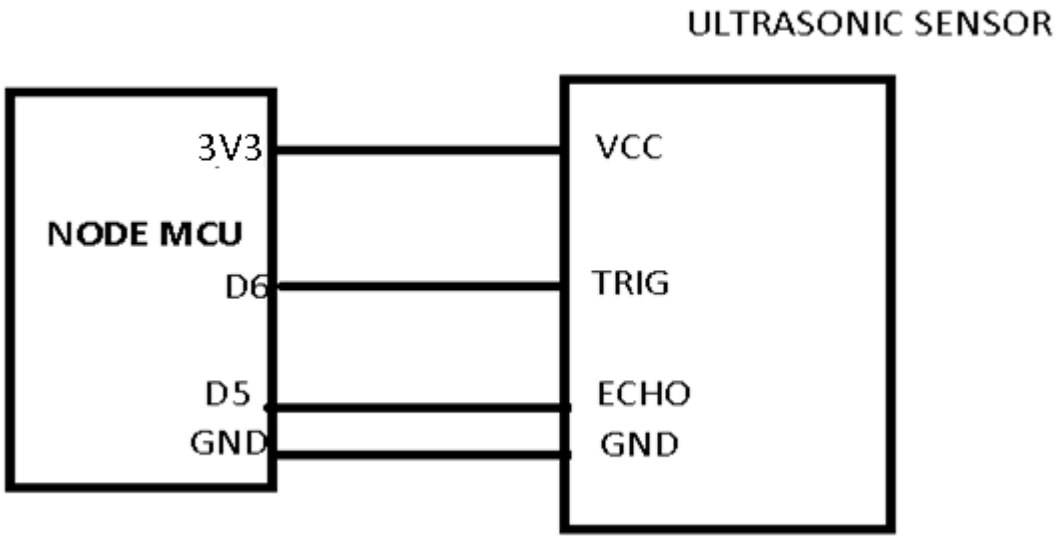
ALGORITHM:

1. Initialize the serial communication at a baud rate of 9600 bits per second using the **Serial.begin()** function.
2. Define the Trigger and Echo pins of the Ultrasonic sensor as digital output and input pins, respectively, using the **pinMode()** function.
3. In the loop function, send a 10 microsecond pulse to the Trigger pin of the Ultrasonic sensor using the **digitalWrite()** function.
4. Wait for the pulse to return to the Echo pin of the Ultrasonic sensor and measure its duration in microseconds using the **pulseIn()** function.
5. Calculate the distance to the object in centimeters by dividing the duration by 58, which is the time it takes for the sound to travel 1 centimeter.
6. Print the distance to the object on the serial monitor using the **Serial.println()** function.
7. Wait for a short delay before taking the next measurement using the **delay()** function.

TITLE: Interface Ultrasonic sensor with NodeMCU and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-1 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. What are the features of ESP8266?

2. List the pin numbers that can be used as a GPIO in ESP8266.

3.what is function of Serial.begin() .

4. what is function of digitalWrite().

5. what is function of pinmode().

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE : Interface temperature and humidity sensor with NodeMCU and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-2 **Date: 17/01/2023**

OBJECTIVE:

To interface temperature and humidity sensor with NodeMCU and display the sensor output.

APPARATUS:

1. NodeMCU board
2. Temperature and humidity sensor(DHT11)
3. Jumper wires

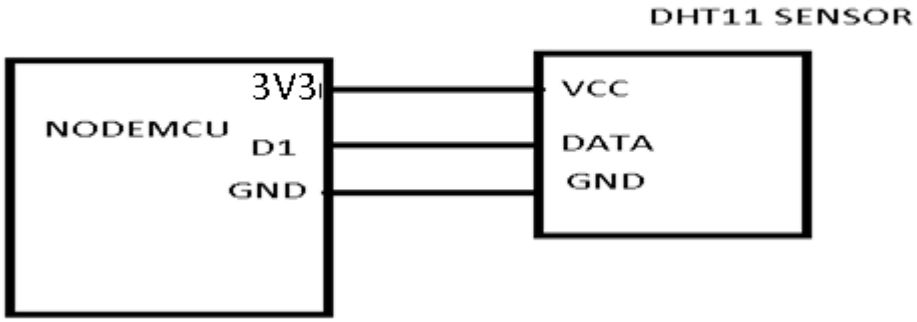
ALGORITHM:

1. Define the data pin to which the sensor is connected and the type of sensor.

2. In the setup() function, initialize the serial communication and the DHT sensor

3. The loop() function, read the temperature and humidity values from the sensor using the readTemperature() and readHumidity() methods of the DHT object, and print them to the serial monitor using the Serial.print()

CONNECTION DIAGRAM:



TITLE : Interface temperature and humidity sensor with NodeMCU and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-2 **Date: 17/01/2023**

RESULT:

QUESTIONS:

1. list any two application of temperature and humidity sensor with NodeMCU .

2. List the pin numbers that can be used as a GPIO in ESP8266 for inertfacing DHT sensor.

3.Which libraries should be imported for working of DHT sensor.

4. What is baud rate.

5. What is function of dht.begin()

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE : Interface PIR sensor with NodeMCU and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-3 **Date: 17/01/2023**

OBJECTIVE:

To interface PIR sensor with NodeMCU and display the sensor output.

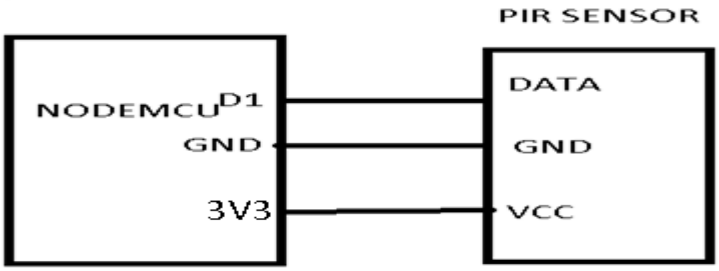
APPARATUS:

1. NodeMCU board
2. PIR sensor
3. Jumper wires

ALGORITHM:

1. Define the PIR pin.
2. Initialize the serial monitor.
3. Set the PIR pin as input.
4. Read the PIR sensor output .
5. Print the sensor output to the serial monitor.
6. Add a delay for 500 milliseconds .

CONNECTION DIAGRAM:



TITLE : Interface PIR sensor with NodeMCU and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-3 **Date: 17/01/2023**

RESULT:

QUESTIONS:

1. List any two application of PIR sensor with NodeMCU .

2. List the pin numbers that can be used as a GPIO in ESP8266 for inertfacing PIR sensor sensor.

3.What is function of void loop.

4. What is function is used to read the data from PIR sensor

5. What is function of serial monitor.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE:Interface IR sensor with NodeMCU and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-4 **Date: 17/01/2023**

OBJECTIVE:

To interface IR sensor with NodeMCU and display the sensor output.

