



# Clasificador de tarjetas

---

M.I. Sergio Hernández Sánchez  
[sergio.hernandez.sanchez@itam.mx](mailto:sergio.hernandez.sanchez@itam.mx)

---

# Entradas Digitales

# ESP32 DEVKIT v1 - PINOUT

# GPIO = General Purpose Input/Output

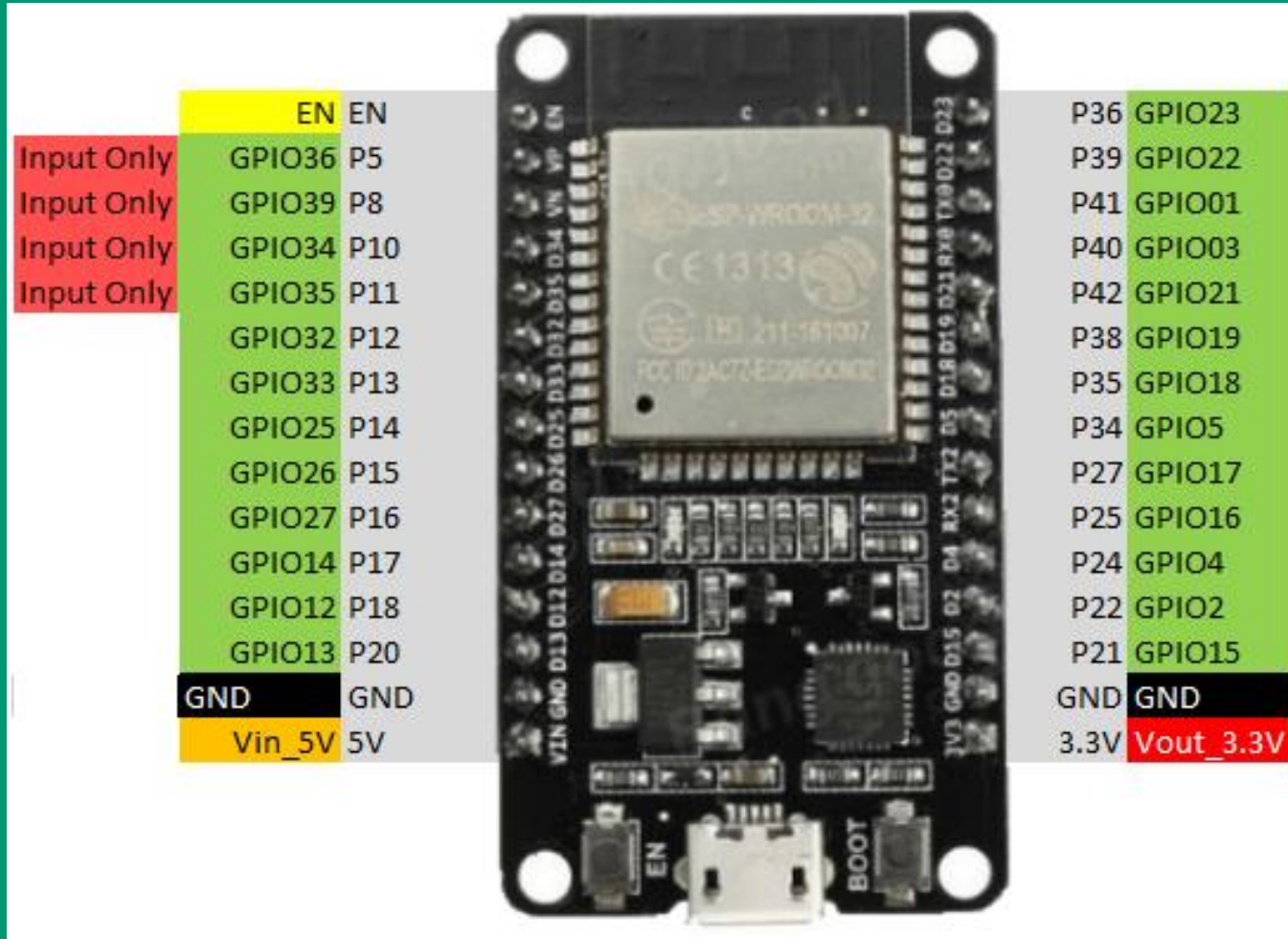


		RESET	EN	EN
Input Only	RTC0	ADC1_0	GPIO36	P5
Input Only	RTC3	ADC1_3	GPIO39	P8
Input Only	RTC4	ADC1_6	GPIO34	P10
Input Only	RTC5	ADC1_7	GPIO35	P11
RTC9	TOUCH_9	ADC1_4	GPIO32	P12
RTC8	TOUCH_8	ADC1_5	GPIO33	P13
	DAC0	ADC1_8	GPIO25	P14
	DAC1	ADC1_9	GPIO26	P15
	TOUCH_7	ADC2_7	GPIO27	P16
	TOUCH_6	ADC2_6	GPIO14	P17
	TOUCH_5	ADC2_5	GPIO12	P18
	TOUCH_4	ADC2_4	GPIO13	P20
			GND	GND
			Vin_5V	5V



P36	GPIO23	V_SPI_D	MOSI	
P39	GPIO22	V_SPI_WP	I2C_SCL	
P41	GPIO01	TX0		
P40	GPIO03	RX0		
P42	GPIO21	V_SPI_HD	I2C_SDA	
P38	GPIO19	V_SPI_Q	MISO	
P35	GPIO18	V_SPI_CLK	SCK	
P34	GPIO5	V_SPI_CS0	SS	
P27	GPIO17	TX2		
P25	GPIO16	RX2		
P24	GPIO4	ADC2_0	TOUCH_0	
P22	GPIO2	ADC2_2	TOUCH_2	LED_0
P21	GPIO15	ADC2_3	TOUCH_3	
GND	GND			
3.3V	Vout_3.3V			

