		
Laboratory 2: Integration of Sensors with ESP32	School of Applied Digital Technology	
Name: Name:	ID: ID:	Section: Section:
Date:	Due date:	

Objectives:

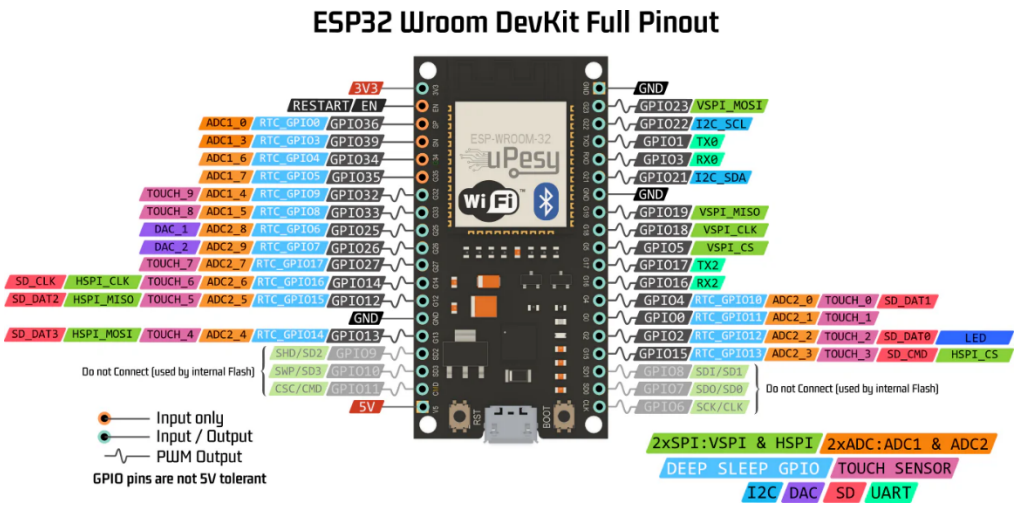
- Learn how to wire the DHT11 and MQ-2 sensors to the ESP32 board.
- Understand the basic setup and configuration using ESPHome.
- Test and view sensor data through Home Assistant.

Experiment 1: Integration of Sensors with ESP32

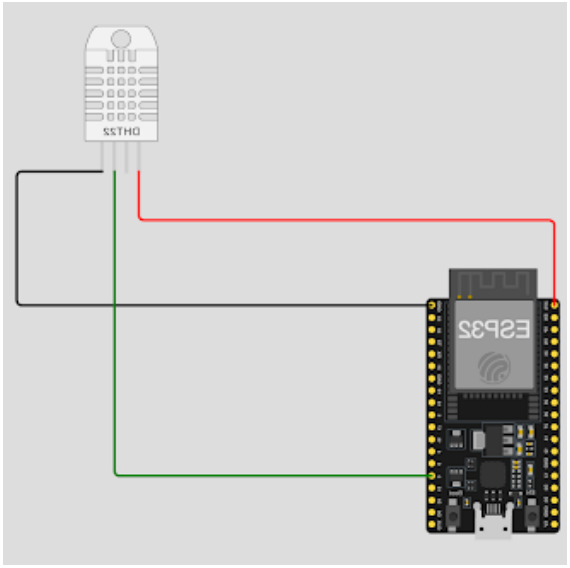
Equipment:

	Quantity
ESP32	1
Fan Motor	1
DHT11 Sensor	1

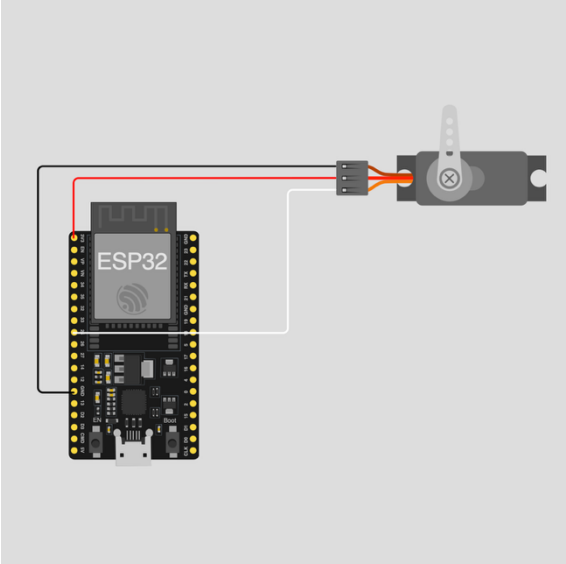
1. Wiring the Sensors to the ESP Board



- DHT11 (Temperature and Humidity Sensor)
    - VCC ⟶ 3.3V
    - GND ⟶ GND
    - DATA ⟶ GPIO22



- Fan Motor
    - VCC ⟶ 3.3V
    - GND ⟶ GND
    - DATA ⟶ GPIO25



2. Open ESPHome and Select the Device

- Go to the ESPHome Dashboard
- Select your device (e.g., sensor)
- Click EDIT to modify the YAML configuration

3. Add Code for the Sensors in the YAML File

Add the following to your configuration file:

- DHT11 (Temperature and Humidity Sensor)

```
sensor:
  - platform: dht
    pin: 22
    model: DHT11
    temperature:
      name: "Temperature"
```

```
    id: temperature
    unit_of_measurement: "°C"
  humidity:
    name: "Humidity"
    id: humidity
    unit_of_measurement: "%"
  update_interval: 10s
```

- **Fan Motor**

```
output:
  - platform: ledc
    pin: 25
    id: fan_pwm
    frequency: 25000 Hz
    inverted: true

fan:
  - platform: speed
    name: "DC Fan"
    output: fan_pwm
    speed_count: 100
    restore_mode: ALWAYS_ON
```

4. Save and Install the Firmware

- Click SAVE
- Click INSTALL
- Choose Wirelessly or USB depending on your setup
- Wait for the upload to complete and let the device reboot

5. View Sensor Data in Home Assistant

- Go to your Home Assistant Overview Dashboard
- You should now see sensor readings such as:

- Temperature / Humidity from DHT11
- Gas Detection Level from MQ-2

Take 1: Design and Build an Automated Fan Control System Using Temperature and Humidity Sensors.

Requirements:

1. Use an ESP32/ESP8266 board and a DHT11 sensor.
2. Connect a DC fan controlled via PWM (LEDC) to allow variable fan speed.
3. Read temperature and humidity every 10 seconds.
4. Combine the sensor reading and fan control code into a single ESPHome YAML file.
5. mplement automation so the fan operates according to the following conditions:
    - Temperature ≥ 30°C ⟶ Fan runs at 100% (full speed)
    - Temperature < 27°C ⟶ Fan turns OFF
6. Display temperature and humidity values in Home Assistant.

Home Assistant Dashboard:

Code:

----- Have a good day -----

Answer:

```yaml
sensor:
  - platform: dht
    pin: 22
    model: DHT11
    temperature:
      name: "Room Temperature"
      id: temperature
      unit_of_measurement: "°C"
    humidity:
      name: "Room Humidity"
      id: humidity
      unit_of_measurement: "%"
    update_interval: 10s

output:
  - platform: ledc
    pin: 25
    id: fan_pwm
    frequency: 25000 Hz
    inverted: true

fan:
  - platform: speed
    name: "DC Fan"
    id: dc_fan
    output: fan_pwm
    speed_count: 100
    restore_mode: ALWAYS_OFF

interval:
  - interval: 10s
    then:
      - lambda: |-
          if (id(temperature).has_state()) {
            float t = id(temperature).state;

            if (t >= 30.0) {
              id(dc_fan).turn_on();         // เปิดพัดลม
              id(dc_fan).speed = 1.0;       // ตั้งความเร็ว 100%
            } else if (t < 27.0) {
              id(dc_fan).turn_off();        // ปิดพัดลม
            }
          }
```