

Exercise 3 – Interfacing DHT11 with Arduino and Print Temperature and Humidity

Aim:

To interface DHT11 temperature sensor with Arduino UNO and write a program to print temperature and humidity.

Apparatus Required:

Sign Number	Name of the Equipment	Quantity
1	Arduino UNO	1
2	Computer with Arduino IDE	1
3	USB Cable	1
4	DHT11 Temperature sensor	1
5	Breadboard	1
6	Jumper Wires	As Required

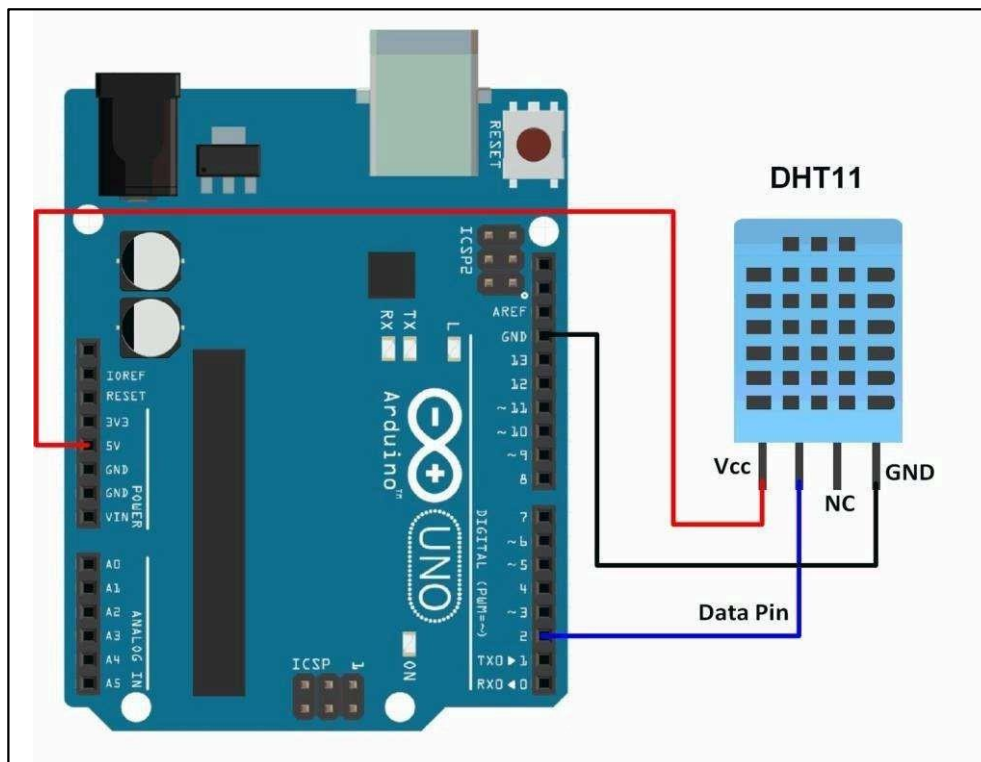
Theory:

DHT11 is a low-cost digital sensor for sensing temperature and humidity. This sensor can be easily interfaced with any micro-controller such as Arduino, Raspberry Pi etc. to measure humidity and temperature instantaneously. DHT11 humidity and temperature sensor is available as a sensor and as a module. The difference between this sensor and module is the pull-up resistor and a power-on LED.

DHT11 is a relative humidity sensor. To measure the surrounding air this sensor uses a thermistor and for measuring temperature this sensor uses a Negative Temperature coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers.

The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz. That is, it gives one reading for every second. DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5mA.

Circuit Diagram:



Code:**Procedure:**

1. Make connections as per the circuit diagram.
2. Open the Arduino IDE in your computer and write the above sketch.
3. Compile the sketch and upload it to Arduino UNO.
4. Once uploaded, open the Serial Monitor.
5. The temperature and humidity will be printed there.

```
#include <Adafruit_Sensor.h>
#include <DHT.h>
#include <DHT_U.h>

#define DHTPIN 7

#define DHTTYPE DHT11 // DHT 11

DHT_Unified dht(DHTPIN, DHTTYPE); // define dht class

void setup() {

  // serial monitor setup
  Serial.begin(9600);

  dht.begin(); // sensor starting up

  // sensor takes time to start up so wait
  Serial.println("Sensor Starting up....");
  delay(2000); // 2 seconds wait time

}

void loop() {

  sensors_event_t event;
  dht.temperature().getEvent(&event);

  if (isnan(event.temperature)) {
    Serial.println(F("Error reading temperature!"));
  }
  else {
    Serial.print(F("Temperature: "));
    Serial.print(event.temperature);
    Serial.println(F("°C"));
  }

  // Get humidity event and print its value.
  dht.humidity().getEvent(&event);
  if (isnan(event.relative_humidity)) {
```

Thus, the DHT11 temperature and humidity sensor is interfaced with Arduino UNO and the temperature and humidity data is printed successfully.

Exercise 3B – Interfacing Motor with Arduino Via Relay and Control It with PushButton

Aim: To interface motor with Arduino via relay module and turn it ON or OFF with a push button.

