

# Nivel Avanzado

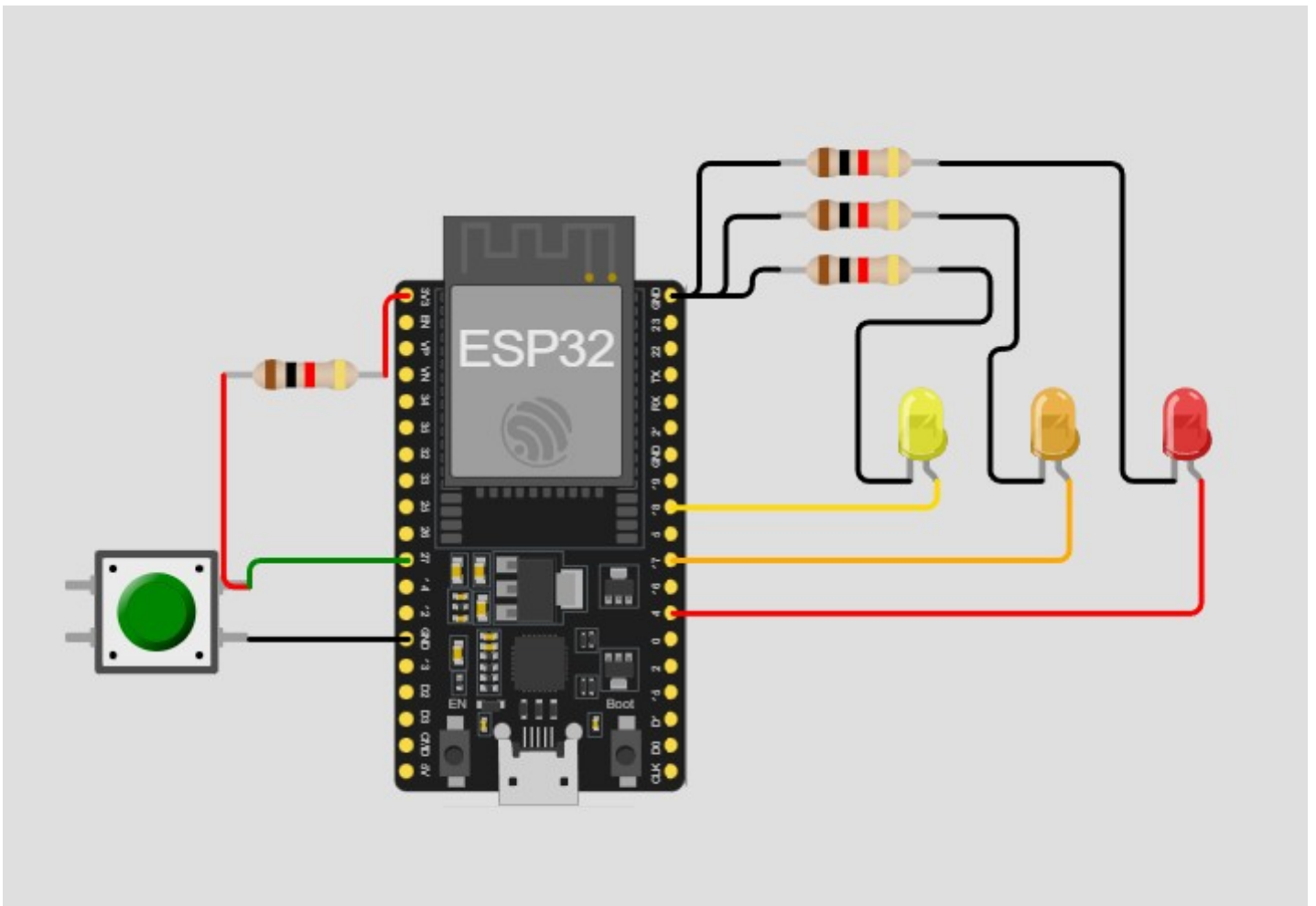
## Ejercicio 9: Secuencia de LEDs con botón

- Crea una secuencia de luces que avance cada vez que se presione btn1.