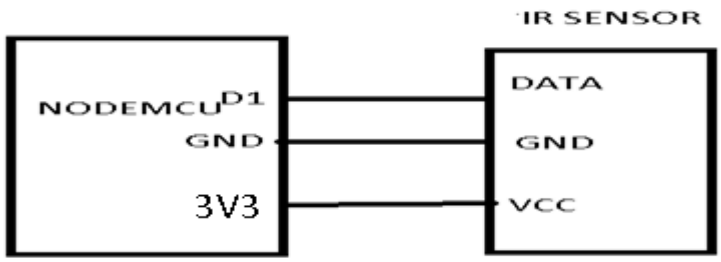
APPARATUS:

1. NodeMCU board
2. IR sensor
3. Jumper wires

ALGORITHM:

1. Define the IR_PIN as D1 and set it as an input pin.
2. In the **loop()** function, read the output of the IR sensor using the **digitalRead()** function and store the value in the **irValue** variable.
3. Then print the value of **irValue** to the serial monitor using the **Serial.print()** and **Serial.println()** functions.
4. Finally, add a delay of 500 milliseconds using the **delay()** function to avoid

CONNECTION DIAGRAM:



G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE:Interface IR sensor with NodeMCU and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-4 **Date: 17/01/2023**

RESULT:

QUESTIONS:

1. List any two application of IR sensor with NodeMCU .

2. List the pin numbers that can be used as a GPIO in ESP8266 for inertfacing IR sensor sensor.

3.what is function of analoglread ().

4. what is function is used to read the data from IR sensor

5. what is function of Serial.println().

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Interface Array of LEDs with NodeMCU and a) turn on ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left.

GPREC/DEEE/EXPT-AIOT (P)-5 **Date: 17/01/2023**

OBJECTIVE:

To Interface Array of LEDs with NodeMCU and a) turn on ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left.

APPARATUS:

1. NodeMCU board
2. Array of LEDS
3. Jumper wires

ALGORITHM: **a) Turn on ALL LEDs once**

1.Define the LED pins.

2. Set all LED pins as output.

3. Turn on all the LEDs at once by setting all the pins to HIGH using a for loop.

4. Wait for one second using the delay() function before turning off all the LEDs by setting

all the pins to LOW.

5. Give a delay of ONE second again before starting the loop again

ALGORITHM: **b) Turn on LEDs one by one from left to right and then from right to left.**

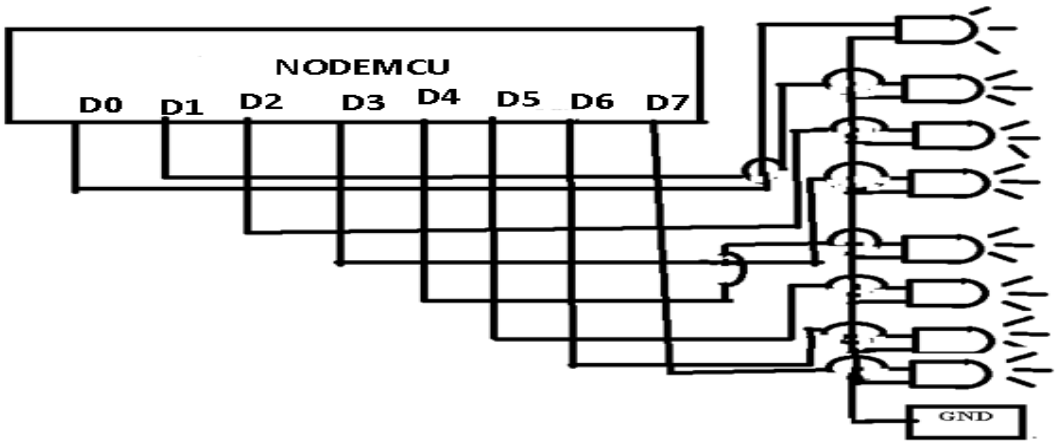
1. First define an array ledPins that contains the digital output pins that we connected the LEDs to**.**
2. In the setup() function, set all the pins in the ledPins array as output pins using the pinMode() function.
3. Turn on the LEDs one by one from left to right using a for loop and the delay() function to create a delay between each LED turning on.
4. Turn off all the LEDs and wait for 500 milliseconds using the delay() function.

TITLE: Interface Array of LEDs with NodeMCU and a) turn on ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left.

GPREC/DEEE/EXPT-AIOT (P)-5 **Date: 17/01/2023**

5. Turn on the LEDs one by one from right to left using a for loop and the delay() function to create a delay between each LED turning on.
6. Turn off all the LEDs again and wait for 500 milliseconds before starting the loop.

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. List any two application of array LEDS with NodeMCU .

2. List the pin numbers that can be used as a GPIO in ESP8266 for inertfacing array of LEDs

3.What is function of delay ().

4. Write a logic to blink out of 8 leds four blink at a time and remaining four at another time.

TITLE: Interface Array of LEDs with NodeMCU and a) turn on ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left.

GPREC/DEEE/EXPT-AIOT (P)-5 **Date: 17/01/2023**

5. What kind of device is the ESP8266 Wi-Fi module.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE: Interface an Array of LEDs with NodeMCU and turn on LEDs to display a hexadecimal number equivalent value from 00 to FF.

GPREC/DEEE/EXPT-AIOT (P)-6 **Date: 17/01/2023**

OBJECTIVE:

To interface an Array of LEDs with NodeMCU and turn on LEDs to display a hexadecimal number equivalent value from 00 to FF.

APPARATUS:

1. NodeMCU board
2. Array of LEDS
3. Jumper wires

ALGORITHM:

1. Define a constant NUM_LEDS to specify the number of LEDs in the array.
2. Define an array LED_PINS with NUM_LEDS elements to specify the pin numbers where the LEDs are connected.
3. In the setup() function:

   a. Use a for loop to iterate over all the elements in the LED_PINS array.

   b. For each element, use the pinMode() function to set the corresponding pin as an output.

4. In the loop() function: a. Use a for loop to iterate over the values from 0 to 255 (hexadecimal 00 to FF).

   b. For each iteration of the outer loop: i. Use another for loop to iterate over all the elements in the LED_PINS array. ii. For each element, use the bitRead() function to read the corresponding bit of the current value. iii. Use the digitalWrite() function to set the corresponding LED pin to HIGH or LOW based on the value of the bit.

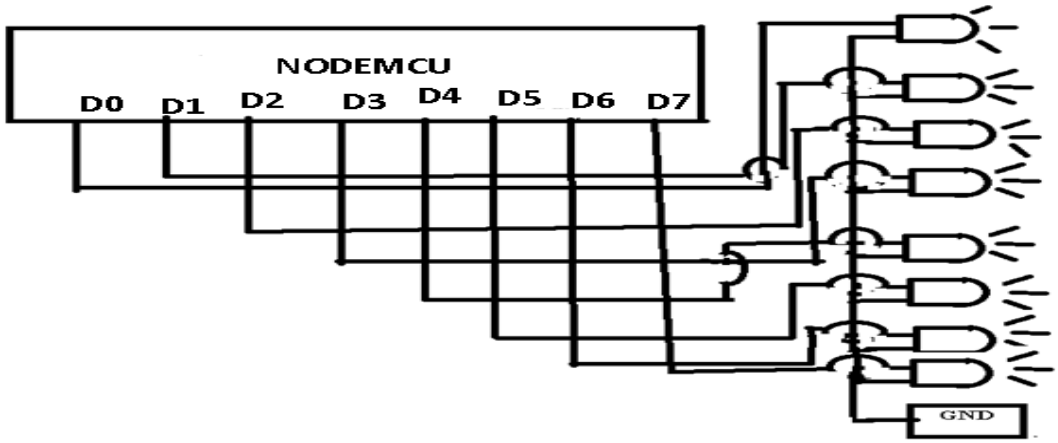
   c. Use the delay() function to wait for a certain amount of time (e.g., 1 second) before displaying the next value

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE: Interface an Array of LEDs with NodeMCU and turn on LEDs to display a hexadecimal number equivalent value from 00 to FF.

