

Module 1iv

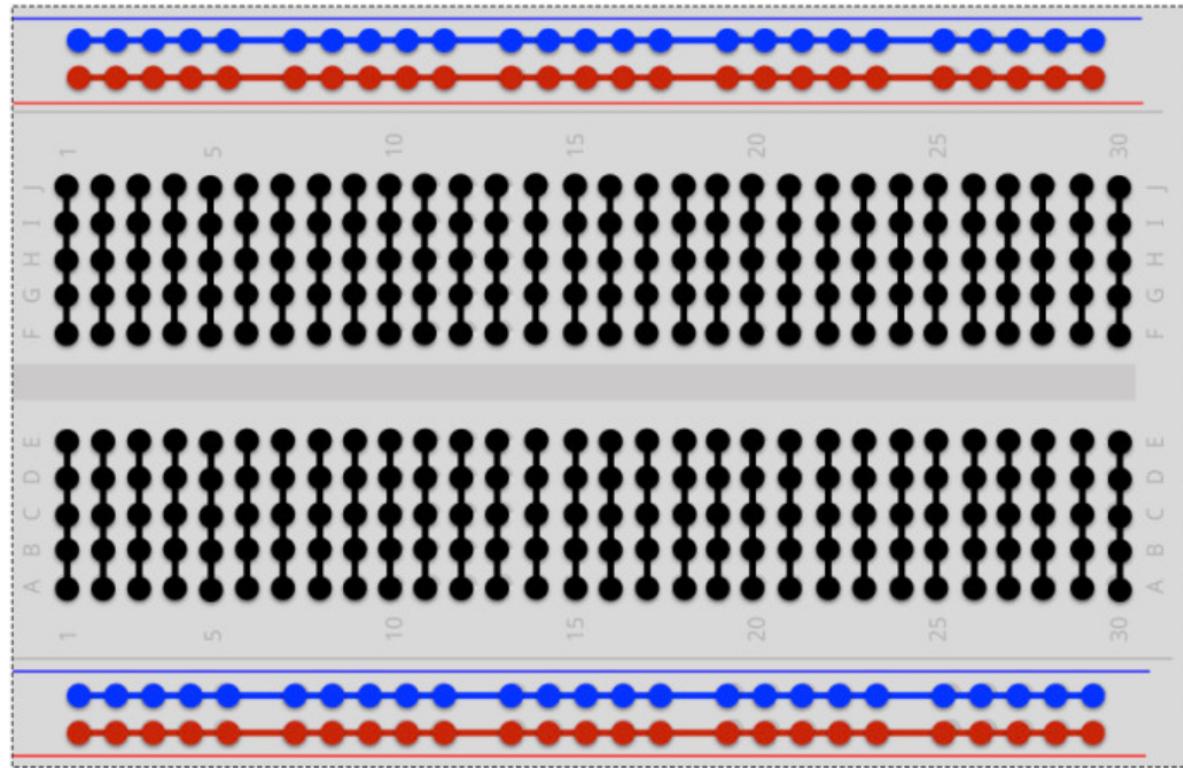
The Fundamental

safyzan salim
019 622 0575

1iv. The Fundamental

Breadboard Tips

- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises



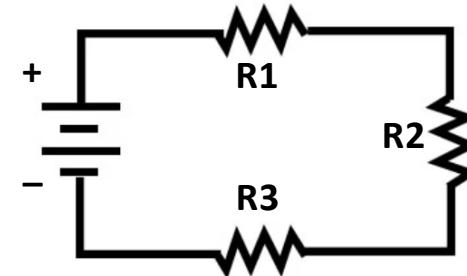
fritzing

1iv. The Fundamental

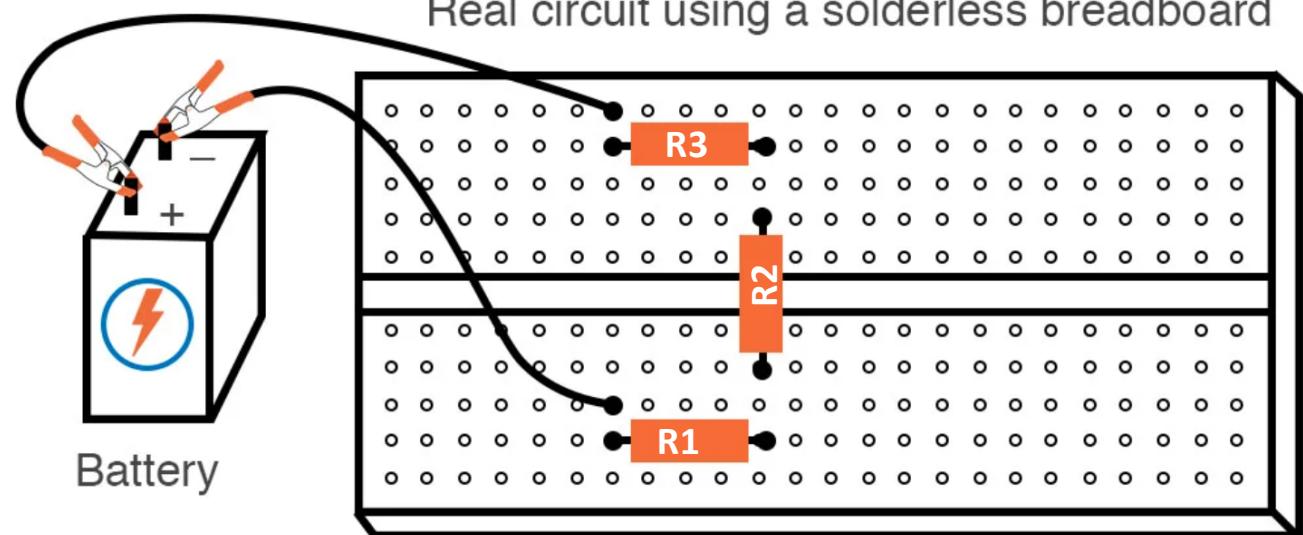
- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises

Method 1

Schematic diagram



Real circuit using a solderless breadboard



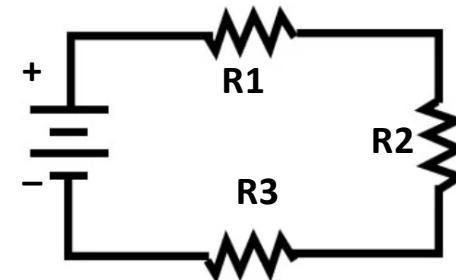
<https://www.allaboutcircuits.com/textbook/direct-current/chpt-5/building-simple-resistor-circuits/>

1iv. The Fundamental

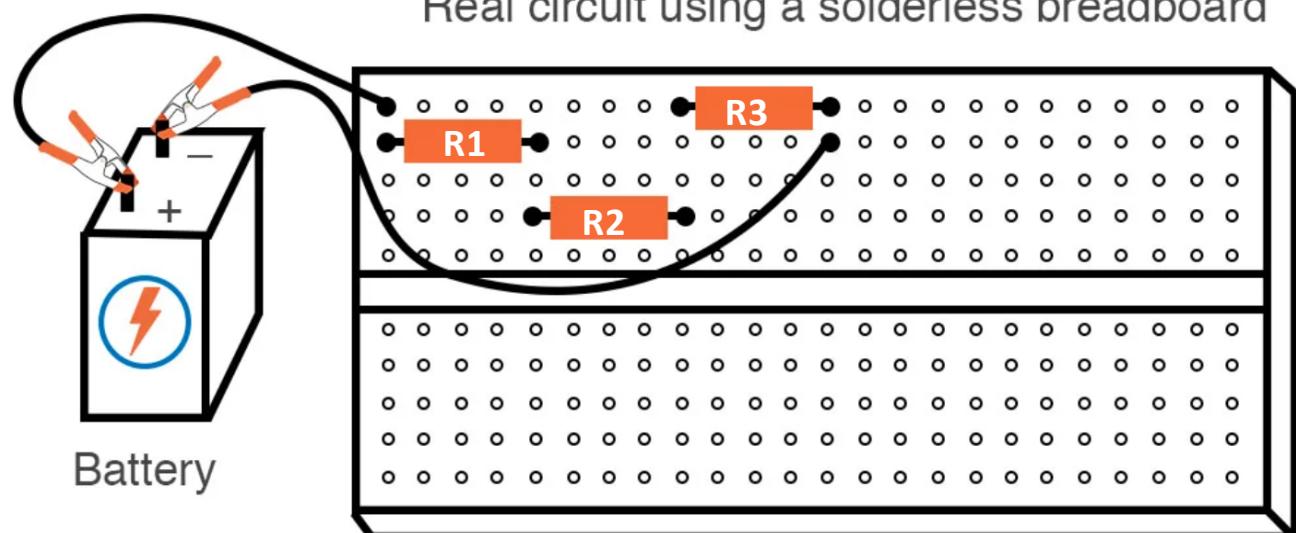
- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises

Method 2

Schematic
diagram



Real circuit using a solderless breadboard

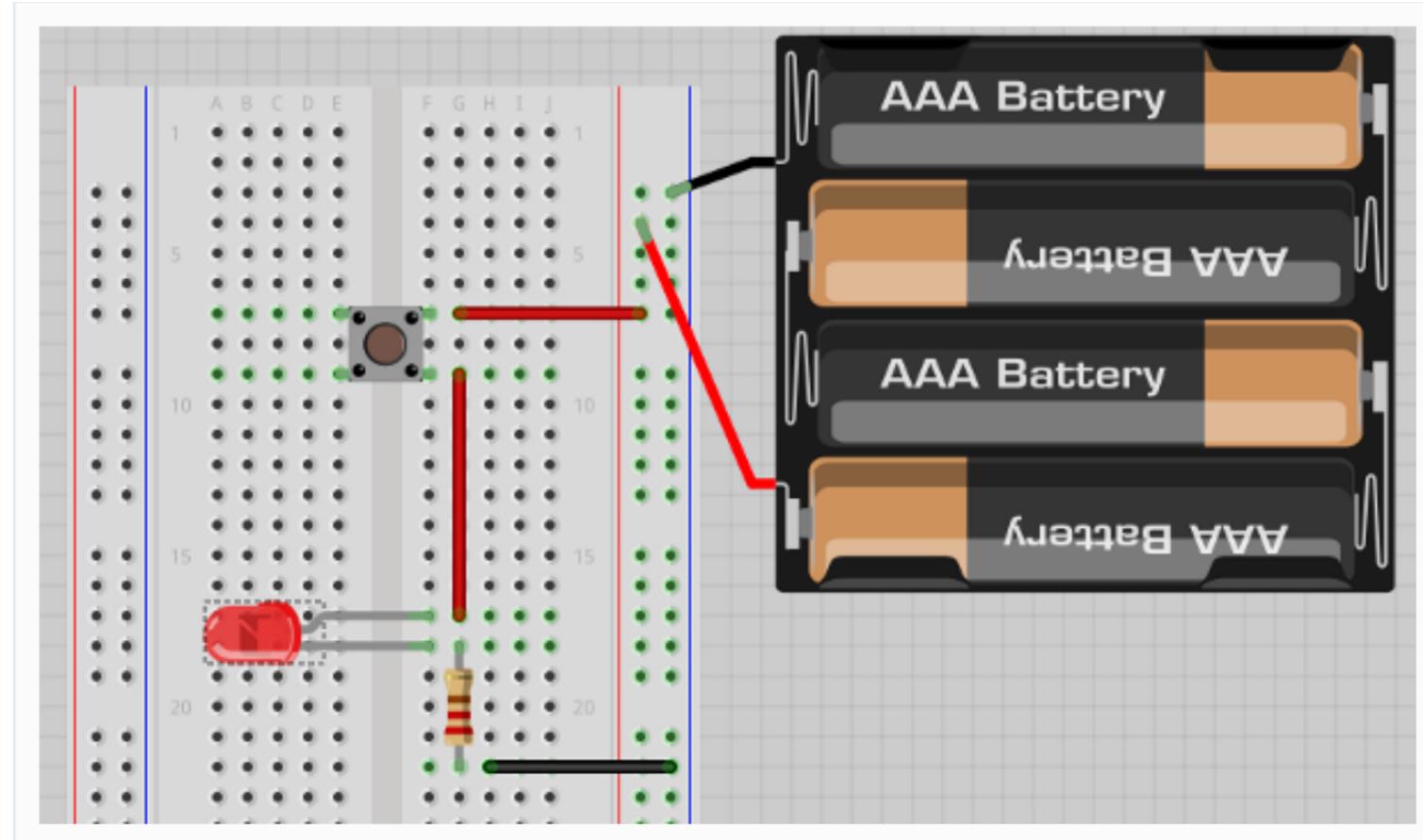


<https://www.allaboutcircuits.com/textbook/direct-current/chpt-5/building-simple-resistor-circuits/>