<https://wokwi.com/projects/396897155125895169>

```
4
5  #include <Arduino.h>
6
7  #define LED_PIN_1 18
8  #define LED_PIN_2 17
9  #define LED_PIN_3 4
10 #define BUTTON_PIN 27
11
12 int secuencia = 0;
13
14 void setup() {
15     pinMode(LED_PIN_1, OUTPUT);
16     pinMode(LED_PIN_2, OUTPUT);
17     pinMode(LED_PIN_3, OUTPUT);
18     pinMode(BUTTON_PIN, INPUT_PULLUP);
19 }
20
21
22 void loop() {
23     if (digitalRead(BUTTON_PIN) == LOW) {
24         secuencia++;
25         if (secuencia > 2) {
26             secuencia = 0;
27         }
28         digitalWrite(LED_PIN_1, LOW);
29         digitalWrite(LED_PIN_2, LOW);
30         digitalWrite(LED_PIN_3, LOW);
31     }
```

```
31
32     if (secuencia == 0) {
33         digitalWrite(LED_PIN_1, HIGH);
34         delay(150);
35         digitalWrite(LED_PIN_1, LOW);
36         delay(150);
37         digitalWrite(LED_PIN_2, HIGH);
38         delay(150);
39         digitalWrite(LED_PIN_2, LOW);
40         delay(150);
41         digitalWrite(LED_PIN_3, HIGH);
42         delay(150);
43         digitalWrite(LED_PIN_3, LOW);
44     } else if (secuencia == 1) {
45         digitalWrite(LED_PIN_3, HIGH);
46         delay(150);
47         digitalWrite(LED_PIN_3, LOW);
48         delay(150);
49         digitalWrite(LED_PIN_2, HIGH);
50         delay(150);
51         digitalWrite(LED_PIN_2, LOW);
52         delay(150);
53         digitalWrite(LED_PIN_1, HIGH);
54         delay(150);
55         digitalWrite(LED_PIN_1, LOW);
56     } else if (secuencia == 2) {
57         digitalWrite(LED_PIN_2, HIGH);
58         delay(150);
59         digitalWrite(LED_PIN_2, LOW);
60         delay(150);
61         digitalWrite(LED_PIN_3, HIGH);
62         delay(150);
63         digitalWrite(LED_PIN_3, LOW);
64         delay(150);
65         digitalWrite(LED_PIN_1, HIGH);
66         delay(150);
67         digitalWrite(LED_PIN_1, LOW);
68     }
69     delay(200);
70 }
71 }
72
```



## Ejercicio 10: Control de velocidad de parpadeo con dip switch

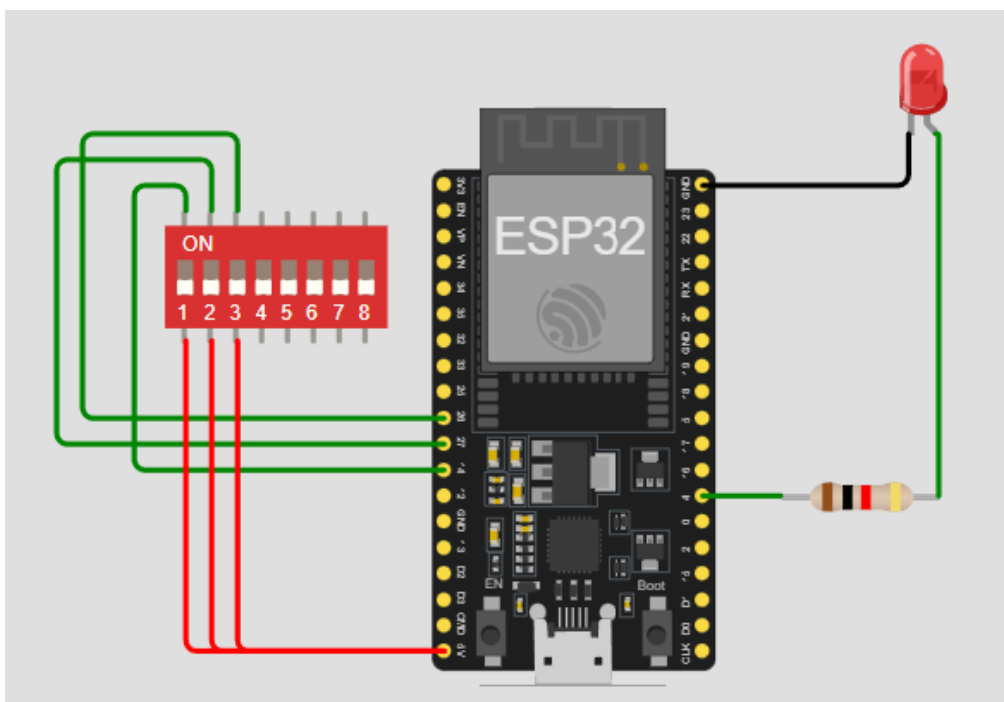
- Utiliza los dip switches sw1.1 a sw1.3 para controlar la velocidad de parpadeo de led1, asignando distintas velocidades.