Apparatus Required:

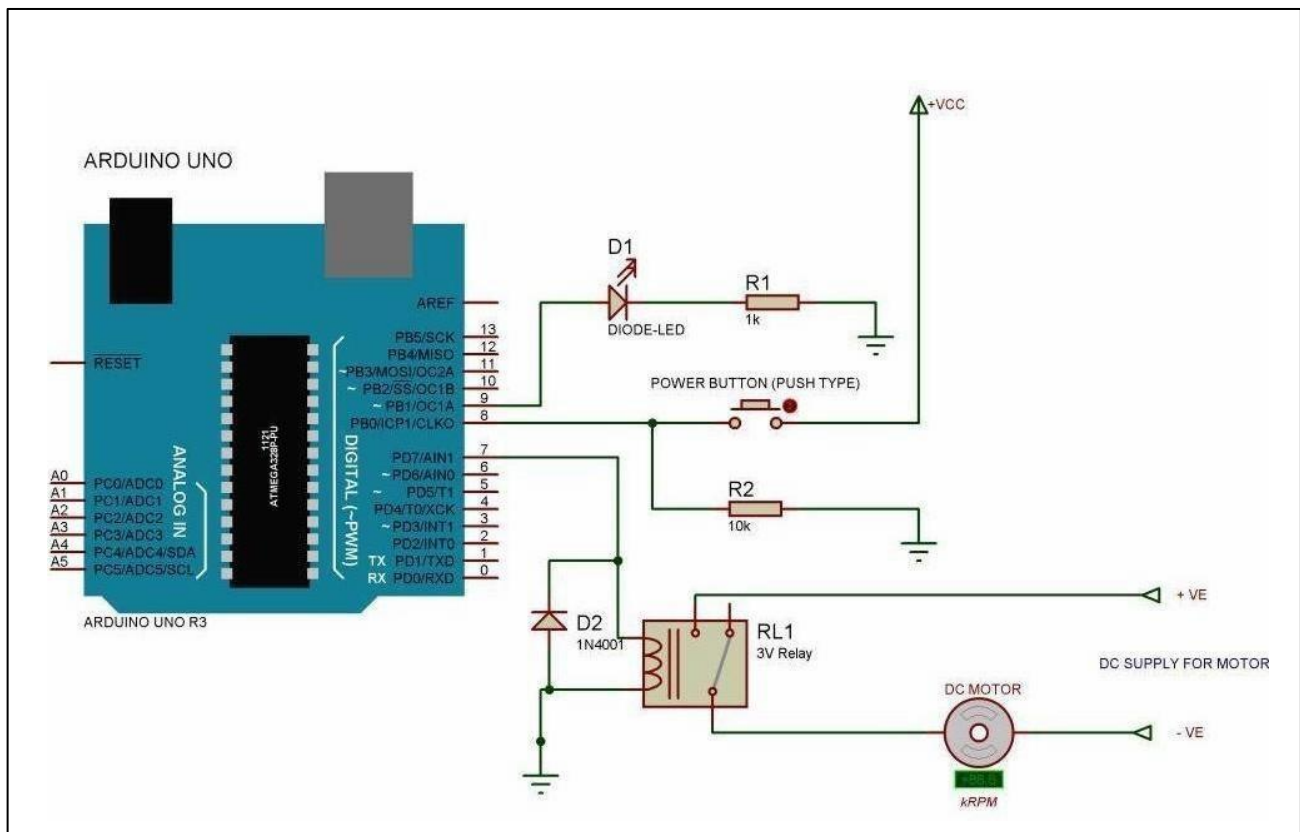
Sign Number	Name of the Equipment	Quantity
1	Arduino UNO	1
2	Computer with Arduino IDE	1
3	USB Cable	1
4	Relay Module	1
5	Push Button	1
6	Breadboard	1
7	Jumper Wires	As Required

Theory:

The pushbutton is a component that connects two points in a circuit when you press it. When the pushbutton is open (unpressed) there is no connection between the two legs of the pushbutton, so the pin is connected to 5 volts (through the pull-up resistor) and we read a HIGH. When the button is closed (pressed), it makes a connection between its two legs, connecting the pin to ground, so that we read a LOW. (The pin is still connected to 5 volts, but the resistor in-between them means that the pin is "closer" to ground.)

A relay is an electrically operated switch. It consists of a set of input terminals for a single or multiple control signals, and a set of operating contact terminals. The switch may have any number of contacts in multiple contact forms, such as make contacts, break contacts, or combinations thereof.

Circuit Diagram:



Code:

```
#define PUSH_BUTTON 8 // Use push button with pull-down resistor. #define RELAY
7

bool motorState = false; void

setup()
{
  pinMode(RELAY, OUTPUT);
  pinMode(PUSH_BUTTON, INPUT);
}

void loop()
{
  if(digitalRead(PUSH_BUTTON) == HIGH)
  {
    if(motorState == false) // If already OFF, turn it ON.
    {
      digitalWrite(RELAY, HIGH);
      motorState = true;
    }
    else
    {
      digitalWrite(RELAY, LOW); // Else, turn it OFF. motorState =
      false;
    }
    delay(100); // Delay to avoid debounce of push button.
  }
}
```

Procedure:

1. Make connections as per the circuit diagram.
2. Open the Arduino IDE in your computer and write the above sketch.
3. Compile the sketch and upload it to Arduino UNO.
4. Once uploaded, press the button. The relay turns ON and hence, the motor runs.
5. If the button is pressed again, the relay turns OFF and hence, the motor stops running.

Result:

Thus, motor is successfully interfaced with Arduino via relay module and is turned ON and OFF with push button.