GPREC/DEEE/EXPT-AIOT (P)-6 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. How many pins are present in the ESP8266 WiFi Module?

2. What is the maximum source current that is required to operate the ESP8266 WiFi Module?

3. What is the optimum supply current that is required to operate the ESP8266 WiFi Module?

4. What is the type of waves that the ESP8266 WiFi Module detects?.

5. What is the use of the ESP8266 WiFi Module?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Interfacing DC motor with NodeMCU.

GPREC/DEEE/EXPT-AIOT (P)-7 **Date: 17/01/2023**

OBJECTIVE:

Interfacing DC motor with NodeMCU.

APPARATUS:

1. NodeMCU board
2. L293D motor driver IC
3. DC motor

ALGORITHM:

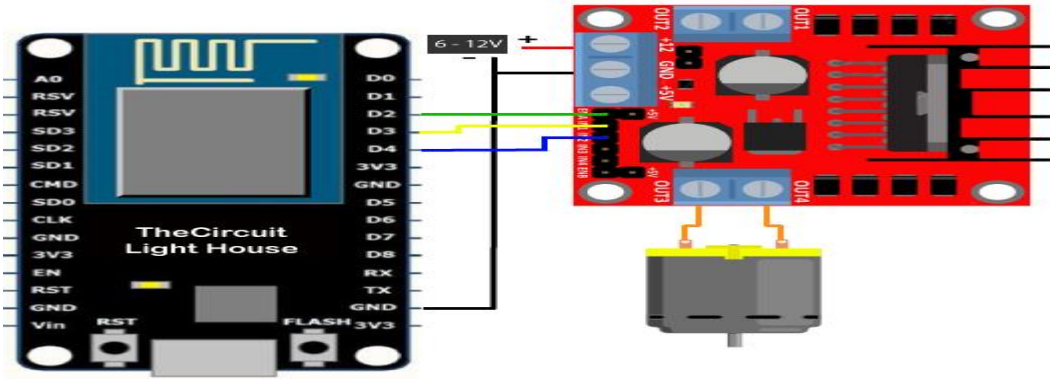
1. First define the IN1, IN2, and ENA(depend on driver circuit) pins to match the wiring configuration.

2. Setup function, set the pins as output.

3. In the loop function, we use analogWrite() function to set the ENA pin to half of the maximum value (255). This sets the motor speed to half of the maximum.

4. Use digitalWrite() function to set the IN1 and IN2 pins to move the motor forward for 2 seconds, stop the motor for 1 second, move the motor backward for 2 seconds, and stop the motor for 1 second.

CONNECTION DIAGRAM:



TITLE: Interfacing DC motor with NodeMCU.

GPREC/DEEE/EXPT-AIOT (P)-7 **Date: 17/01/2023**

Note: Make sure to use a separate power supply for the motor, and connect the ground of the power supply to the ground of NodeMCU. Also, make sure to use a suitable motor driver module based on the motor specifications.

RESULT:

QUESTIONS:

1.What is the use of the Motor Driver (L293D) Module?

2. How many pins are present in the Motor Driver (L293D) Module?

3.What is the use of the Enable pin.

4.write a logic to run motor in anti clock wise direction in four seconds then in clock wise for three seconds.

5. What is the use of the Vcc pin in L293D?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Interface Ultrasonic sensor with NodeMCU and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-8 **Date: 17/01/2023**

OBJECTIVE:

To interface Ultrasonic sensor with NodeMCU and display the sensor output.

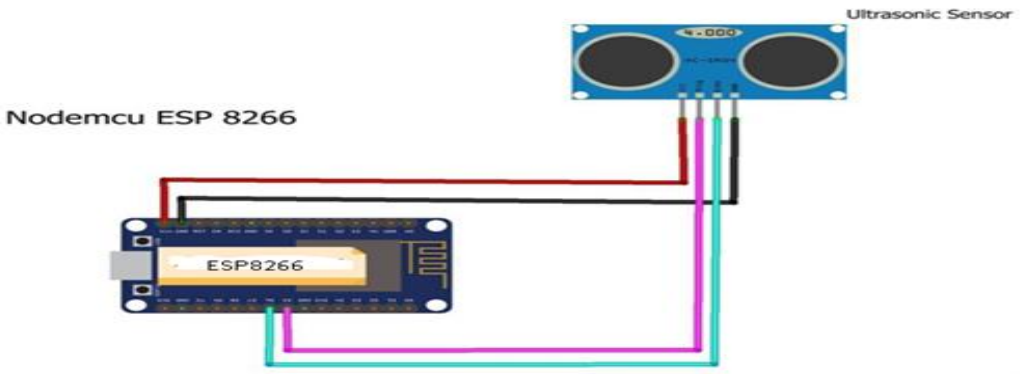
APPARATUS:

1. NodeMCU board
2. Ultrasonic sensor (HC-SR04)
3. Jumper wires

ALGORITHM:

1. Define the Trig and Echo pins as constants
2. In the setup function, initialize the Serial communication and set the Trig and Echo pins as output and input respectively
3. In the loop function, send a 10 microsecond pulse to the Trig pin to initiate the Ultrasonic Sensor measurement
4. Measure the time duration of the pulse using the pulseIn function and calculate the distance using the formula distance = duration * 0.034 / 2
5. Display the distance value on the Serial Monitor.

CONNECTION DIAGRAM:



RESULT:

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Interface Ultrasonic sensor with NodeMCU and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-8 **Date: 17/01/2023**

QUESTIONS:

1. List any two application of Ultrasonic sensor with NodeMCU .
2. How many pins are present in the Ultrasonic Sensor?
3. What will happen if we supply a voltage of 25V to the Vcc of the Ultrasonic sensor?
4. what is function of ECHO pin in ultrasonic sensor
5. What kind of waves does the Ultrasonic Sensor work on?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE:Interface temperature and humidity sensor to ESP32 and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-9 **Date: 17/01/2023**

OBJECTIVE:

To interface temperature and humidity sensor to ESP32 and display the sensor output..

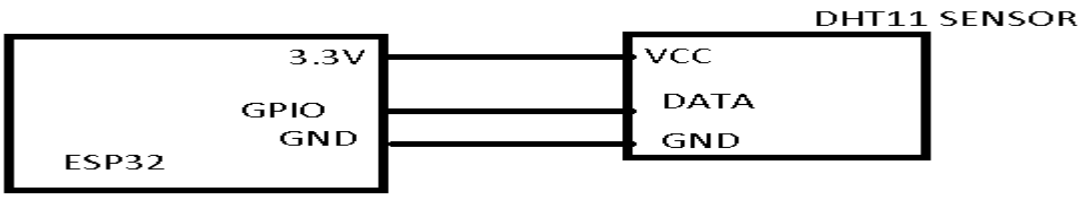
APPARATUS:

1. ESP32 board
2. Temperature and humidity sensor
3. Jumper wires

ALGORITHM:

1. Initialize the ESP32 and the sensor
2. Define the pin configurations of the sensor and the ESP32
3. Set up the serial communication to display the sensor readings
4. Continuously read the temperature and humidity data from the sensor
5. Print the temperature and humidity data to the serial monitor
6. Delay for a certain amount of time before taking the next reading
7. Repeat steps 4-6 indefinitely.

