LAB

REPORT

# IRE 212 : IoT Architecture and Technologies Sessional

|  |  |
| --- | --- |
| PREPARED BY  Mehrin Farzana  ID: 2101013  Session: 2021-2022  Date: 24/09/2024 | SUPERVISED BY  Suman Saha  Lecturer  Department of IRE, BDU |

|  |  |
| --- | --- |
| logo1 | BANGABANDHU SHEIKH MUJIBUR RAHMAN DIGITAL UNIVERSITY  (BDU) |
|  |  |

**List of Experiments**

1. Home Automation Using Arduino and Bluetooth Control
   1. Interfacing Arduino uno with PIR motion sensor
   2. Interfacing Arduino uno with LDR
   3. Interfacing Arduino uno with DHT22 Temperature sensor
   4. Interfacing Arduino uno with HC-05 Bluetooth Module

**Experiment No.:** 01

**Experiment Statement:** Home Automation Using Arduino and Bluetooth Control

Components and supplies:

* PIR Motion Sensor (generic)
* LDR (LIGHT DEPENDENT RESISTER)
* Temperature Sensor
* HC-05 Bluetooth Module
* LED (generic)
* Arduino UNO
* Jumper wires (generic)

**Interfacing Arduino uno with PIR motion sensor:**

Pin Definitions:

* led: The LED is connected to digital pin 9 of the Arduino, which will light up when motion is detected.
* sensor: The signal pin of the PIR motion sensor is connected to digital pin 5, which will read the motion sensor's output.

**Circuit:**

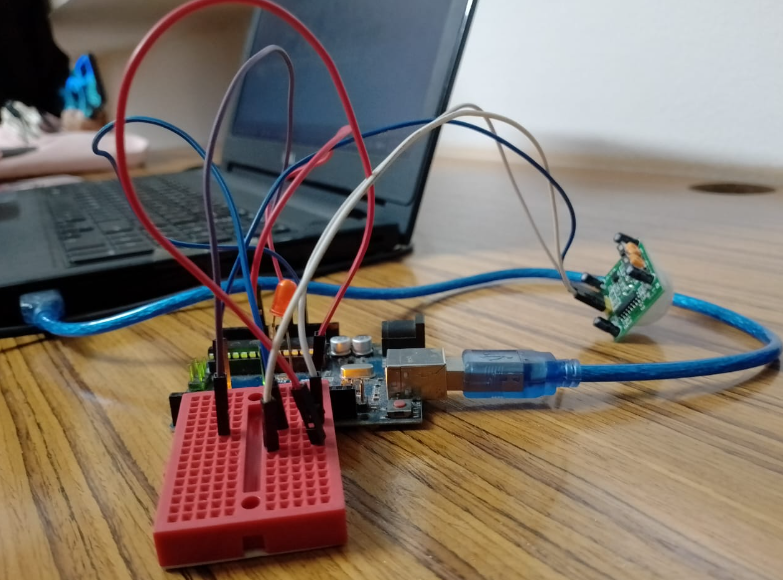
****

Figure 1: Circuit for Interfacing Arduino Uno with PIR motion sensor

**Code:**

const int led = 9;

const int sensor = 5; //signal pin of sensor to digital pin 5.

int state = LOW;

int val = 0;

void setup() { // Void setup is ran only once after each powerup or reset of the Arduino board.

  pinMode(led, OUTPUT); // Led is determined as an output here.

  pinMode(sensor, INPUT); // PIR motion sensor is determined is an input here.

  Serial.begin(9600);

}

void loop(){ // Void loop is ran over and over and consists of the main program.

  val = digitalRead(sensor);

  if (val == HIGH) {

    digitalWrite(led, HIGH);

    delay(500); // Delay of led is 500

    if (state == LOW) {

      Serial.println(" Motion detected");

      state = HIGH;

    }

  }

  else {

    digitalWrite(led, LOW);

    delay(500);

    if (state == HIGH){

      Serial.println("The action/ motion has stopped");

      state = LOW;