<https://wokwi.com/projects/396874723523880961>

```

3  #include <Arduino.h>
4
5  #define PIN1 14
6  #define PIN2 27
7  #define PIN3 26
8  #define LED 4
9
10 void setup() {
11     pinMode(PIN1, INPUT);
12     pinMode(PIN2, INPUT);
13     pinMode(PIN3, INPUT);
14     pinMode(LED, OUTPUT);
15 }
16
17 void loop() {
18     int reading1 = digitalRead(PIN1);
19     int reading2 = digitalRead(PIN2);
20     int reading3 = digitalRead(PIN3);
21
22     if (reading1 == LOW && reading2 == LOW && reading3 == LOW) {
23         digitalWrite(LED, HIGH);
24         delay(1000);
25         digitalWrite(LED, LOW);
26         delay(1000);
27     } else if (reading1 == LOW && reading2 == LOW && reading3 == HIGH) {
28         digitalWrite(LED, HIGH);
29         delay(500);
30         digitalWrite(LED, LOW);
31         delay(500);
32     } else if (reading1 == LOW && reading2 == HIGH && reading3 == LOW) {
33         digitalWrite(LED, HIGH);
34         delay(200);
35         digitalWrite(LED, LOW);
36         delay(200);
37     } else if (reading1 == LOW && reading2 == HIGH && reading3 == HIGH) {
38         digitalWrite(LED, HIGH);
39         delay(100);
40         digitalWrite(LED, LOW);
41         delay(100);
42     } else {
43         digitalWrite(LED, LOW);
44     }
45 }

```



## Ejercicio 11: Patrón de parpadeo de LEDs con dip switches

- Establece un patrón de parpadeo para los led1 a led8 basado en la combinación de estados de sw1.1 a sw1.4. Por ejemplo, cada posición activa del switch puede representar un patrón diferente (como parpadeo rápido, lento, secuencial, etc.).

<https://wokwi.com/projects/397245698145516545>

```
4  #include <Arduino.h>
5
6  #define led1 22
7  #define pul1 35
8  #define led2 23
9  #define pul2 32
10 #define led3 21
11 #define pul3 33
12 #define led4 19
13 #define pul4 25
14 #define led5 18
15 #define led6 5
16 #define led7 16
17 #define led8 2
18
19 void setup() {
20   Serial.begin(115200);
21   pinMode(led1, OUTPUT);
22   pinMode(led2, OUTPUT);
23   pinMode(led3, OUTPUT);
24   pinMode(led4, OUTPUT);
25   pinMode(led5, OUTPUT);
26   pinMode(led6, OUTPUT);
27   pinMode(led7, OUTPUT);
28   pinMode(led8, OUTPUT);
29 }
30
31 void loop() {
32   if((digitalRead(pul1))==HIGH){
33     digitalWrite(led1, 1);
34     delay(200);
35     digitalWrite(led2, 1);
36     delay(200);
37     digitalWrite(led3, 1);
38     delay(200);
39     digitalWrite(led4, 1);
40     delay(200);
41     digitalWrite(led1, 0);
42     delay(200);
43     digitalWrite(led2, 0);
44     delay(200);
45     digitalWrite(led3, 0);
46     delay(200);
47     digitalWrite(led4, 0);
48   }
```

```
45     digitalWrite(led3, 0);
46     delay(200);
47     digitalWrite(led4, 0);
48 }
49 else if ((digitalRead(pul1))==LOW){
50     digitalWrite(led1, 0);
51     digitalWrite(led2, 0);
52     digitalWrite(led3, 0);
53     digitalWrite(led4, 0);
54 }
55 if((digitalRead(pul2))==HIGH){
56     digitalWrite(led1, 1);
57     delay(200);
58     digitalWrite(led3, 1);
59     delay(200);
60     digitalWrite(led6, 1);
61     delay(200);
62     digitalWrite(led8, 1);
63     delay(200);
64     digitalWrite(led1, 0);
65     delay(200);
66     digitalWrite(led3, 0);
67     delay(200);
68     digitalWrite(led6, 0);
69     delay(200);
70     digitalWrite(led8, 0);
71 }
72 else if ((digitalRead(pul2))==LOW){
73     digitalWrite(led1, 0);
74     digitalWrite(led3, 0);
75     digitalWrite(led6, 0);
76     digitalWrite(led8, 0);
77 }
78 if((digitalRead(pul3))==HIGH){
79     digitalWrite(led1, 1);
80     delay(500);
81     digitalWrite(led2, 1);
82     delay(500);
83     digitalWrite(led3, 1);
84     delay(500);
85     digitalWrite(led4, 1);
86     delay(500);
```