CONNECTION DIAGRAM:



TITLE:Interface temperature and humidity sensor to ESP32 and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-9 **Date: 17/01/2023**

RESULT:

QUESTIONS:

1. list any two application of temperature and humidity sensor with ESP32 .

2. List the pin numbers that can be used as a GPIO in ESP32 for inertfacing DHT sensor.

3.Which libraries should be imported for working of DHT sensor.

4. What is baud rate.

5. What is function of dht.begin()

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Interface PIR sensor with ESP32 and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-10 **Date: 17/01/2023**

OBJECTIVE:

To interface PIR sensor with ESP32 and display the sensor output..

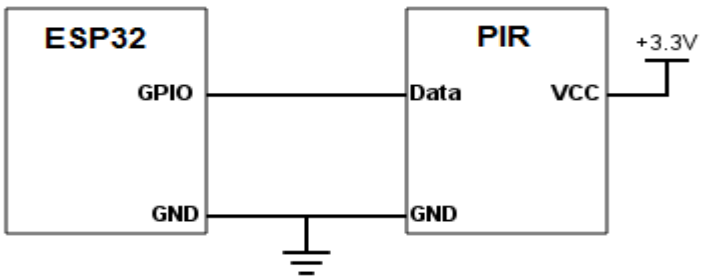
APPARATUS:

1. ESP32 board
2. PIR sensor
3. Jumper wires

ALGORITHM:

1. Define the Out pin as a constant
2. In the setup function, initialize the Serial communication and set the Out pin as an input
3. In the loop function, read the value of the Out pin using the digitalRead function and store it in a variable
4. Display the PIR Sensor output value on the Serial Monitor using the Serial.print and Serial.println functions
5. Add a delay to the loop function to control the frequency of the sensor output display

ONNECTION DIAGRAM:



TITLE: Interface PIR sensor with ESP32 and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-10 **Date: 17/01/2023**

RESULT:

QUESTIONS:

1. List any two application of PIR sensor with ESP32 .

2. List the pin numbers that can be used as a GPIO in ESP32 for inertfacing PIR sensor sensor.

3.What is function of digitalread.

4. Whar are the terminal pins of PIR sensor.

5.what is the working mechanism of PIR sensor.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE: Interface IR sensor with ESP32 and display the sensor output.
GPREC/DEEE/EXPT-AIOT (P)-11 **Date: 17/01/2023**

OBJECTIVE:

To interface IR sensor with ESP32 and display the sensor output.

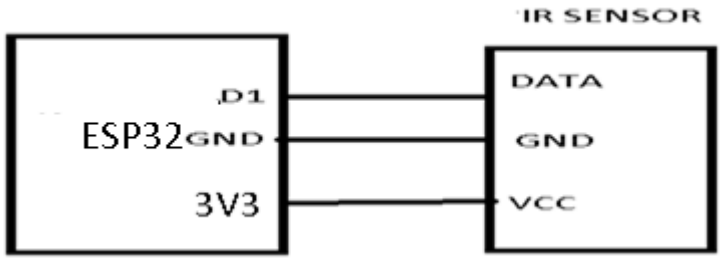
APPARATUS:

1. ESP32 board
2. IR sensor
3. Jumper wires

ALGORITHM:

1. Define the pin number where the IR sensor is connected.
2. Declare a variable to store the sensor reading
3. initialize serial communication at 9600 baud
4. Read the sensor value
5. Print the sensor value on a new line
6. wait for 100 milliseconds before reading the sensor again

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. List any two application of IR sensor with ESP32 .

TITLE: Interface IR sensor with ESP32 and display the sensor output.
GPREC/DEEE/EXPT-AIOT (P)-11 **Date: 17/01/2023**

2. List the pin numbers that can be used as a GPIO in ESP32 for inertfacing IR sensor sensor.

3.what is function used for reading the sensor value.

4. what are the terminal pins of IR sensor

5. what is the working mechanism of IR sensor.

TITLE: Interface an Array of LEDs with ESP32 and a) turn on the ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left

GPREC/DEEE/EXPT-AIOT (P)-12 **Date: 17/01/2023**

OBJECTIVE:

To interface an Array of LEDs with ESP32 and a) turn on the ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left

APPARATUS:

1. ESP32board
2. Array of LEDS
3. Jumper wires

ALGORITHM:

1. Define a constant **NUM_LEDS** to specify the number of LEDs in the array.
2. Define an array **LED_PINS** with **NUM_LEDS** elements to specify the pin numbers where the LEDs are connected.
3. In the **setup()** function:

   a. Use a **for** loop to iterate over all the elements in the **LED_PINS** array.

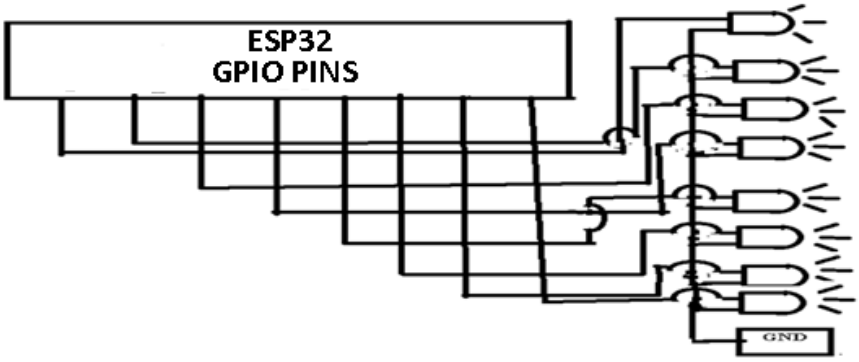
   b. For each element, use the **pinMode()** function to set the corresponding pin as an output.
4. In the **loop()** function:
5. a. Use a **for** loop to iterate over all the elements in the **LED_PINS** array.

   b. For each element, use the **digitalWrite()** function to set the corresponding pin to **HIGH** to turn on the LED.and to trun off set the corresponding pin to LOW.

TITLE: Interface an Array of LEDs with ESP32 and a) turn on the ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left

GPREC/DEEE/EXPT-AIOT (P)-12 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. List any two application of array LEDS with ESP32 .

2. List the pin numbers that can be used as a GPIO in ES32 for inertfacing array of LEDs

3.What is function of digital write.

4. Write a logic to blink out of 8 leds three blink at a time and remaining five at another time.

5.How many number of GPIO ports are available in ESP32.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE: Interface an Array of LEDs with ESP32 and turn on LEDs to display a hexadecimal number equivalent value from 00 to FF.

GPREC/DEEE/EXPT-AIOT (P)-13 **Date: 17/01/2023**

OBJECTIVE:

To interface an Array of LEDs with ESP32 and turn on LEDs to display a hexadecimal number equivalent value from 00 to FF.