    }

  }

}

**Output:**

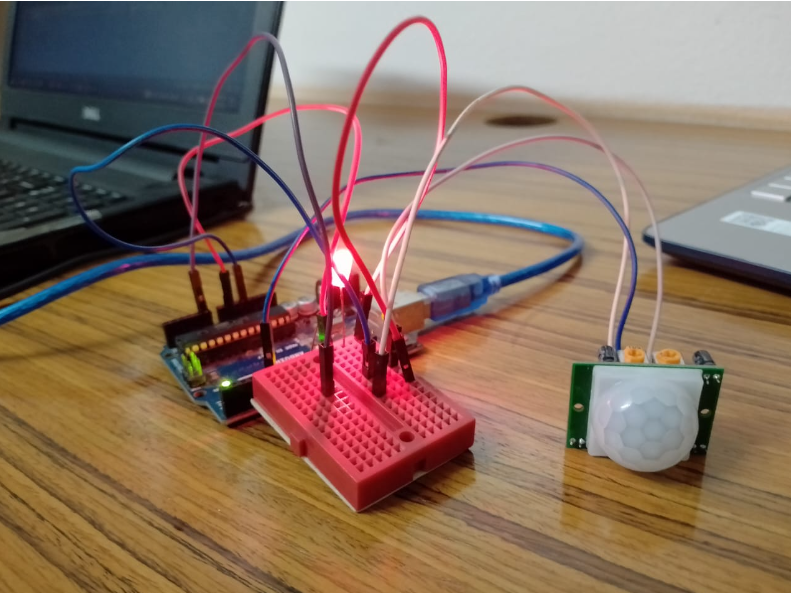
****

Figure 2: LED Turns On When Motion is Detected

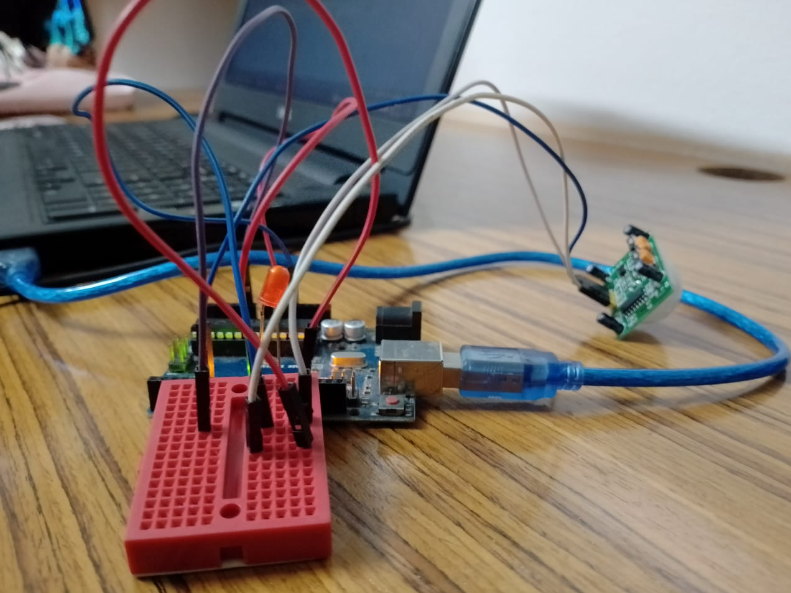
****

Figure 3: LED Turns Off When No Motion is Detected

**Interfacing Arduino uno with LDR :**

Pin Definitions:

* The LDR is connected to analog pin A0, which reads the varying light intensity values.
* An LED is connected to digital pin 9, which will be controlled based on the LDR reading.

**Circuit:**

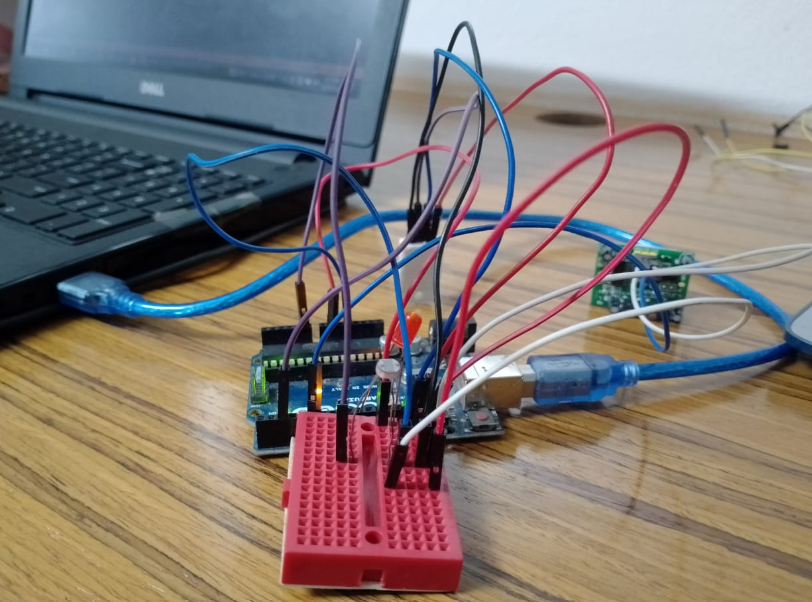
****

Figure 4: Circuit for Interfacing Arduino Uno with LDR

**Code:**

#define LDRpin A0

int LDRValue = 0;

int LedPin = 9;               // NEW

void setup()