# Salidas digitales



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

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

---

# Comunicación Serial

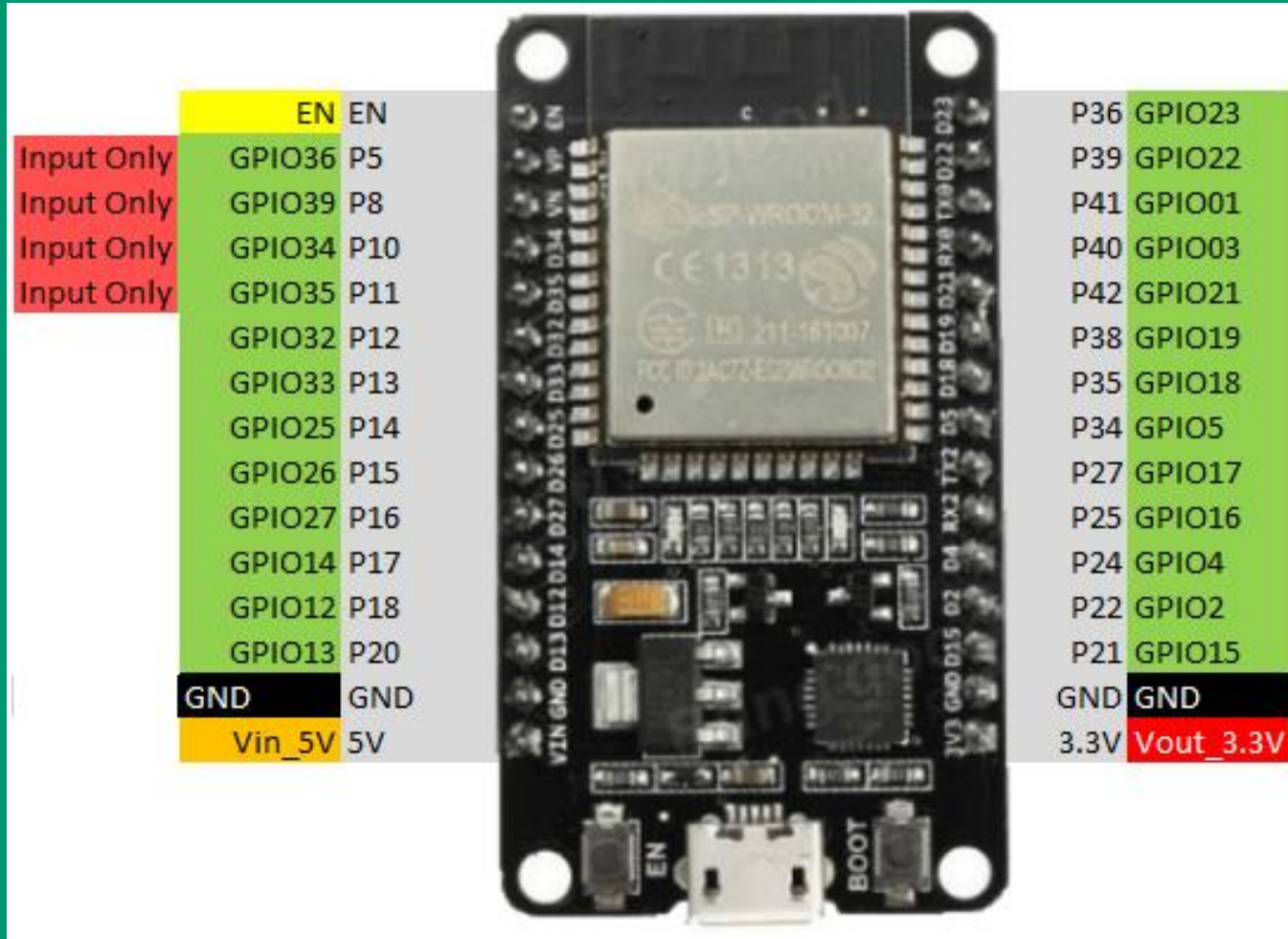
# Comunicación Serial

---

```
void setup() {  
    Serial.begin(115200);  
}  
  
void loop() {  
    Serial.print("texto");  
    Serial.print(variable);  
    Serial.println("texto_con_salto_de_línea");  
    Serial.println(variable_con_salto_de_línea);  
}
```