APPARATUS:

1. ESP32 board
2. Array of LEDS
3. Jumper wires

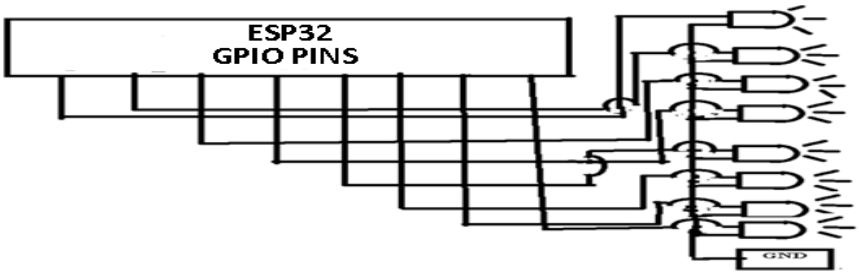
ALGORITHM:

1. Define a constant NUM_LEDS to specify the number of LEDs in the array.
2. Define an array LED_PINS with NUM_LEDS elements to specify the pin numbers where the LEDs are connected.
3. In the setup() function: a. Use a for loop to iterate over all the elements in the LED_PINS array. b. For each element, use the pinMode() function to set the corresponding pin as an output.
4. In the loop() function: a. Use a for loop to iterate over the values from 0 to 255 (hexadecimal 00 to FF). b. For each iteration of the outer loop: i. Use another for loop to iterate over all the elements in the LED_PINS array. ii. For each element, use the bitRead() function to read the corresponding bit of the current value. iii. Use the digitalWrite() function to set the corresponding LED pin to HIGH or LOW based on the value of the bit. c. Use the delay() function to wait for a certain amount of time (e.g., 1 second) before displaying the next value

TITLE: Interface an Array of LEDs with ESP32 and turn on LEDs to display a hexadecimal number equivalent value from 00 to FF.

GPREC/DEEE/EXPT-AIOT (P)-13 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. List any two application of array LEDS with ESP32 .
2. List the pin numbers that can be used as a GPIO in ESP32 for inertfacing array of LEDs
3. What is function of delay ().
4. Write a logic to blink out of 8 leds two at a time and remaining six at another time.
5. Which pins are used only for input purpose only.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE : To interface DC motor with ESP32.

GPREC/DEEE/EXPT-AIOT (P)-14 **Date: 17/01/2023**

OBJECTIVE:

To interface DC motor with ESP32.

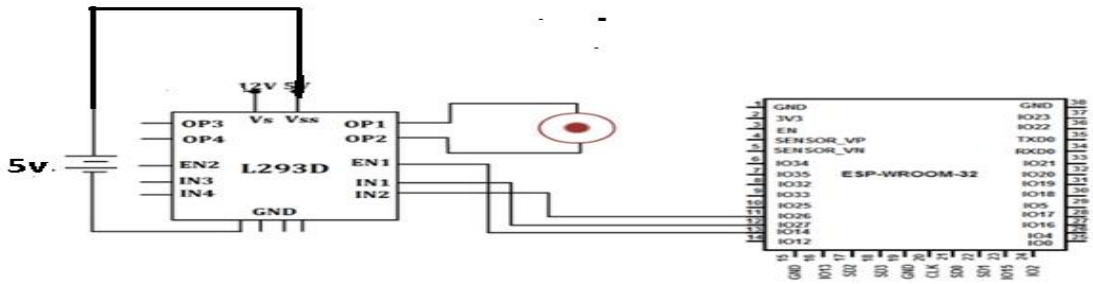
APPARATUS:

1. ESP32 board
2. DC motor
3. L293D
4. Jumper wires

ALGORITHM:

1. Choose a GPIO pin on the ESP32 to connect to the motor's input signal.
2. In the setup() function:
    a. Use the pinMode() function to set the chosen GPIO pin as an output.
3. In the loop() function:
    a. Use the analogWrite() function to set the output voltage of the chosen GPIO pin.
    b. Choose a value between 0 and 255 to represent the desired motor speed. A value of 0 means the motor is off, and a value of 255 means the motor is running at full speed.
    c. Use a delay() function to pause the loop for a short period of time before updating the motor speed again.

CONNECTION DIAGRAM:



TITLE : To interface DC motor with ESP32.

GPREC/DEEE/EXPT-AIOT (P)-14 **Date: 17/01/2023**

RESULT:

QUESTIONS:

1.What is the use of the Motor Driver (L293D) Module?

2. How many pins are present in the Motor Driver (L293D) Module?

3.What is the use of the Enable pin.

4.write a logic to run motor in anti clock wise direction in four seconds then in clock wise for three seconds.

5. What is the use of the Vcc pin in L293D?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: To interface Ultrasonic sensor with Raspberry PI and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-15 **Date: 17/01/2023**

OBJECTIVE:

To interface Ultrasonic sensor with Raspberry PI and display the sensor output

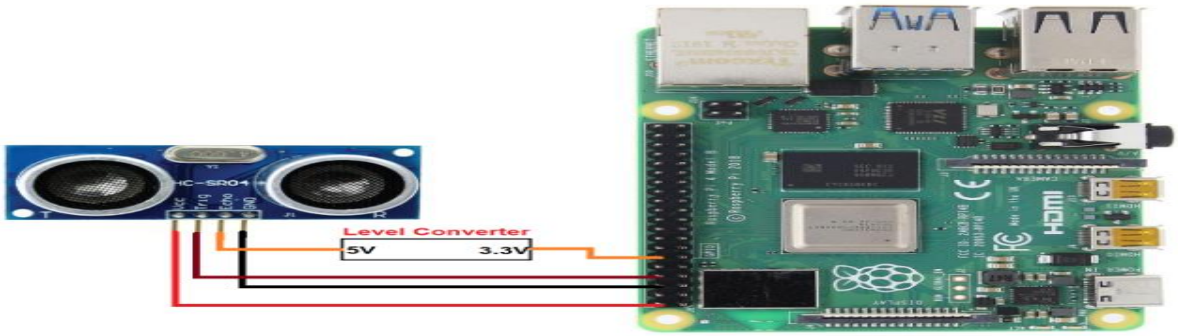
APPARATUS:

1. Raspberry PI board
2. Ultrasonic sensor (HC-SR04)
3. Jumper wires

ALGORITHM:

1. Import required libraries: **import RPi.GPIO as GPIO** and **import time**
2. Set GPIO mode and pins: **GPIO.setmode(GPIO.BCM)** and **GPIO.setup(23, GPIO.OUT)** and **GPIO.setup(24, GPIO.IN)**
3. Call the **read_distance()** function and print the output: **print("Distance: %.2f cm" % read_distance())**
4. Save the python file with a suitable name, for example ,ultrasonic_sensor.py
5. Run the python file and observe the distance measured by the Ultrasonic Sensor printed on the terminal.

CONNECTION DIAGRAM:



TITLE: To interface Ultrasonic sensor with Raspberry PI and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-15 **Date: 17/01/2023**

RESULT:

QUESTIONS:

1. What is an ultrasonic sensor and how does it work?
2. How do you connect an ultrasonic sensor to a Raspberry Pi, and what types of pins are used?
3. Can you explain the difference between digital and analog output modes on an ultrasonic sensor, and how they can be used with a Raspberry Pi?
4. How do you write code in Python to read data from an ultrasonic sensor connected to a Raspberry Pi, and what libraries might you use?
5. What are some potential applications for an ultrasonic sensor connected to a Raspberry Pi, and how might you expand upon this project to build a more complex system?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Interface temperature and humidity sensor with Raspberry PI and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-16 **Date: 17/01/2023**

OBJECTIVE:

To interface temperature and humidity sensor with Raspberry PI and display the sensor output

APPARATUS:

1. Rasberry pi board
2. Temperature and humidity sensor
3. Jumper wires

ALGORITHM:

