|  |  |  |  |
| --- | --- | --- | --- |
| Ex. No.: 4 **Date:05/07/2022** | **Working with Environmental Sensors** | | |
| a. Write the Arduino sketch to demonstrate the use of Flame with LED or buzzer.  b. Write the Arduino sketch to demonstrate the use of Water level sensor with LED or buzzer.  c. Write the Arduino sketch to demonstrate the use of digital temperature sensor with LED or buzzer.  d. Write the Arduino sketch to demonstrate the use of Analog temperature sensor with LED or buzzer.  e. Write the Arduino sketch to demonstrate the use of Gas sensor.  f. Write the Arduino sketch to demonstrate the use of Soil and Moisture sensor. | | | |
| **Aim:**   1. To write an Arduino sketch to demonstrate the use of flame sensor with LED using Arduino and sensor. 2. To write an Arduino sketch to demonstrate the use of Water level sensor using Arduino and sensor. 3. To write an Arduino sketch to demonstrate the use of digital temperature using Arduino and sensor. 4. To write an Arduino sketch to demonstrate the use of analog temperature using Arduino and sensor. 5. To write an Arduino sketch to demonstrate the use of gas sensor LED using Arduino and sensor. 6. To write an Arduino sketch to demonstrate the use of soil and moisture sensor with LED using Arduino and sensor. | | | |
| **Apparatus required:**   1. Arduino UNO board, Arduino IDE, USB connector, connecting wires and flame sensor with LED. 2. Arduino UNO board, Arduino IDE, USB connector, connecting wires and Water level sensor. 3. Arduino UNO board, Arduino IDE, USB connector, connecting wires and digital temperature sensor. 4. Arduino UNO board, Arduino IDE, USB connector, connecting wires and analog temperature sensor. 5. Arduino UNO board, Arduino IDE, USB connector, connecting wires and gas/ smoke sensor. 6. Arduino UNO board, Arduino IDE, USB connector, connecting wires and soil and moisture sensor. | | | |
| **Circuit Connection**  **a)**    **b)**    **c)**    **d)**  **C:\Users\Elango\Desktop\iot\temp.png**  **e)**    **f)** | | | |
| **Arduino Code:**  **a)**  // lowest and highest sensor readings:  const int sensorMin = 0; // sensor minimum  const int sensorMax = 1024; // sensor maximum  void setup() {  // initialize serial communication @ 9600 baud:  Serial.begin(9600);  }  void loop() {  // read the sensor on analog A0:  int sensorReading = analogRead(A0);  // map the sensor range (four options):  // ex: 'long int map(long int, long int, long int, long int, long int)'  int range = map(sensorReading, sensorMin, sensorMax, 0, 3);    // range value:  switch (range) {  case 0: // A fire closer than 1.5 feet away.  Serial.println(sensorReading);  Serial.println("\*\* Close Fire \*\*");  break;  case 1: // A fire between 1-3 feet away.  Serial.println(sensorReading);  Serial.println("\*\* Distant Fire \*\*");  break;  case 2: // No fire detected.  Serial.println(sensorReading);  Serial.println("No Fire");  break;  }  delay(1000);  // delay between reads  }  **b)**  // Sensor pins  #define sensorPin A0  // Value for storing water level  int val = 0;  void setup() {  Serial.begin(9600);  }  void loop() {  //get the reading from the function below and print it  int level = readSensor();  Serial.print("Water level: ");  Serial.println(level);  delay(1000);  }  //This is a function used to get the reading  int readSensor() {  //digitalWrite(sensorPower, HIGH); // Turn the sensor ON  delay(10); // wait 10 milliseconds  val = analogRead(sensorPin); // Read the analog value form sensor  //digitalWrite(sensorPower, LOW); // Turn the sensor OFF  return val; // send current reading  }  **c)**  #include <OneWire.h>  #include <DallasTemperature.h>  // Data wire is plugged into pin 2 on the Arduino  #define ONE\_WIRE\_BUS 2  // Setup a oneWire instance to communicate with any OneWire devices  // (not just Maxim/Dallas temperature ICs)  OneWire oneWire(ONE\_WIRE\_BUS);  // Pass our oneWire reference to Dallas Temperature.  