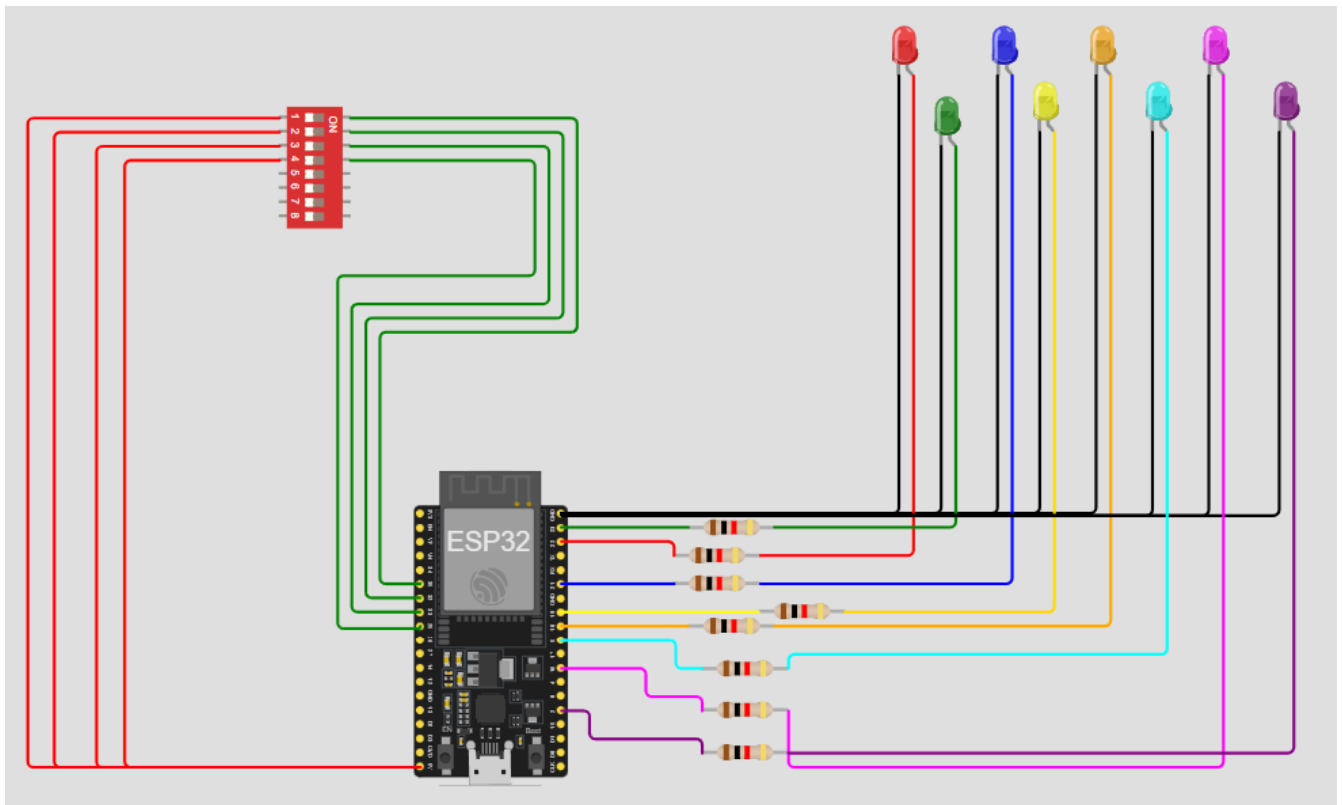
```

86     delay(500);
87     digitalWrite(led5, 1);
88     delay(500);
89     digitalWrite(led6, 1);
90     delay(500);
91     digitalWrite(led7, 1);
92     delay(500);
93     digitalWrite(led8, 1);
94     delay(500);
95     digitalWrite(led1, 0);
96     delay(500);
97     digitalWrite(led2, 0);
98     delay(500);
99     digitalWrite(led3, 0);
100    delay(500);
101    digitalWrite(led4, 0);
102    delay(500);
103    digitalWrite(led5, 0);
104    delay(500);
105    digitalWrite(led6, 0);
106    delay(500);
107    digitalWrite(led7, 0);
108    delay(500);
109    digitalWrite(led8, 0);
110 }
111 else if ((digitalRead(pul3))==LOW){
112     digitalWrite(led1, 0);
113     digitalWrite(led2, 0);
114     digitalWrite(led3, 0);
115     digitalWrite(led4, 0);
116     digitalWrite(led5, 0);
117     digitalWrite(led6, 0);
118     digitalWrite(led7, 0);
119     digitalWrite(led8, 0);
120 }
121 if((digitalRead(pul4))==HIGH){
122     digitalWrite(led8, 1);
123     delay(100);
124     digitalWrite(led7, 1);
125     delay(100);
126     digitalWrite(led6, 1);