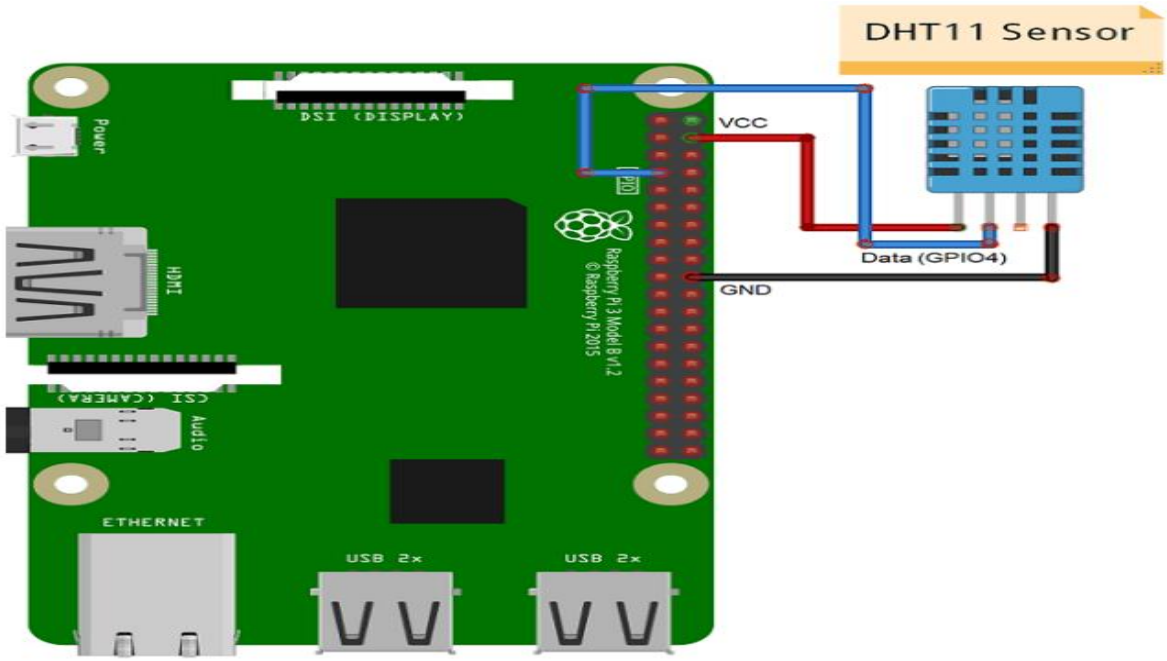
1. Import the required library: **import Adafruit_DHT**
2. Define the sensor type and GPIO pin: **sensor = Adafruit_DHT.DHT11** and **pin = 17**
3. Use the **read_retry()** function to read sensor data: **humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)**
4. Check if the sensor data is valid: **if humidity is not None and temperature is not None:**
5. Print the sensor data: **print('Temperature: {0:0.1f} C Humidity: {1:0.1f} %'.format(temperature, humidity))**
6. Save the Python file with a suitable name, for example, **temp_humidity_sensor.py**
7. Run the Python file using the command: **python3 temp_humidity_sensor.py**
8. The temperature and humidity measured by the sensor will be displayed on the terminal.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: Interface temperature and humidity sensor with Raspberry PI and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-16 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. What is a temperature and humidity sensor, and how does it work?
2. How do you connect a temperature and humidity sensor to a Raspberry Pi, and what types of pins are used?
3. Can you explain the difference between digital and analog output modes on a temperature and humidity sensor, and how they can be used with a Raspberry Pi?
4. How do you write code in Python to read data from a temperature and humidity sensor connected to a Raspberry Pi, and what libraries might you use?
5. What are some potential applications for a temperature and humidity sensor connected to a Raspberry Pi, and how might you expand upon this project to build a more complex system?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE: To interface PIR sensor with Raspberry PI and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-17 **Date: 17/01/2023**

OBJECTIVE:

To interface PIR sensor with Raspberry PI and display the sensor output

APPARATUS:

1. Raspberry PI board
2. PIR sensor
3. Jumper wires

ALGORITHM:

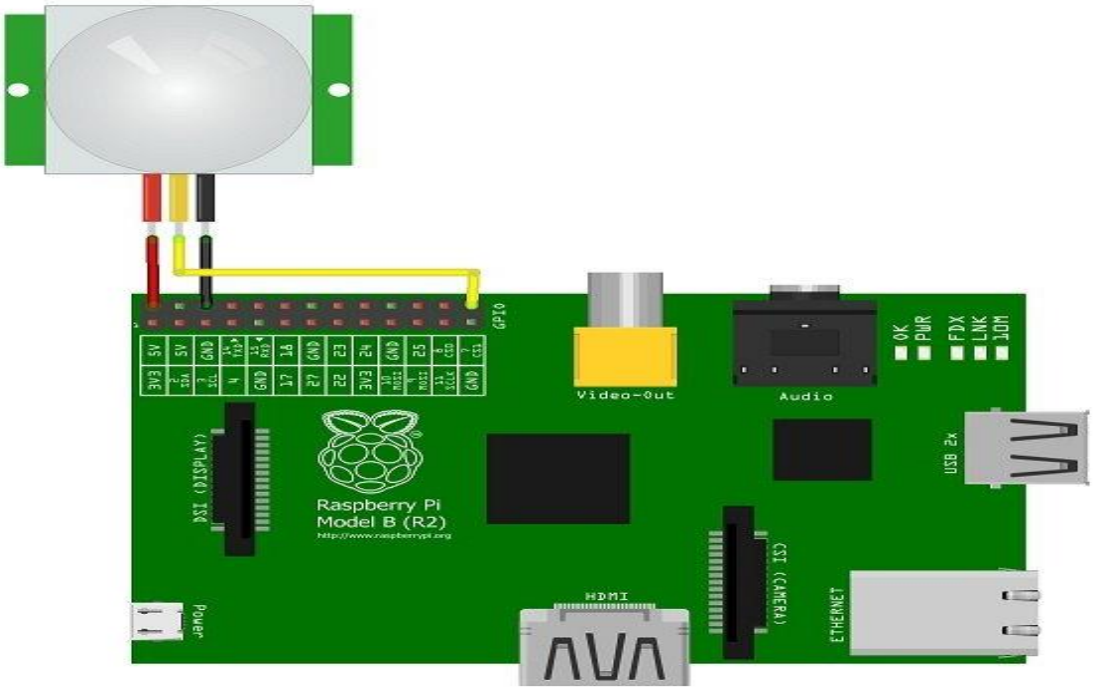
1. Import the required library: **import RPi.GPIO as GPIO**
2. Set GPIO mode and pin: **GPIO.setmode(GPIO.BCM)** and **GPIO.setup(18, GPIO.IN)**
3. Define a function to read sensor output
4. Call the **read_motion()** function to read sensor output: **read_motion()**
5. Save the Python file with a suitable name, for example, **pir_sensor.py**
6. Run the Python file using the command: **python3 pir_sensor.py**
7. The sensor output will be displayed on the terminal. If motion is detected, the message "Motion Detected!" will be printed, otherwise, the message "No Motion Detected." will be printed.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY(AIOT(P))

TITLE: To interface PIR sensor with Raspberry PI and display the sensor output

GPREC/DEEE/EXPT-AIOT (P)-17 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. What is a PIR sensor, and how does it work?
2. How do you connect a PIR sensor to a Raspberry Pi, and what types of pins are used?
3. Can you explain how to read data from a PIR sensor connected to a Raspberry Pi using Python, and what libraries might you use?
4. What is the purpose of displaying the output of a PIR sensor, and how might you do so using a Raspberry Pi?
5. What are some potential applications for a PIR sensor connected to a Raspberry Pi, and how might you expand upon this project to build a more complex system?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: To interface IR sensor with Raspberry PI and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-18 **Date: 17/01/2023**

OBJECTIVE:

To interface IR sensor with Raspberry PI and display the sensor output.

