# INTERNET OF THINGS LAB

# EXP-05

Name: Priyanshu Teotia

Reg.no: 22MIA1062

## INTERFACE DHT11 SENSOR WITH RASPBERRY PI AND ARDUINO

**DESCRIPTION:**

Both Arduino and Raspberry Pi platforms can connect to DHT11 Temperature and Humidity Sensors to monitor environmental measurements according to this project. This system obtains immediate temperature and humidity measurements which it transfers to Serial Monitor through Arduino along with remote system connectivity through Raspberry Pi network communications. The system deploys Serial Communication for Ardunio data transmission but uses TCP/IP-based data transmission for Raspberry Pi data monitoring. The project delivers essential learning about sensor coupling techniques together with microcontroller development and distributed IoT application capabilities.

**REQUIRED COMPONENTS:**

- Arduino Board
- Raspberry Pi
- DHT11 Temperature and Humidity Sensor
- 10kΩ Pull-up Resistor
- Jumper Wires
- Breadboard
- USB Cable / Power Adapter
- Wi-Fi or Ethernet (For Raspberry Pi Communication)

**HARDWARE SPECIFICATIONS:**

**1. Arduino Board:**

- The system requires 5V operating power that users can supply either through the USB connection or an external power source.
- Maximum Input Voltage: 12V (external supply)
- The system functions to obtain DHT11 sensor measurements that it distributes through the Serial Monitor interface for live observation.

**2. Raspberry Pi:**

- Users can power up the system with either a 5V micro-USB connection or a corresponding USB-C power adapter.

- GPIO Voltage Level: 3.3V
- This system collects DHT11 sensor input before forwarding it to remote network clients through TCP/IP.

**3. DHT11 Temperature & Humidity Sensor:**

- Operating Voltage: 3.3V - 5.5V
- Purpose: Captures temperature and humidity and transmits the readings in digital format.

**4. 10kΩ Resistor:**

- The pull-up resistor works as a signal stabilization method for DHT11 sensor transmission.

**5. Network Connectivity (Raspberry Pi Only):**

- Purpose: Enables data transmission over Wi-Fi or Ethernet for remote monitoring.

**HARDWARE DESCRIPTIONS:**

**1. Arduino Board:**

- The Arduino obtains temperature and humidity information from the DHT11 sensor which it displays on its built-in Serial Monitor through USB.
- Users can watch environmental data in real-time through this system irrespective of LCD display requirements.

**2. Raspberry Pi:**

- With Raspberry Pi users can utilize it as a sensor node allowing the board to collect readings from the DHT11 sensor before sending this information through TCP/IP networks to remote devices.
- Sensor data functions as input for three processing options including storage in a database as well as transmission to cloud services and web-based display platforms.

**3. DHT11 Sensor:**

- After measuring temperature and humidity data the DHT11 device converts these measurements into digital format which will be sent to an Arduino or Raspberry Pi implementation unit for further processing.

**4. 10kΩ Pull-up Resistor:**

- The VCC interface with the data pin through a pull-up resistor creates a system of reliable signal stability required to measure sensor values accurately.

