Jai Talreja 59 D15A

WT LAB 2

Study of Various Types of Sensors and Display Devices

Aim :-

To study and demonstrate the interfacing of various sensors (e.g., DHT-11/22, HC-SR04, MFRC522, PIR Sensor) and display devices using Arduino/Raspberry Pi. Additionally, to select a mini-project topic based on the concepts learned.

Theory :-

Sensors are critical components in wireless and IoT systems, enabling the collection of real-world data for processing and analysis. This experiment focuses on understanding the working principles of commonly used sensors and their integration with microcontrollers like Arduino or Raspberry Pi. Display devices are used to visualize the data collected by the sensors.

List of Sensors and Display Devices

1. **DHT-11/22**: Temperature and humidity sensor.
2. **HC-SR04**: Ultrasonic sensor for distance measurement.
3. **MFRC522**: RFID module for contactless communication.
4. **PIR Sensor**: Passive Infrared sensor for motion detection.
5. **Display Devices**: LCD, OLED, or 7-segment displays for output visualization.

Experiment Setup

Components Required

- Arduino Uno/Raspberry Pi
- DHT-11/22 Sensor
- HC-SR04 Ultrasonic Sensor
- MFRC522 RFID Module

- PIR Sensor
- LCD/OLED Display
- Jumper Wires
- Breadboard
- Power Supply

Interfacing and Demonstration

1. DHT-11/22 Sensor (Temperature and Humidity)

Working Principle: Measures temperature and humidity using a capacitive humidity sensor and a thermistor.

Interfacing with Arduino:

- Connect VCC to 5V, GND to GND, and Data Pin to a digital pin (e.g., D2).
- Use the DHT library to read data.

Code Example:

```
#include <DHT.h>
#define DHTPIN 2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

void setup()
  { Serial.begin(9600);
  dht.begin();
}

void loop() {
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  Serial.print("Humidity: "); Serial.print(h);
  Serial.print(" %, Temperature: "); Serial.print(t); Serial.println(" °C");
  delay(2000);
}
```

2. HC-SR04 Ultrasonic Sensor (Distance Measurement)

Working Principle: Uses ultrasonic waves to measure distance by calculating the time taken for the wave to reflect back.
Interfacing with Arduino:

- Connect VCC to 5V, GND to GND, Trig to D9, and Echo to D10.

Code Example:

```
#define trigPin 9
#define echoPin 10

void setup()
 { Serial.begin(9600);
 pinMode(trigPin, OUTPUT);
 pinMode(echoPin, INPUT);
}

void loop() {
 long duration, distance;
 digitalWrite(trigPin, LOW); delayMicroseconds(2);
 digitalWrite(trigPin, HIGH); delayMicroseconds(10);
 digitalWrite(trigPin, LOW);
 duration = pulseIn(echoPin, HIGH);
 distance = (duration / 2) / 29.1;
 Serial.print("Distance: "); Serial.print(distance); Serial.println(" cm");
 delay(500);
}
```

3. MFRC522 RFID Module

Working Principle: Uses radio frequency to read and write data on RFID tags.
Interfacing with Arduino:

- Connect VCC to 3.3V, GND to GND, RST to D9, SDA to D10, MOSI to D11, MISO to D12, and SCK to D13.

Code Example:

```
#include     <SPI.h>  #i
include <MFRC522.h>
#define RST_PIN 9
#define SS_PIN 10
MFRC522 mfrc522(SS_PIN, RST_PIN);
```

```
void setup()
 { Serial.begin(9600);
 SPI.begin();
 mfrc522.PCD_Init();
 Serial.println("Scan RFID Tag...");
}

void loop() {
 if (mfrc522.PICC_IsNewCardPresent() && mfrc522.PICC_ReadCardSerial())
  { Serial.print("UID: ");
  for (byte i = 0; i < mfrc522.uid.size; i++)
   { Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : "
   "); Serial.print(mfrc522.uid.uidByte[i], HEX);
  }
  Serial.println();
  mfrc522.PICC_HaltA();
 }
}
```

4. PIR Sensor (Motion Detection)

Working Principle: Detects infrared radiation emitted by objects in its field of view.
Interfacing with Arduino:

- Connect VCC to 5V, GND to GND, and OUT to D2.

Code Example:

```
#define pirPin 2
void setup()
{ Serial.begin(9600)
;
 pinMode(pirPin, INPUT);
}

void loop() {
 int motion = digitalRead(pirPin);
 if (motion == HIGH) {
  Serial.println("Motion Detected!");
 } else {
  Serial.println("No Motion");
 }
 delay(1000);
}
```

5. Display Devices (LCD/OLED)

Interfacing with Arduino:

- Connect VCC to 5V, GND to GND, SDA to A4, and SCL to A5 for I2C communication.

Code Example:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h> LiquidCrystal_I2C
lcd(0x27, 16, 2);

void setup() { lcd.begin();
  lcd.backlight();
  lcd.print("Hello, World!");
}

void loop() { lcd.setCursor(0,
  1);
  lcd.print("Temp: 25 C"); delay(1000);
}
```

Conclusion :-

The various types of sensors, types of displays, their functioning, how they interface with Arduino, and example code to use them, was studied, and understood.