APPARATUS:

1. Raspberry PI board
2. IRsensor
3. Jumper wires

ALGORITHM:

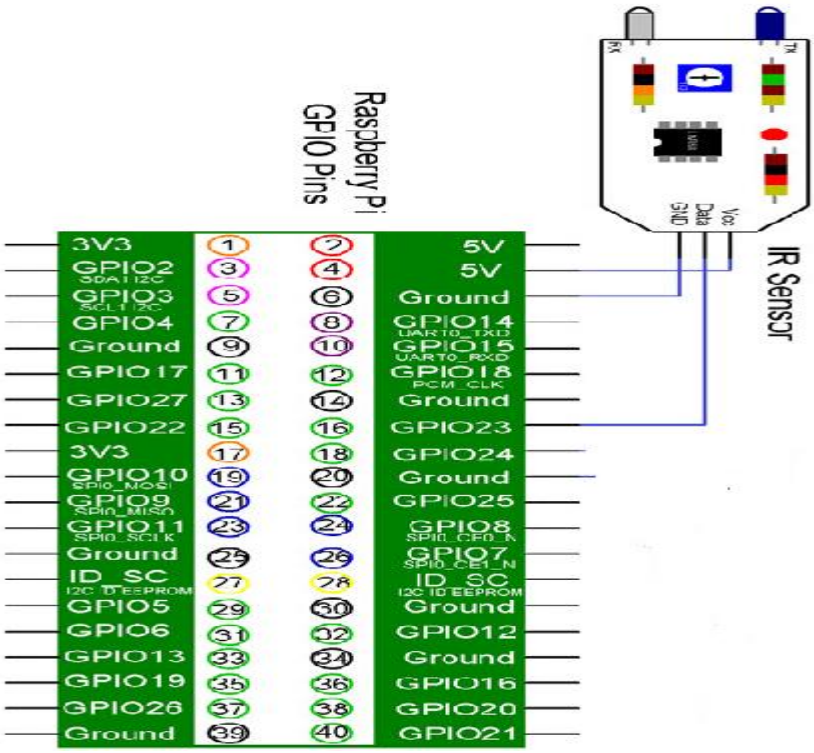
1. Import the required library: **import RPi.GPIO as GPIO**
2. Set GPIO mode and pin: **GPIO.setmode(GPIO.BCM)** and **GPIO.setup(21, GPIO.IN)**
3. Define a function to read sensor output
4. Call the **read_ir()** function to read sensor output: **read_ir()**
5. Save the Python file with a suitable name, for example, **ir_sensor.py**
6. Run the Python file using the command: **python3 ir_sensor.py**
7. The sensor output will be displayed on the terminal. If IR is detected, the message "IR Detected!" will be printed, otherwise, the message "No IR Detected." will be printed.

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: To interface IR sensor with Raspberry PI and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-18 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. What is an IR sensor, and how does it work? What are some common applications for IR sensors?
2. How would you physically connect an IR sensor to a Raspberry Pi, and what factors should you consider when choosing which pins to use?
3. What software libraries are required to interface an IR sensor with a Raspberry Pi, and how would you install and import these libraries in your Python code?

TITLE: To interface IR sensor with Raspberry PI and display the sensor output.

GPREC/DEEE/EXPT-AIOT (P)-18 **Date: 17/01/2023**

4. How does the Python code you wrote detect IR signals from the IR sensor, and what steps are involved in interpreting and responding to these signals?
5. What are some potential challenges or limitations of using an IR sensor with a Raspberry Pi ?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: To interface an Array of LEDs with Raspberry PI and

a) turn on the ALL LEDs once b) turn on LEDs one by one from left to right and then from right to leftGPREC/DEEE/EXPT-AIOT (P)-19 Date:

OBJECTIVE:

To interface an Array of LEDs with Raspberry PI and a) turn on the ALL LEDs once b) turn on LEDs one by one from left to right and then from right to left

APPARATUS:

1. Raspberry PI board
2. Array of LEDs
3. Jumper wires

ALGORITHM: a) turn on the ALL LEDs once

1. Import the required library **import RPi.GPIO as GPIO**
2. Set GPIO mode and pins **GPIO.setmode(GPIO.BCM)** and **GPIO.setup(17, GPIO.OUT)** to **GPIO.setup(24, GPIO.OUT)**
3. Set all GPIO pins to high to turn on all LEDs
4. Give a delay after which Set all GPIO pins to Low to turn OFF all LEDs

b) turn on LEDs one by one from left to right and then from

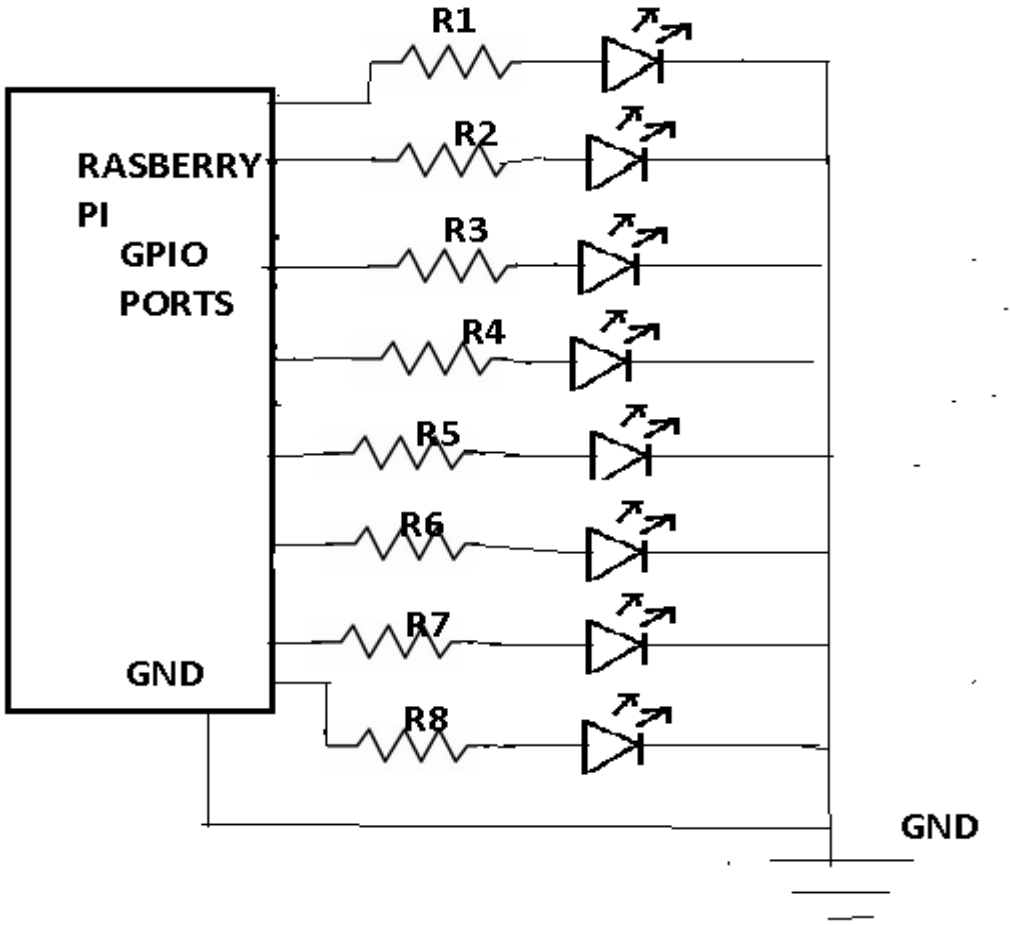
1. Import the required library: **import RPi.GPIO as GPIO**
2. Set GPIO mode and pins: **GPIO.setmode(GPIO.BCM)** and **GPIO.setup(17, GPIO.OUT)** to **GPIO.setup(24, GPIO.OUT)**
3. Write a loop to turn on LEDs from left to right
4. Write a loop to turn on LEDs from right to left
5. Save the Python file with a suitable name, for example, **led_array.py**
6. Run the Python file using the command: **python3 led_array.py**

