

1:Arduino Installation and Blink LED using NodeMCU ESP8266

## Correct Steps to Blink LED in Wokwi (NodeMCU ESP8266)

### 1. Create Project

1. Go to <https://wokwi.com>
  2. Click **Create New Project**
  3. Select **NodeMCU ESP8266**
- 

### 2. Add LED

4. Click “+ Add Part”
  5. Search **LED**
  6. Add **LED**
- 

### 3. Connect LED

7. LED **Anode (+)** → **D4**
8. LED **Cathode (-)** → **GND**

For NodeMCU:

- **D3 = GPIO0**

```
#define LED_PIN 0 // GPIO0 (D3)
```

```
void setup() {  
  pinMode(LED_PIN, OUTPUT);  
}
```

```
void loop() {  
  digitalWrite(LED_PIN, HIGH); // LED ON  
  delay(1000);  
  digitalWrite(LED_PIN, LOW); // LED OFF
```

```
delay(1000);  
}
```

## 5. Run Simulation

9. Click **Start Simulation**
10. LED will **blink every 1 second** 

Ex2 Interfacing IR (Infrared Proximity) Sensor with NodeMCU ESP8266

## Steps to Interface IR Receiver (DTT) with ESP32

### 1. Create Project

1. Go to <https://wokwi.com>
  2. Click **Create New Project** → **ESP32**
- 

### 2. Add Parts

1. Click + **Add Part** → **IR Receiver**
  2. Click + **Add Part** → **LED**
- 

### 3. Connections

| Part            | Pin         | ESP32 Pin                  |
|-----------------|-------------|----------------------------|
| IR Receiver DTT |             | GPIO14 ( <i>safe pin</i> ) |
| IR Receiver VCC |             | 3V3                        |
| IR Receiver GND |             | GND                        |
| LED             | Anode (+)   | GPIO2                      |
| LED             | Cathode (–) | GND                        |

### Code Example

```
#define IR_PIN 14 // Connect DTT here  
  
#define LED_PIN 2 // LED
```

```

void setup() {
    pinMode(IR_PIN, INPUT);
    pinMode(LED_PIN, OUTPUT);
}

void loop() {
    int value = digitalRead(IR_PIN);

    if (value == LOW) {    // IR signal detected
        digitalWrite(LED_PIN, HIGH);
    } else {
        digitalWrite(LED_PIN, LOW);
    }
}

```

Next:

```

#define LED_PIN 2  // GPIO2

void setup() {
    pinMode(LED_PIN, OUTPUT);
}

void loop() {
    digitalWrite(LED_PIN, HIGH); // LED ON
    delay(500);                  // 0.5 second
    digitalWrite(LED_PIN, LOW);  // LED OFF
    delay(500);                  // 0.5 second
}

```

```
}
```

## Ex3Interfacing Light Sensor with Node MCU Esp8266

### Step 1: Correct Connections

#### Component ESP32 Pin

LDR VCC 3.3V

LDR GND GND

LDR AO GPIO34

LED + GPIO2

LED – GND

#### Code

```
#define LDR_PIN 34
```

```
#define LED_PIN 2
```

```
void setup() {
```

```
    pinMode(LED_PIN, OUTPUT);
```

```
    Serial.begin(115200);
```

```
}
```

```
void loop() {
```

```
    int lightValue = analogRead(LDR_PIN);
```

```
    Serial.println(lightValue);
```

```
    if(lightValue > 1500){        // Adjust threshold based on Serial Monitor
```

```
        // Blink LED while it is dark
```

```
        digitalWrite(LED_PIN, HIGH);
```

```
    delay(500);  
    digitalWrite(LED_PIN, LOW);  
    delay(500);  
  } else {  
    digitalWrite(LED_PIN, LOW); // Turn off LED in bright light  
  }  
}
```

Ex4Interfacing Temperature & Humidity Sensor (DHT11/DHT22) with NodeMCU  
ESP8266

### Step 1: Open Wokwi

1. Go to <https://wokwi.com>
  2. Click “Create New Project” → ESP32
- 

### Step 2: Add DHT22 Sensor

1. Click + **Add Part**
2. Search for **DHT22**
3. Add it to the project

Wokwi allows you to simulate DHT22 without physical hardware.

---

### Step 3: Connect DHT22 to ESP32

#### DHT22 Pin ESP32 Pin

|      |       |
|------|-------|
| VCC  | 3.3V  |
| GND  | GND   |
| DATA | GPIO4 |

Code

```
#include "DHT.h"
```

```
#define DHTPIN 4
#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();
}

void loop() {
  float temperature = dht.readTemperature();
  float humidity = dht.readHumidity();

  if (isnan(temperature) || isnan(humidity)) {
    Serial.println("Failed to read DHT22 sensor!");
  } else {
    Serial.print("Temperature: "); Serial.print(temperature); Serial.print(" °C\t");
    Serial.print("Humidity: "); Serial.print(humidity); Serial.println(" %");
  }

  delay(2000);
}
```

Ex5Soil Moisture Sensor Interfacing with NodeMCU ESP8266

## **Step 2: Connections**

| <b>Component</b>                 | <b>ESP32 Pin</b>            |
|----------------------------------|-----------------------------|
| Potentiometer VCC                | 3.3V                        |
| Potentiometer GND                | GND                         |
| Potentiometer Wiper (middle pin) | GPIO34 (ADC)                |
| LED Anode (+)                    | GPIO2                       |
| LED Cathode (-)                  | GND (through 220Ω resistor) |

**Code:**

```
#define SOIL_AO 34 // Analog input pin
#define LED_PIN 2 // LED to indicate dry soil

void setup() {
  Serial.begin(115200);
  pinMode(LED_PIN, OUTPUT);
}

void loop() {
  int soilValue = analogRead(SOIL_AO); // 0-4095
  Serial.print("Soil Moisture Value: ");
  Serial.println(soilValue);

  // Dry soil if analog value is low
  if(soilValue < 1500){
    digitalWrite(LED_PIN, HIGH); // LED ON → dry soil
  } else {
    digitalWrite(LED_PIN, LOW); // LED OFF → wet soil
  }
}
```

```
    delay(500); // 0.5 second delay  
}
```