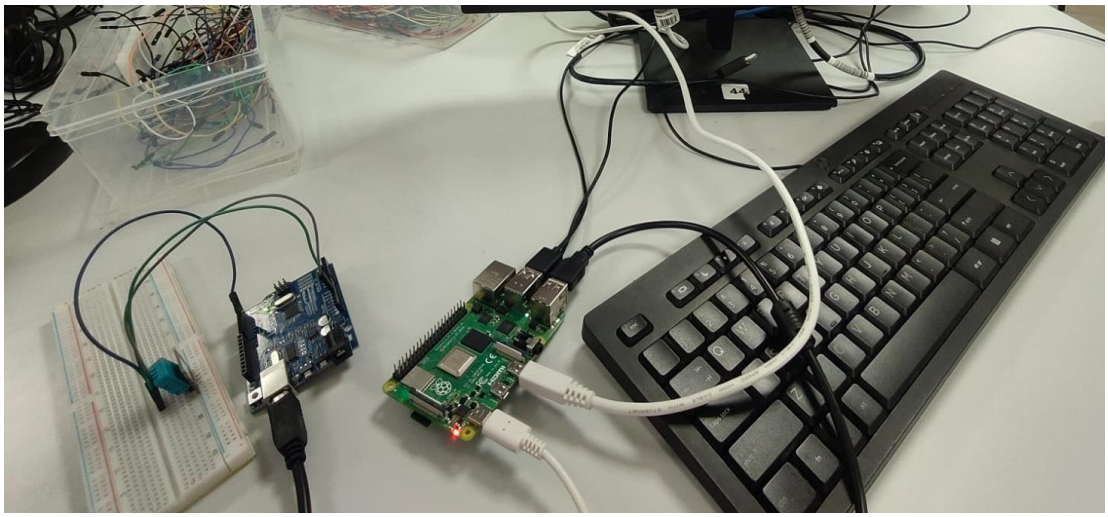
**CIRCUIT CONNECTIONS:**

**Arduino:**

| DHT11 Pin | Label | Purpose | Arduino Connection |
|---|---|---|---|
| 1 | VCC | Power supply (3.3V – 5V) | 5V |
| 2 | Data | Outputs digital humidity & temperature data | Digital Pin 2 |
| 3 | NC | Not connected | Leave unconnected |
| 4 | GND | Ground connection | GND |

**Raspberry Pi:**

| DHT11 Pin | Label | Purpose | Raspberry Pi Connection |
|---|---|---|---|
| 1 | VCC | Power supply (3.3V – 5V) | 3.3V |
| 2 | Data | Outputs digital humidity & temperature data | GPIO4 (Pin 7) |
| 3 | NC | Not connected | Leave unconnected |
| 4 | GND | Ground connection | GND (Pin 6) |



**CODE:**

```
// REQUIRES the following Arduino libraries:
// - DHT Sensor Library: https://github.com/adafruit/DHT-sensor-library
// - Adafruit Unified Sensor Lib: https://github.com/adafruit/Adafruit_Sensor

#include "DHT.h"

#define DHTPIN 2     // Digital pin connected to the DHT sensor
// Feather HUZZAH ESP8266 note: use pins 3, 4, 5, 12, 13 or 14 --
// Pin 15 can work but DHT must be disconnected during program upload.

// Uncomment whatever type you're using!
//#define DHTTYPE DHT11   // DHT 11
#define DHTTYPE DHT11   // DHT 22  (AM2302), AM2321
//#define DHTTYPE DHT21   // DHT 21 (AM2301)

// Connect pin 1 (on the left) of the sensor to +5V
// NOTE: If using a board with 3.3V logic like an Arduino Due connect pin 1
// to 3.3V instead of 5V!
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 3 (on the right) of the sensor to GROUND (if your sensor has 3 pins)
// Connect pin 4 (on the right) of the sensor to GROUND and leave the pin 3 EMPTY (if your sensor has 4 pins)
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor

// Initialize DHT sensor.
// Note that older versions of this library took an optional third parameter to
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));

  dht.begin();
}

void loop() {
  // Wait a few seconds between measurements.
  delay(2000);
```

**OUTPUT:**

```
DHTxx test!
Humidity: 64.00%  Temperature: 24.80°C 76.64°F  Heat index: 25.01°C 77.01°F
Humidity: 64.00%  Temperature: 24.80°C 76.64°F  Heat index: 25.01°C 77.01°F
Humidity: 64.00%  Temperature: 24.80°C 76.64°F  Heat index: 25.01°C 77.01°F
Humidity: 64.00%  Temperature: 24.80°C 76.64°F  Heat index: 25.01°C 77.01°F
Humidity: 64.00%  Temperature: 24.80°C 76.64°F  Heat index: 25.01°C 77.01°F
Humidity: 64.00%  Temperature: 24.80°C 76.64°F  Heat index: 25.01°C 77.01°F
```

Autoscroll  Show timestamp  Newline  9600 baud  Clear

```
#define DHTTYPE DHT11   // DHT 22  (AM2302), AM2321
//#define DHTTYPE DHT21   // DHT 21 (AM2301)

// Connect pin 1 (on the left) of the sensor to +5V
// NOTE: If using a board with 3.3V logic like an Arduino Due connect pin 1
// to 3.3V instead of 5V!
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 3 (on the right) of the sensor to GROUND (if your sensor has 3 pins)
// Connect pin 4 (on the right) of the sensor to GROUND and leave the pin 3 EMPTY (if your sensor has 4 pins)
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor

// Initialize DHT sensor.
// Note that older versions of this library took an optional third parameter to
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));
```

**RESULT:**

Thus the Expected Output Achieved Successful.