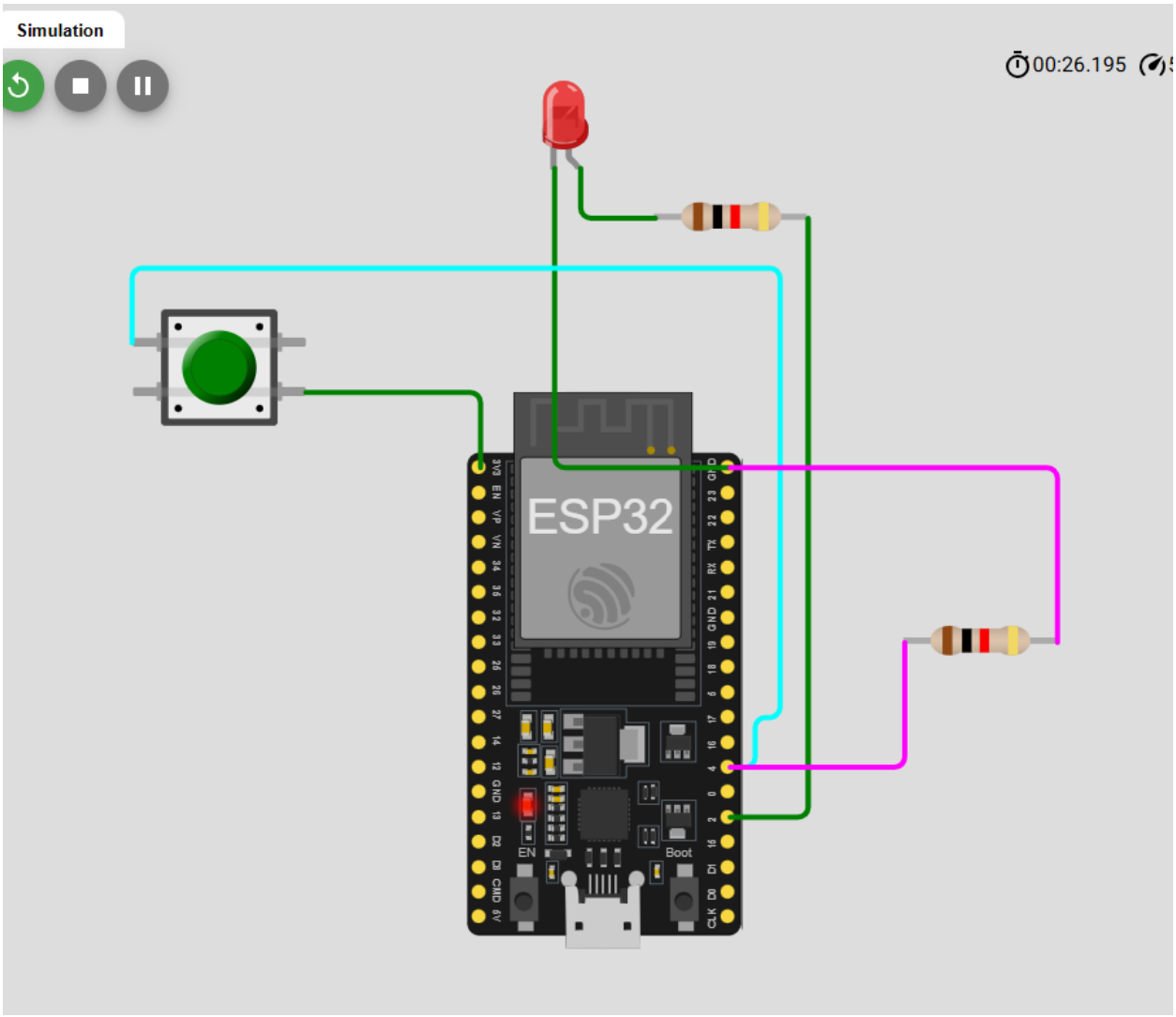
Esp32 pull-down configuration for Led blink:-



Current Wiring :-

- **Button side 1 → 3.3V**
- **Button side 2 → GPIO 4**
- **GPIO 4 → 10kΩ resistor → GND**

This is a **pull-down configuration**, which works perfectly:

1. When the button is **not pressed**:
    - GPIO 4 is connected to GND through the resistor → reads **LOW**
2. When the button **is pressed**:
    - GPIO 4 connects directly to 3.3V → reads **HIGH**

Code for it simple **Arduino program** that turns an LED on or off depending on a button press:-

```
#define LED_PIN 2
#define BUTTON_PIN 4

void setup() {
  pinMode(LED_PIN, OUTPUT);
  pinMode(BUTTON_PIN, INPUT); // pull-down external resistor
}

void loop() {
  int buttonState = digitalRead(BUTTON_PIN);

  if (buttonState == HIGH) {  // button pressed
    digitalWrite(LED_PIN, HIGH); // LED ON
  } else {
    digitalWrite(LED_PIN, LOW);  // LED OFF
  }
}
```

https://wokwi.com/projects/458819166707402753

Let's go **step by step** on the **optional improvement** using the ESP32's **internal pull-down resistor**. This simplifies your wiring and avoids needing a physical resistor.

---

1. Code Change

In your current `setup()`, you have:

```
pinMode(BUTTON_PIN, INPUT); // using external pull-down resistor
```

Change it to use the internal pull-down:

```
pinMode(BUTTON_PIN, INPUT_PULLDOWN); // use ESP32 internal pull-down
```

 Explanation:

- `INPUT_PULLDOWN` tells the ESP32 to automatically connect a weak resistor between the pin and GND internally.
- This means the pin **reads LOW** when the button is **not pressed**.
- When you press the button, it connects the pin to **3.3V**, which reads **HIGH**.
- No need for a 10kΩ resistor on the breadboard.

---

2. Wiring with Internal Pull-down

Now your pushbutton connection becomes simpler:

Button Pin	Connect To
Side 1	**3.3V**
Side 2	**GPIO 4**

That's it—**no external resistor needed**.

□ 3. How it Works

- **Button NOT pressed:**
  GPIO 4 is pulled down internally → reads **LOW** → LED OFF
- **Button pressed:**
  GPIO 4 connects to 3.3V → reads **HIGH** → LED ON

This is exactly the behavior you want.

□ 4. Updated Full Code

```
#define LED_PIN 2
#define BUTTON_PIN 4

void setup() {
  pinMode(LED_PIN, OUTPUT);
  pinMode(BUTTON_PIN, INPUT_PULLDOWN); // use internal pull-down resistor
}

void loop() {
  int buttonState = digitalRead(BUTTON_PIN);

  if (buttonState == HIGH) {  // button pressed
    digitalWrite(LED_PIN, HIGH); // LED ON
  } else {
    digitalWrite(LED_PIN, LOW);  // LED OFF
  }
}
```

□ 5. Benefits of Internal Pull-down

1. Fewer components → simpler breadboard
2. Cleaner wiring → less chance of mistakes
3. Saves a resistor for other circuits