1iv. The Fundamental

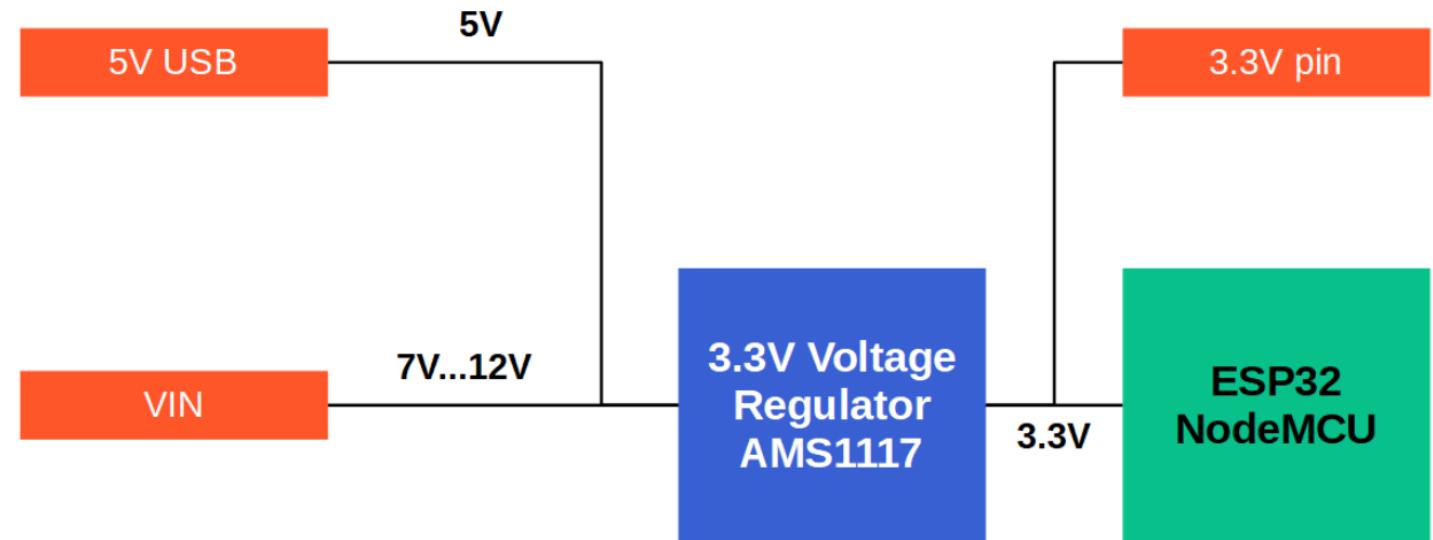
- Breadboard Tips
- Component Placement on Breadboard**
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises

Method 3



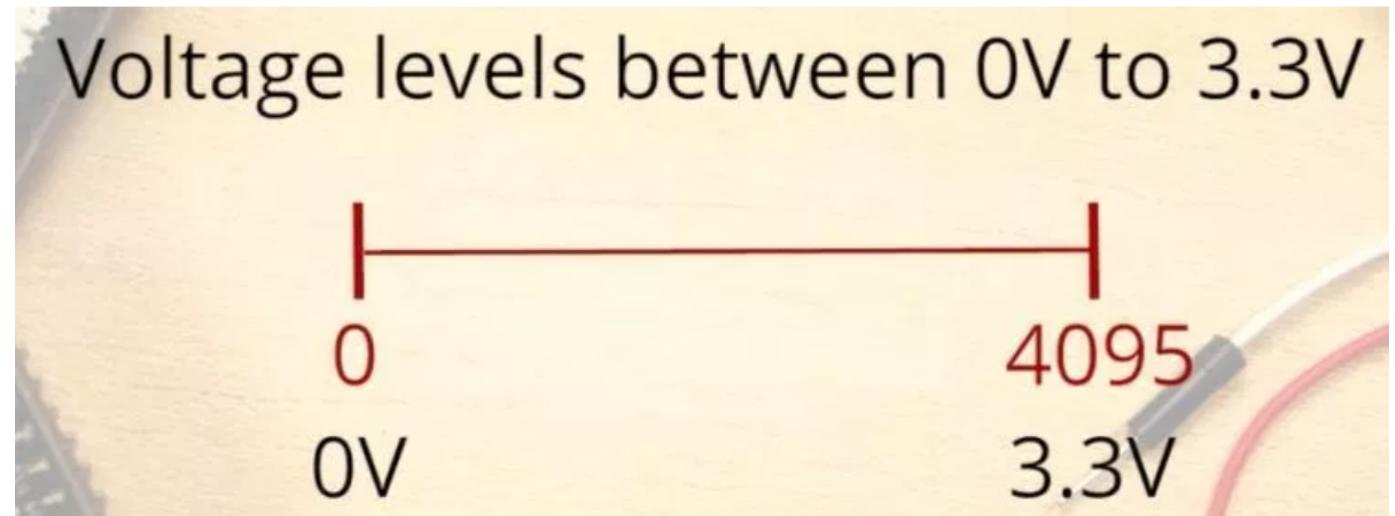
1iv. The Fundamental

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board**
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises



1iv. The Fundamental

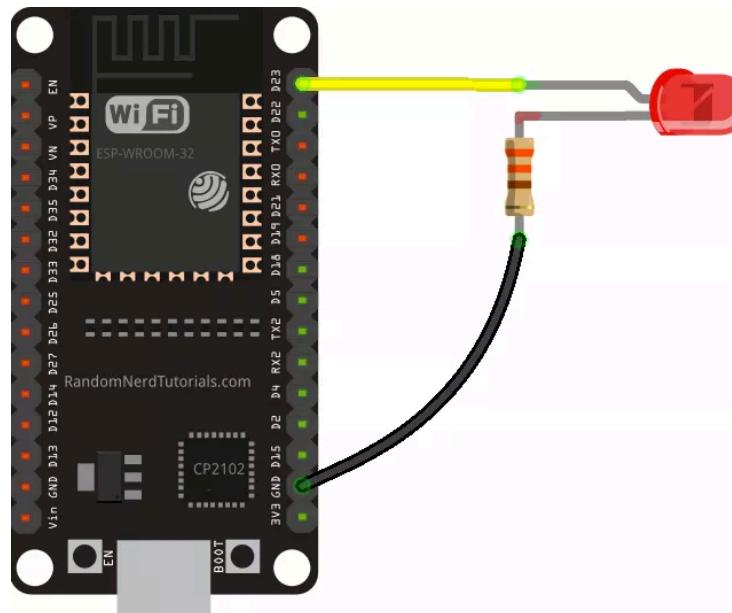
- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution**
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises



- **ESP32 ADC have 12 bits resolution or 2^{12} ranging from 0 – 4095**
- **Uno, NodeMCU/ESP8266 have 10 bits resolution (2^{10})**

1iv. The Fundamental

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- **Blink LED at Digital Port**
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises



- For LED, longer pin is the positive terminal;
- Uno, NodeMCU/ESP8266 have 10 bits resolution (2^{10});
- ESP32 has its own Built-in LED → GPIO2 @ D2.

→ From a total 30 pins, only 24 pins can be used as input & output;

→ GPIO34, GPIO35, GPIO36, GPIO37, GPIO38 & GPIO39 can be used as input pins only .

1iv. The Fundamental

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch**
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises

```
/* Blink */  
// ledPin refers to ESP32 GPIO 23  
const int ledPin = 23;  
// the setup function runs once when you press reset or power the board
```

```
void setup()  
{  
    // initialize digital pin ledPin as an output.  
    pinMode(ledPin, OUTPUT);  
}
```

// the loop function runs over and over again forever

```
void loop()  
{  
    digitalWrite(ledPin, HIGH); // turn the LED on (HIGH is the voltage level)  
    delay(1000); // wait for a second  
    digitalWrite(ledPin, LOW); // turn the LED off by making the voltage LOW  
    delay(1000); // wait for a second  
}
```

1iv. The Fundamental

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port**
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises

Hint 1:

```
void setup()
{ pinMode(pin number, OUTPUT); }

void loop()
{
    digitalWrite(pin number, LOW);
    digitalWrite(pin number, HIGH);
}
```

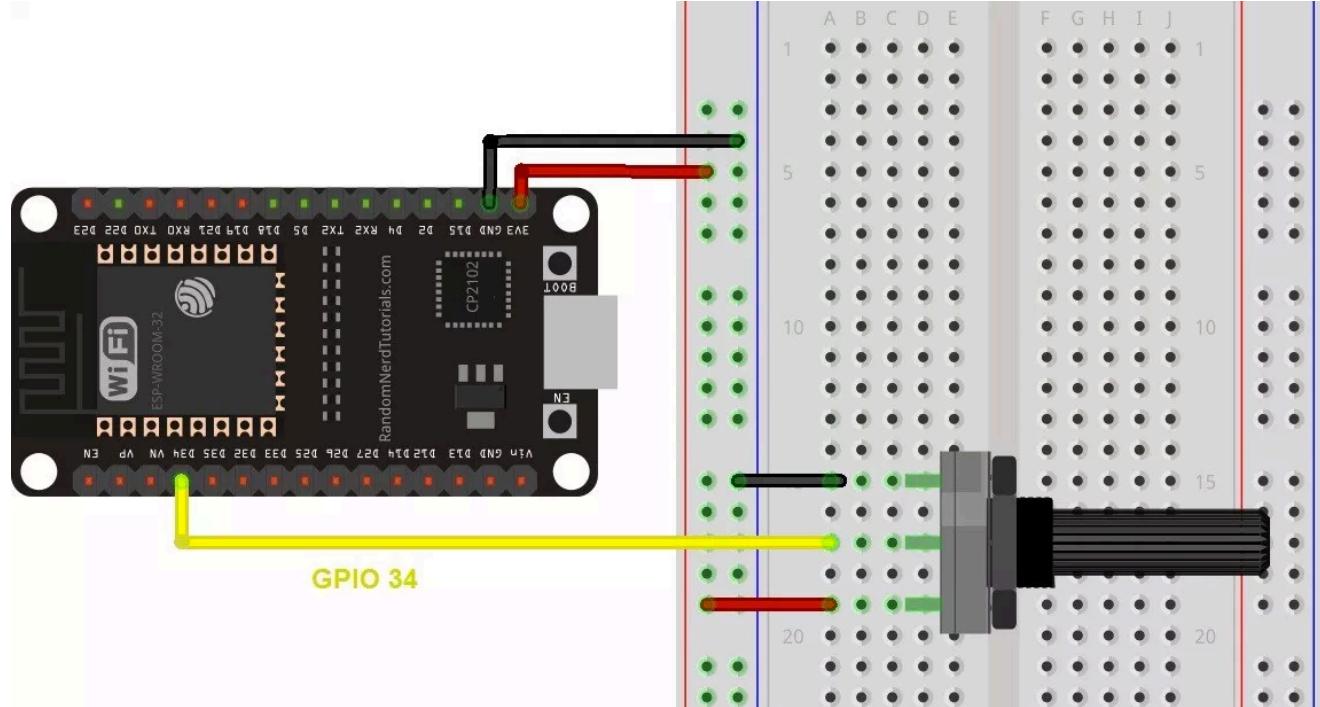
Hint 2:

```
void setup() {
{  pinMode(pin number, INPUT); }

void loop()
{  digitalRead(pin number); }
```

1iv. The Fundamental

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity**
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises



1iv. The Fundamental

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch**
- Hint for Analog Port
- Exercises

```
// Potentiometer is connected to GPIO 34 (Analog ADC1_CH6)
const int potPin = 34;
```

```
// variable for storing the potentiometer value
int potValue = 0;
```

```
void setup() {
    Serial.begin(115200);
    delay(1000);
}
```

```
void loop()
{ // Reading potentiometer value
    potValue = analogRead(potPin);
    Serial.println(potValue);
    delay(500);
}
```

1iv. The Fundamental

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port**
- Exercises

Hint:

```
void setup()  
{ ..... }  
  
void loop()  
{   analogRead(pin number); }
```