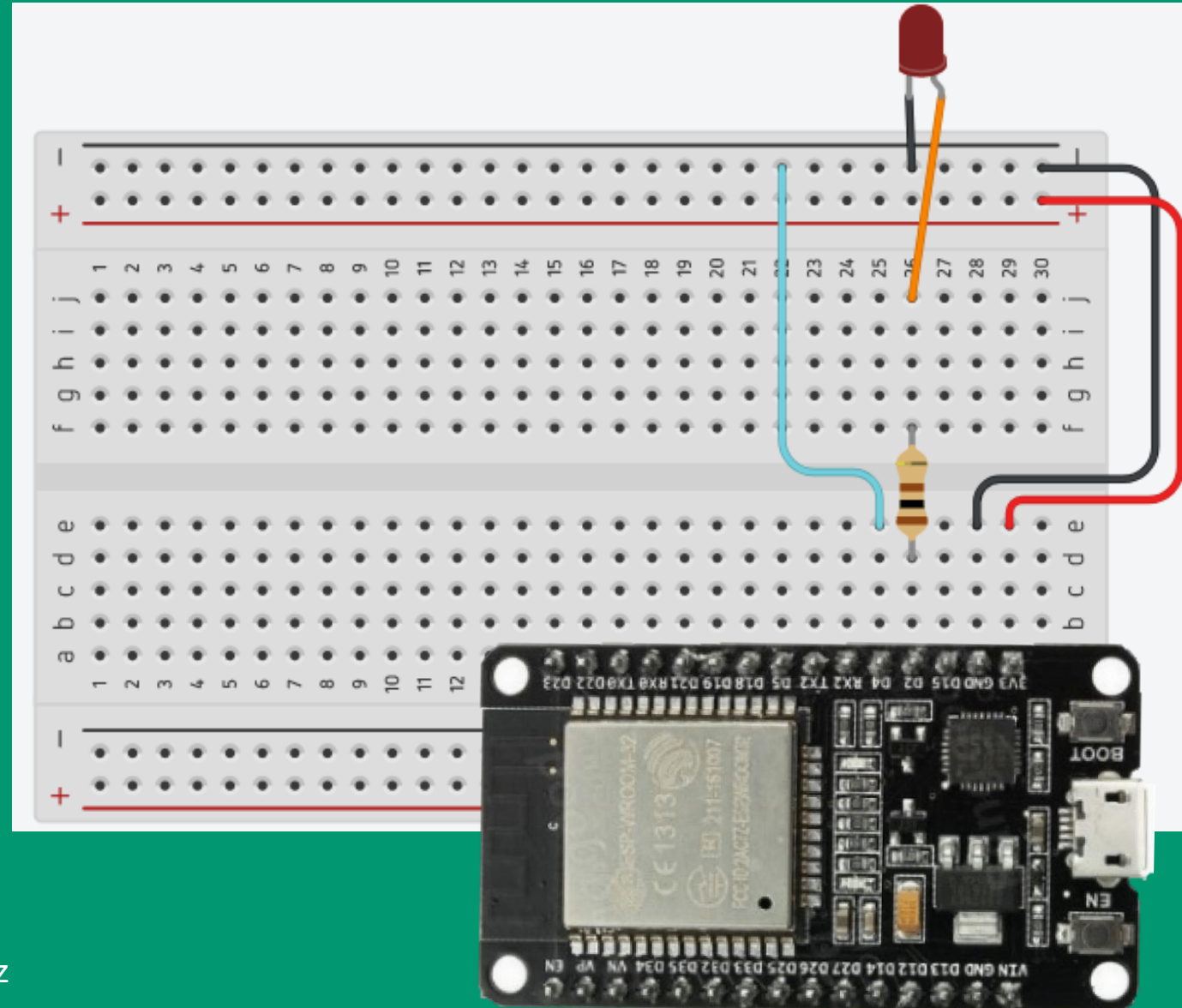
# Entradas digitales



```
void setup() {
    pinMode(pin_name, INPUT);