DallasTemperature sensors(&oneWire);  void setup(void)  {  // start serial port  Serial.begin(9600);  Serial.println("Dallas Temperature IC Control Library Demo");  // Start up the library  sensors.begin();  }  void loop(void)  {  Serial.print(" Requesting temperatures...");  sensors.requestTemperatures(); // Send the command to get temperature readings  Serial.println("DONE");  Serial.print("Temperature is: ");  Serial.print(sensors.getTempCByIndex(0)); // Why "byIndex"?  delay(1000);  }  **d)**  float temp;  int tempPin = 0;  void setup() {  Serial.begin(9600);  }  void loop() {  temp = analogRead(tempPin);  // read analog volt from sensor and save to variable temp  temp = temp \* 0.48828125;  // convert the analog volt to its temperature equivalent  Serial.print("TEMPERATURE = ");  Serial.print(temp); // display temperature value  Serial.print("\*C");  Serial.println();  delay(1000); // update sensor reading each one second  }  **e)**  #define MQ2pin (0)  float sensorValue; //variable to store sensor value  void setup()  {  Serial.begin(9600); // sets the serial port to 9600  Serial.println("Gas sensor warming up!");  delay(1000); // allow the MQ-6 to warm up  }  void loop()  {  sensorValue = analogRead(MQ2pin); // read analog input pin 0    Serial.print("Sensor Value: ");  Serial.print(sensorValue);    if(sensorValue > 100)  {  Serial.println(" | Smoke detected!");  }  else  {  Serial.println("not detected");  }    Serial.println("");  delay(1000); // wait 2s for next reading  }  **f)**  int sensorPin = A0;  int sensorValue;  int limit = 300;  void setup() {  Serial.begin(9600);  pinMode(13, OUTPUT);  }  void loop() {  sensorValue = analogRead(sensorPin);  Serial.println("Analog Value : ");  Serial.println(sensorValue);    if (sensorValue<limit) {  digitalWrite(13, HIGH);  }  else {  digitalWrite(13, LOW);  }    delay(1000);  } | | | |
| **Circuit Description:**  **a)**   |  |  | | --- | --- | |  |  | | **Flame Sensor Pins** | **ARDUINO PINS** | | Sensor’s VCC | 5V | | Sensor’s - | GND | | D0 | D2 | |  |  |   **b)**   |  |  | | --- | --- | | **Water Level Sensor Pins** | **ARDUINO PINS** | | Sensor’s + | 5V | | Sensor’s - | GND | | Sensor’s S | A0 |   **c)**   |  |  | | --- | --- | | **Digital Temperature Sensor Pins** | **ARDUINO PINS** | | Sensor’s Gnd | GND | | Sensor’s Vdd | 5V | | Sensor’s Data | D2 |     **d)**   |  |  | | --- | --- | | **Analog Temperature Sensor Pins** | **ARDUINO PINS** | | Sensor’s + | 5V | | Sensor’s S | A0 | | Sensor’s - | GND |   **e)**   |  |  | | --- | --- | | **Gas Sensor Pins** | **ARDUINO PINS** | | Sensor’s VCC | 5V | | Sensor’s GND | GND | | A0 | A0 |   **f)**   |  |  | | --- | --- | | **Soil Moisture Sensor Pins** | **ARDUINO PINS** | | Sensor’s + | 5V | | Sensor’s GND | GND | | AOUT | A0 | | | | |
| **Output:**  **a)**    **b)**    **c)**    **d)**    **e)**    **f)** | | | |
| **Result:**  Thus, the working with environmental sensors such as flame sensor, water level sensor, analog and digital temperature sensor, gas sensor and soil moisture sensor were implemented using Arduino UNO board and Arduino IDE and the respective sensors. | | | |
| **Remarks by the Course Instructor:** | | **Date of completion** |  |
| **Marks** | **out of 10** |
| **Course Instructor’s signature with date** |  |