{

  Serial.begin(9600);

  pinMode(9, OUTPUT);               // NEW

  digitalWrite(9, LOW);               // NEW

}

void loop()

{

  LDRValue = analogRead(LDRpin);

  Serial.println(LDRValue);

  delay(2000);

  if (LDRValue > 1000)      // NEW

  {

    digitalWrite(LedPin, LOW);

  }

  else

  {

    digitalWrite(LedPin, HIGH);

  }

}

**Output:**

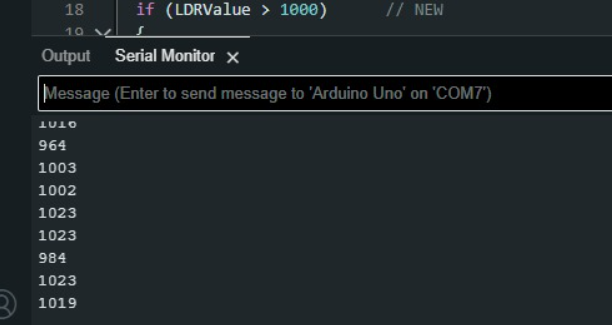
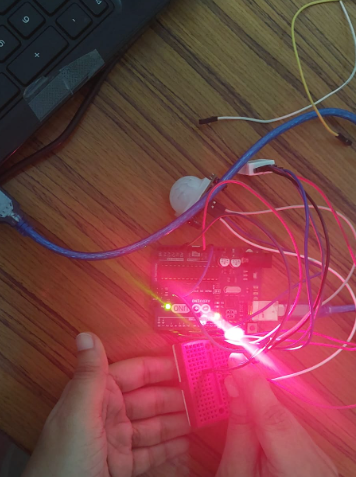
**** ****

Figure 5: LED turns on when the value is less than 1000

**Explanation of code according to output:**

* The LDR value is read from the analog pin and displayed on the serial monitor every 2 seconds.
* If the LDR value exceeds 1000 (bright light), the LED turns off. When the value is less than 1000 (low light), the LED turns on.

**Interfacing Arduino uno with DHT22 Temperature sensor:**

Connecting the DHT22 Sensor:

* The DHT22 sensor has three connections:
* VCC (Power): Connected to the 5V pin of the Arduino.
* GND (Ground): Connected to the ground pin.
* Signal (Data Output): Connected to digital pin 2 of the Arduino to read temperature and humidity data.

**Circuit:**

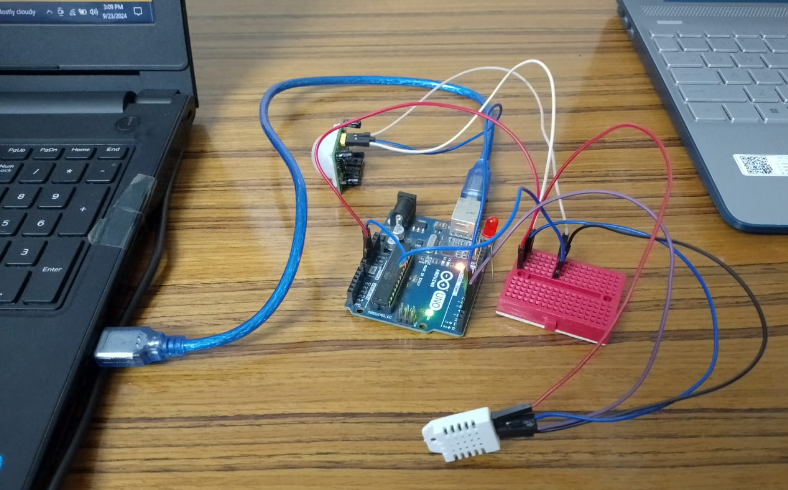
****

Figure 6: Circuit for Interfacing Arduino Uno with DHT22 Temperature sensor

**Code:**

#include <DHT.h>;

#define DHTPIN 2     // what pin we're connected to

#define DHTTYPE DHT22   // DHT 22  (AM2302)

DHT dht(DHTPIN, DHTTYPE); //// Initialize DHT sensor for normal 16mhz Arduino

int chk;

float hum;  //Stores humidity value

float temp; //Stores temperature value

void setup()

{

    Serial.begin(9600);

  dht.begin();

}

void loop()

{

    //Read data and store it to variables hum and temp

    hum = dht.readHumidity();

    temp= dht.readTemperature();

    //Print temp and humidity values to serial monitor

    Serial.print("Humidity: ");

    Serial.print(hum);

    Serial.print(" %, Temp: ");

    Serial.print(temp);

    Serial.println(" Celsius");

    delay(2000); //Delay 2 sec.