}

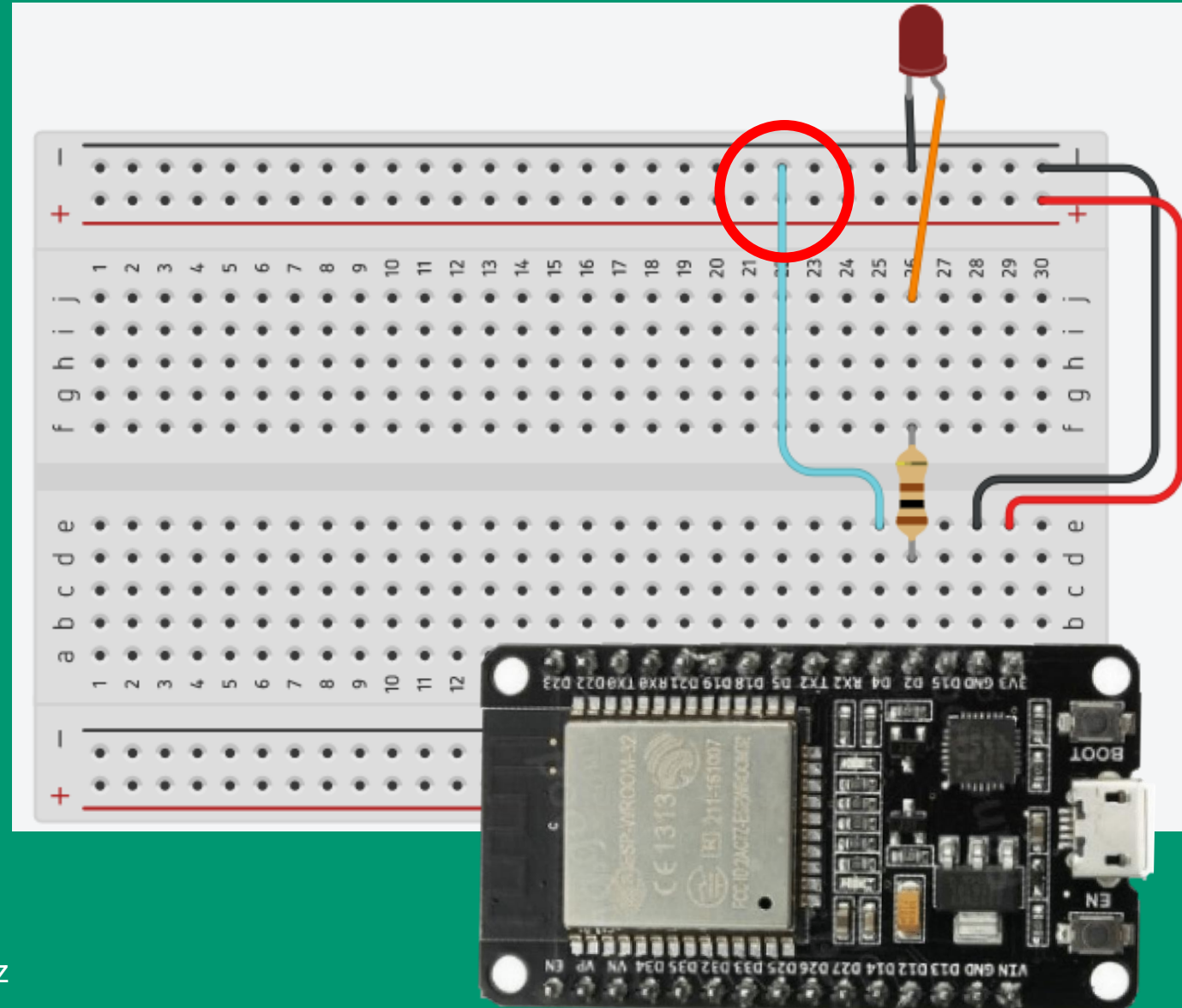
void loop() {
    digitalWrite(pin_name);
}
```