TITLE: To interface an Array of LEDs with Raspberry PI and

a) turn on the ALL LEDs once b) turn on LEDs one by one from left to right and then from right to leftGPREC/DEEE/EXPT-AIOT (P)-19 Date:

7. LEDs connected to Raspberry Pi will turn on one by one from left to right and then turn off one by one from right to left. This process will continue until the Python script is terminated.

CONNECTION DIAGRAM:



TITLE: To interface an Array of LEDs with Raspberry PI and

a) turn on the ALL LEDs once b) turn on LEDs one by one from left to right and then from right to leftGPREC/DEEE/EXPT-AIOT (P)-19 Date:

RESULT:

QUESTIONS:

1. What is the purpose of using a resistor in conjunction with each LED?
2. How would you determine the appropriate resistor value to use for your specific LED and power supply setup?
3. What software libraries are required to interface an array of eight LEDs with a Raspberry Pi, and how would you install and import these libraries in your Python code?
4. How does the Python code you wrote control the individual LEDs in the array?
5. What are some potential sources of error or instability in an LED array system, and how could you troubleshoot or correct for these issues?

G. Pulla Reddy Engineering College (Autonomous): Kurnool
Department of Electrical & Electronics Engineering
B.Tech EEE – VII Semester (Scheme: 2020)
APPLICATIONS OF IOT LABORATORY (AIOT(P))

TITLE: To interface an Array of LEDs with Raspberry PI and turn on the LEDs to display a hexadecimal number equivalent value

GPREC/DEEE/EXPT-AIOT (P)-20 **Date: 17/01/2023**

OBJECTIVE:

To interface Ultrasonic sensor with NodeMCU and display the sensor output.

APPARATUS:

1. Raspberry PI board
2. Array of LED's
3. Jumper wires

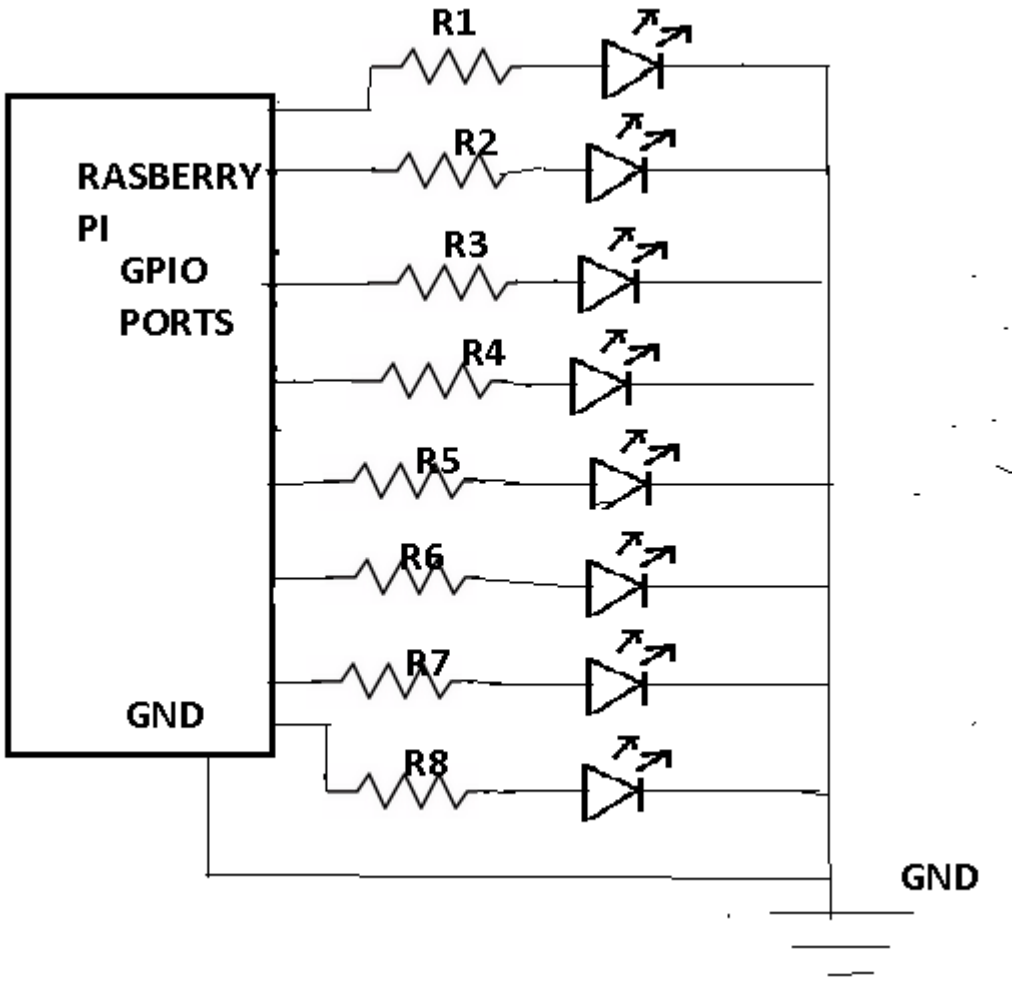
ALGORITHM:

1. Import the required library: **import RPi.GPIO as GPIO**
2. Set GPIO mode and pins: **GPIO.setmode(GPIO.BCM)** and **GPIO.setup(17, GPIO.OUT)** to **GPIO.setup(24, GPIO.OUT)**
3. Write a function to convert decimal to hexadecimal
4. Call the **dec_to_hex()** function to convert a decimal number to a two-digit hexadecimal number: **hex_num = dec_to_hex(decimal_num)**
5. Write a list of binary numbers equivalent to each hexadecimal number

TITLE: To interface an Array of LEDs with Raspberry PI and turn on the LEDs to display a hexadecimal number equivalent value

GPREC/DEEE/EXPT-AIOT (P)-20 **Date: 17/01/2023**

CONNECTION DIAGRAM:



RESULT:

QUESTIONS:

1. What is a Raspberry Pi, and what is it used for?
2. How do you power a Raspberry Pi, and what type of power supply do you need?

TITLE: To interface an Array of LEDs with Raspberry PI and turn on the LEDs to display a hexadecimal number equivalent value

GPREC/DEEE/EXPT-AIOT (P)-20 **Date: 17/01/2023**

3. What is the function of the GPIO pins on a Raspberry Pi, and how can you use them to interface with external components?
4. What is the command line interface, and how can you use it to interact with a Raspberry Pi?
5. What is the purpose of the GPIO pins on a Raspberry Pi, and how can you use these pins to interface with external hardware components such as sensors, motors, or LEDs?