}

**Output:**

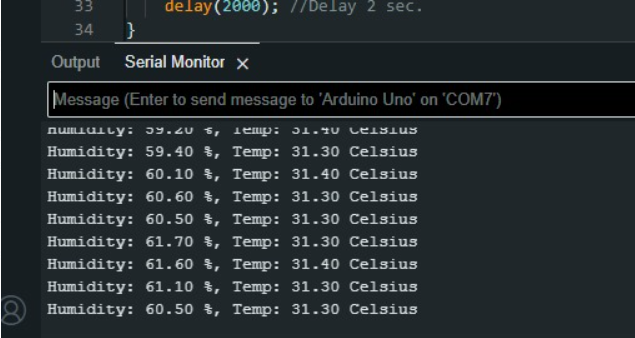
****

Figure 7: Real-time temperature and humidity monitoring with DHT22 temperature sensor

**Interfacing Arduino uno with HC-05 Bluetooth Module:**

Connecting the Bluetooth Module:

* VCC: Connected to the 5V pin of the Arduino.
* GND: Connected to the ground pin of the Arduino.
* RX (Receive): Connected to pin 1 (TX) of the Arduino.
* TX (Transmit): Connected to pin 0 (RX) of the Arduino.

**Circuit:**

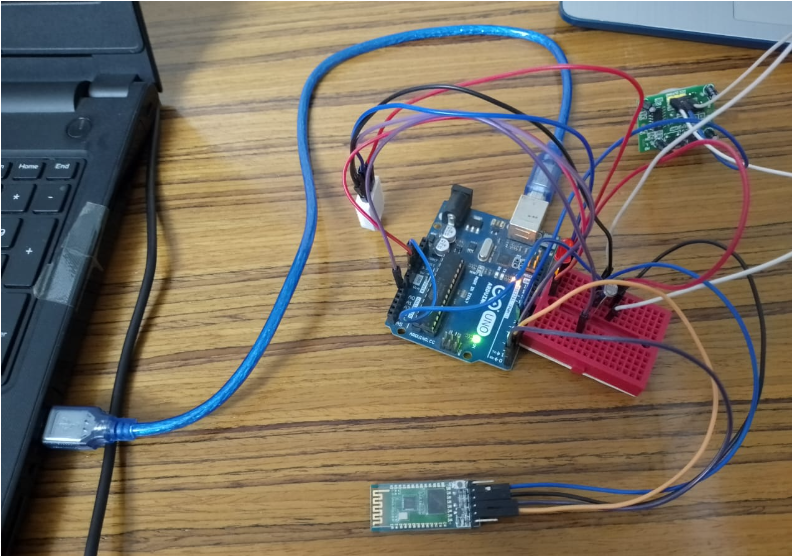
****

Figure 8: Circuit for Interfacing Arduino Uno with DHT22 Temperature sensor

**Code:**

#include <SoftwareSerial.h>

SoftwareSerial BTSerial(0, 1);  // RX, TX

void setup() {

  // Start communication with PC and notify the user

  Serial.begin(9600);

  Serial.println("Enter AT Commands");

  // Start communication with Bluetooth module

  BTSerial.begin(9600);

}

void loop() {

  // If data is received from Bluetooth, send it to the PC serial monitor

  if (BTSerial.available()) {

    while (BTSerial.available()) {

      Serial.write(BTSerial.read());

    }

  }

  // If data is received from PC serial monitor, send it to Bluetooth

  if (Serial.available()) {

    while (Serial.available()) {

      BTSerial.write(Serial.read());

    }

  }

**Output:**

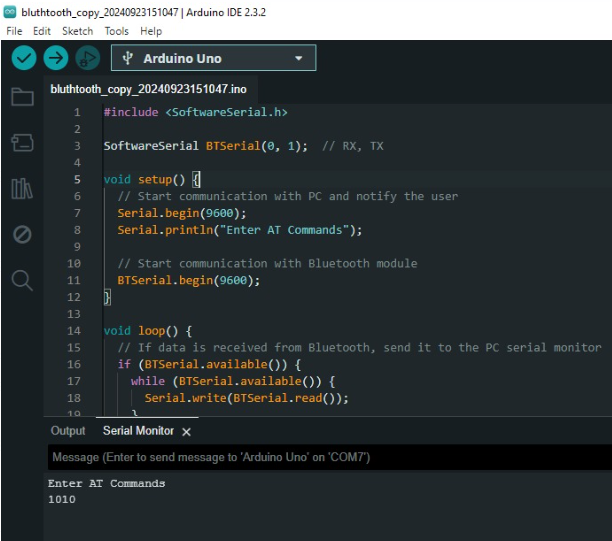
****

Figure 9: Output