```

```
126     digitalWrite(led6, 1);
127     delay(100);
128     digitalWrite(led5, 1);
129     delay(100);
130     digitalWrite(led4, 1);
131     delay(100);
132     digitalWrite(led3, 1);
133     delay(100);
134     digitalWrite(led2, 1);
135     delay(100);
136     digitalWrite(led1, 1);
137     delay(100);
138     digitalWrite(led8, 0);
139     delay(100);
140     digitalWrite(led7, 0);
141     delay(100);
142     digitalWrite(led6, 0);
143     delay(100);
144     digitalWrite(led5, 0);
145     delay(100);
146     digitalWrite(led4, 0);
147     delay(100);
148     digitalWrite(led3, 0);
149     delay(100);
150     digitalWrite(led2, 0);
151     delay(100);
152     digitalWrite(led1, 0);
153 }
154 ✓ else if ((digitalRead(pul4))==LOW){
155     digitalWrite(led1, 0);
156     digitalWrite(led2, 0);
157     digitalWrite(led3, 0);
158     digitalWrite(led4, 0);
159     digitalWrite(led5, 0);
160     digitalWrite(led6, 0);
161     digitalWrite(led7, 0);
162     digitalWrite(led8, 0);
163 }
164 ✓ else{
165     delay(300);
166 }
167 }
168
```