- Breadboard Tips
- Component Placement on Breadboard
- Power Supply on ESP32/NodeMCU Board
- ADC Bits Resolution
- Blink LED at Digital Port
- Blink Sketch
- Hints for Digital Port
- Capturing Analog Port Activity
- Analog Read Serial Sketch
- Hint for Analog Port
- Exercises

EXERCISES

1. Simulate a traffic light activity by using 3 LEDs, ESP32, breadboards and jumpers with delay of 2 seconds.
2. Develop a system that will fulfil the following statements:
 - if reading between 0 – 200: Green **LED ON**, the rest **OFF**;
 - else if reading between 201 – 400: Yellow **LED ON**, the rest **OFF**;
 - else if reading between 401 – 600: Red **LED ON**, the rest **OFF**;
 - else **ALL LED ON**.

1iv. The Fundamental

Breadboard Tips

Component Placement on Breadboard

Power Supply on ESP32/NodeMCU Board

ADC Bits Resolution

Blink LED at Digital Port

Blink Sketch

Hints for Digital Port

Capturing Analog Port Activity

Analog Read Serial Sketch

Hint for Analog Port

Exercises

Answers

Q1.

```
//PART2ii. Q1
const int led_red = 21; //gpio21
const int led_yellow = 22; //gpio22
const int led_green = 23; //gpio23

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    pinMode(led_red, OUTPUT);
    pinMode(led_yellow, OUTPUT);
    pinMode(led_green, OUTPUT);

    Serial.println("Initializing the leds");
    digitalWrite(led_green,HIGH);
    digitalWrite(led_yellow,HIGH);
    digitalWrite(led_red,HIGH);
    delay(2000);
    digitalWrite(led_green,0);
    digitalWrite(led_yellow,0);
    digitalWrite(led_red,0);
    Serial.println("Initialized");
    delay(2000);
}

void loop() {
    // put your main code here, to run repeatedly:
    //green led ON
    Serial.println("GREEN ON");
    digitalWrite(led_green,1);
    digitalWrite(led_yellow,0);
    digitalWrite(led_red,0);
    delay(2000);

    //yellow led ON
    Serial.println("YELLOW ON");
    digitalWrite(led_green,0);
    digitalWrite(led_yellow,1);
    digitalWrite(led_red,0);
    delay(2000);

    //red led ON
    Serial.println("RED ON");
    digitalWrite(led_green,0);
    digitalWrite(led_yellow,0);
    digitalWrite(led_red,1);
    Serial.println("-----");
    delay(2000);
}
```

Q2.

```
//PART2ii. Q2
const int led_red = 21; //gpio21
const int led_yellow = 22; //gpio22
const int led_green = 23; //gpio23
const int pot_pin = 34; //gpio34

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    pinMode(led_red, OUTPUT);
    pinMode(led_yellow, OUTPUT);
    pinMode(led_green, OUTPUT);

    Serial.println("Initializing the leds");
    digitalWrite(led_green,HIGH);
    digitalWrite(led_yellow,HIGH);
    digitalWrite(led_red,HIGH);
    delay(2000);
    digitalWrite(led_green,0);
    digitalWrite(led_yellow,0);
    digitalWrite(led_red,0);
    Serial.println("Initialized");
    delay(2000);
}

void loop() {
    // put your main code here, to run repeatedly:
    //input from potentiometer
    int getinput = analogRead(pot_pin);
    Serial.print("Potentiometer: ");
    Serial.println(getinput);
    if (getinput<=200)
    {
        //green led ON
        Serial.println("GREEN ON");
        digitalWrite(led_green,1);
        digitalWrite(led_yellow,0);
        digitalWrite(led_red,0);
    }
    else if ((getinput>200) && (getinput<=400))
    {
        //yellow led ON
        Serial.println("YELLOW ON");
        digitalWrite(led_green,0);
        digitalWrite(led_yellow,1);
        digitalWrite(led_red,0);
    }
    else if ((getinput>400) && (getinput<=600))
    {
        //red led ON
        Serial.println("RED ON");
        digitalWrite(led_green,0);
        digitalWrite(led_yellow,0);
        digitalWrite(led_red,1);
    }
    else
    {
        Serial.println("ALL LED ON");
        digitalWrite(led_green,1);
        digitalWrite(led_yellow,1);
        digitalWrite(led_red,1);
    }
}
```