# Entradas digitales

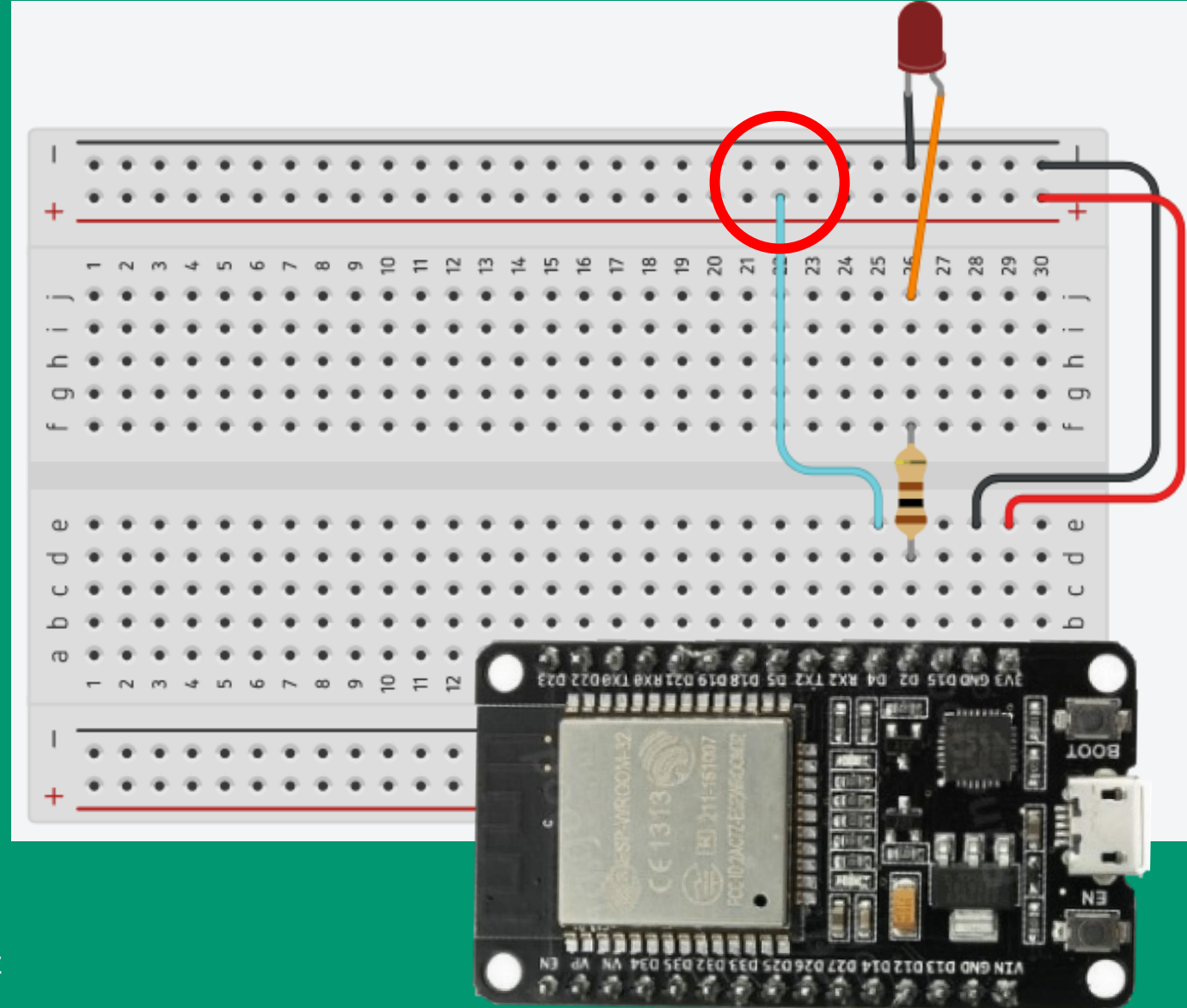




# Entradas digitales



# Entradas digitales



# Entradas digitales

```
#define LED 2
#define entrada 4

bool dato;

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

    pinMode(LED, OUTPUT);
    pinMode(entrada, INPUT);
}
```

```
void loop() {
    dato = digitalRead(entrada);

    if (dato == 1) {
        digitalWrite(LED, HIGH);
        Serial.print("La lectura es de: ");
        Serial.println(dato);
    }

    else {
        digitalWrite(LED, LOW);
        Serial.print("La lectura es de: ");
        Serial.println(dato);
    }
}
```

---

# Funciones

# Funciones

```
void setup() {  
    //Configuracion  
}  
  
void loop() {  
    nombre_funcion()  
}  
  
void nombre_funcion() {  
    //Código_definido  
}
```

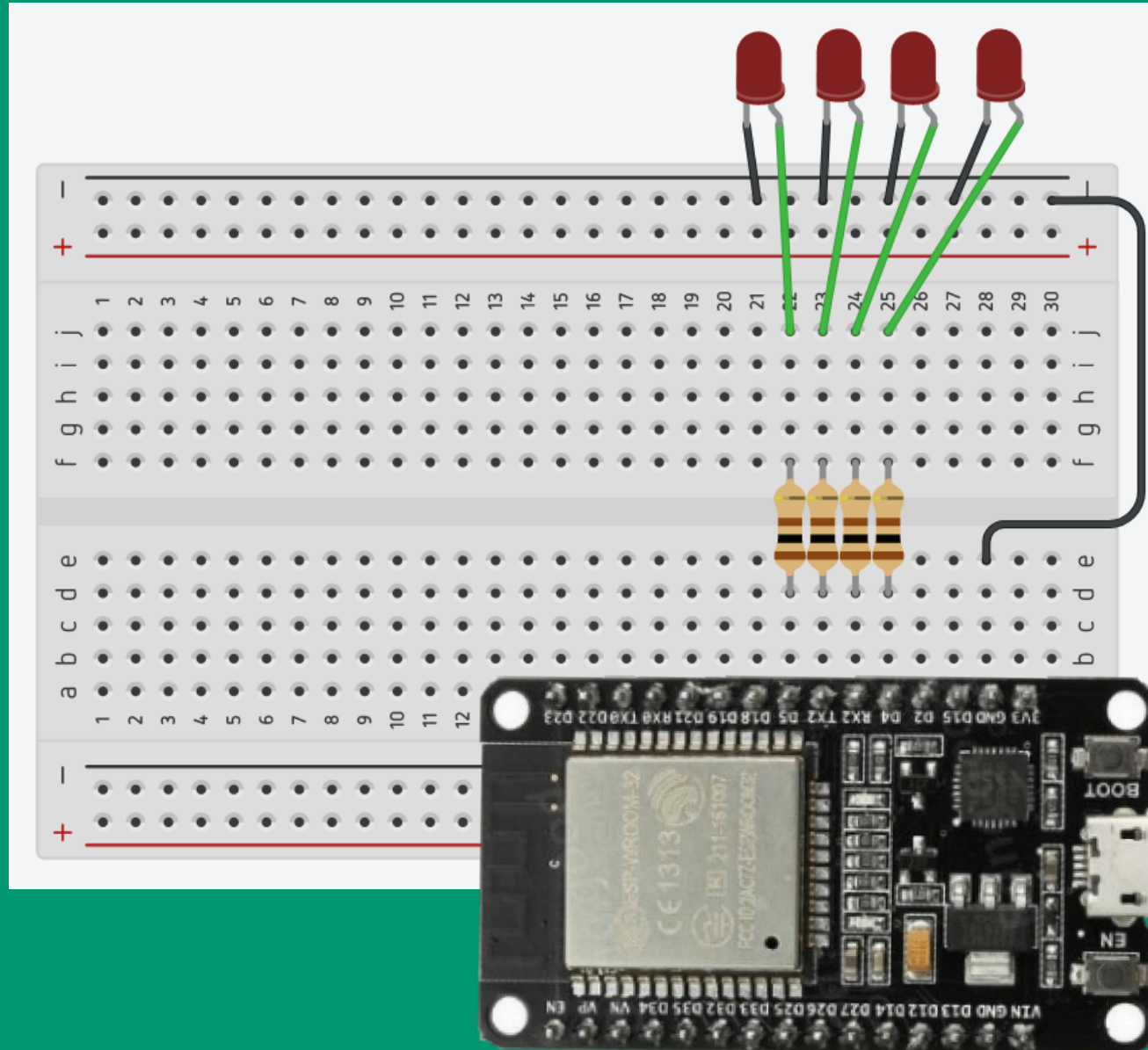
```
void setup() {  
    //Configuracion  
}  
  
void loop() {  
    nombre_funcion(parametro)  
}  
  
void nombre_funcion(int dato) {  
    //Código_definido  
}
```

---

# Codificador binario



# Circuito codificación



# Reto

---

Hacer un contador binario de 0 a 15, incrementando la cuenta de uno en uno cada segundo y se reinicie para volver a hacer la cuenta. Este incremento se debe mostrar con cuatro leds y a su vez, mostrarse en el monitor serial.