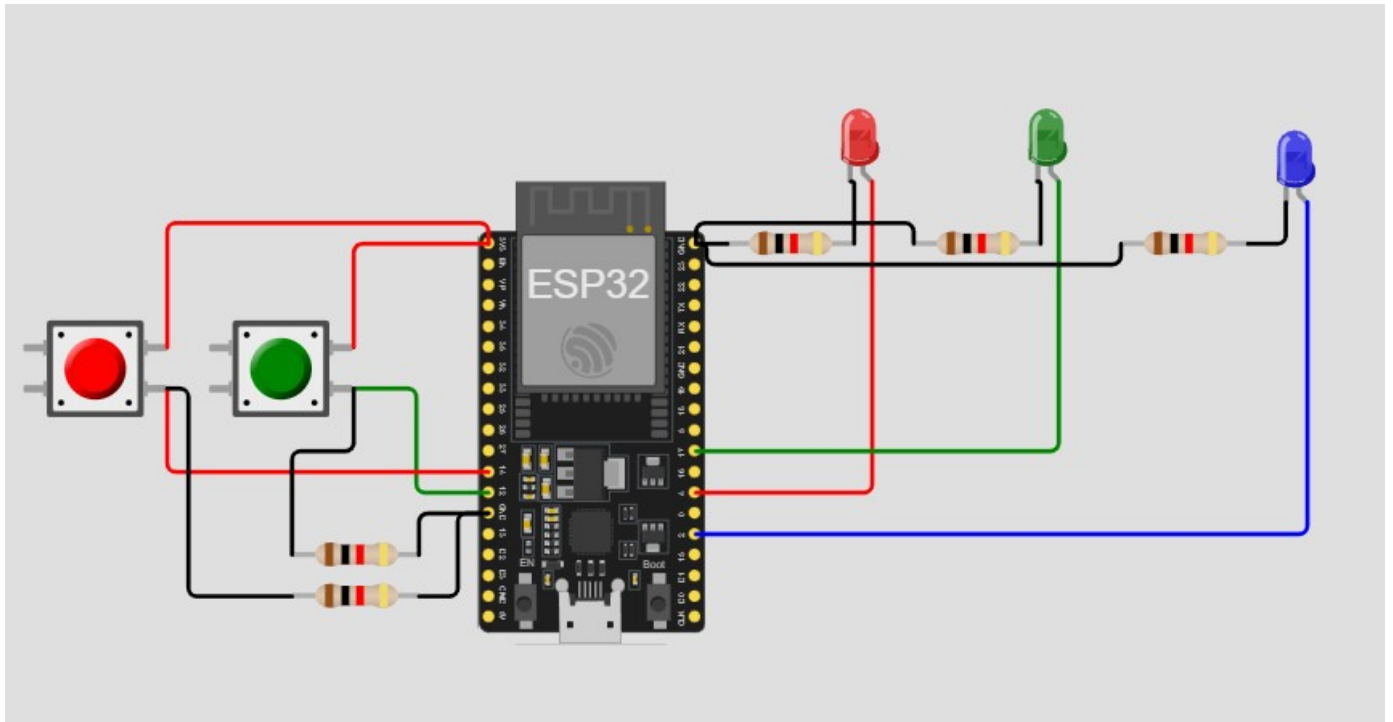


## Ejercicio 12: Medidor de pulsaciones

- Programa un contador de pulsaciones utilizando btn1. El número de pulsaciones debe mostrarse en una secuencia de LEDs (por ejemplo, led5 a led8 donde cada LED representa una cantidad de pulsaciones).

<https://wokwi.com/projects/397280394604626945>

```
3  #include <Arduino.h>
4  #define LED1 4
5  #define LED2 17
6  #define btn1 12
7  #define btn2 14
8  #define LED3 2
9
10 int contador = 0;
11 void setup() {
12     Serial.begin(115200);
13     pinMode(LED1, OUTPUT);
14     pinMode(LED2, OUTPUT);
15     pinMode(LED3, OUTPUT);
16     pinMode(btn1, INPUT);
17     pinMode(btn2, INPUT);
18 }
19
20 void loop() {
21     // contador de pulsacion
22     if(digitalRead(btn1)==HIGH){
23         contador++;
24         digitalWrite(LED3, LOW);
25         digitalWrite(LED2, HIGH);
26         delay(250);
27         digitalWrite(LED2, LOW);
28         delay(250);
29         Serial.print("Cantidad de veces presionado es: ");
30         Serial.println(contador);
31     }
32     // restador de pulsacion
33     if(digitalRead(btn2)==HIGH){
34         contador--;
35         digitalWrite(LED3, LOW);
36         digitalWrite(LED1, HIGH);
37         delay(250);
38         digitalWrite(LED1, LOW);
39         delay(250);
40         Serial.print("Cantidad de veces presionado es: ");
41         Serial.println(contador);
42     }
43     // luz testigo de encendido y espera
44     else{
45         digitalWrite(LED3, HIGH);
46     }
```



### Ejercicio 13: Contraseña con botones

- Implementa un sistema de contraseña usando btn1, btn2, y btn3 donde una secuencia específica de pulsaciones activa led1. Si la secuencia es incorrecta, led2 debería encenderse.

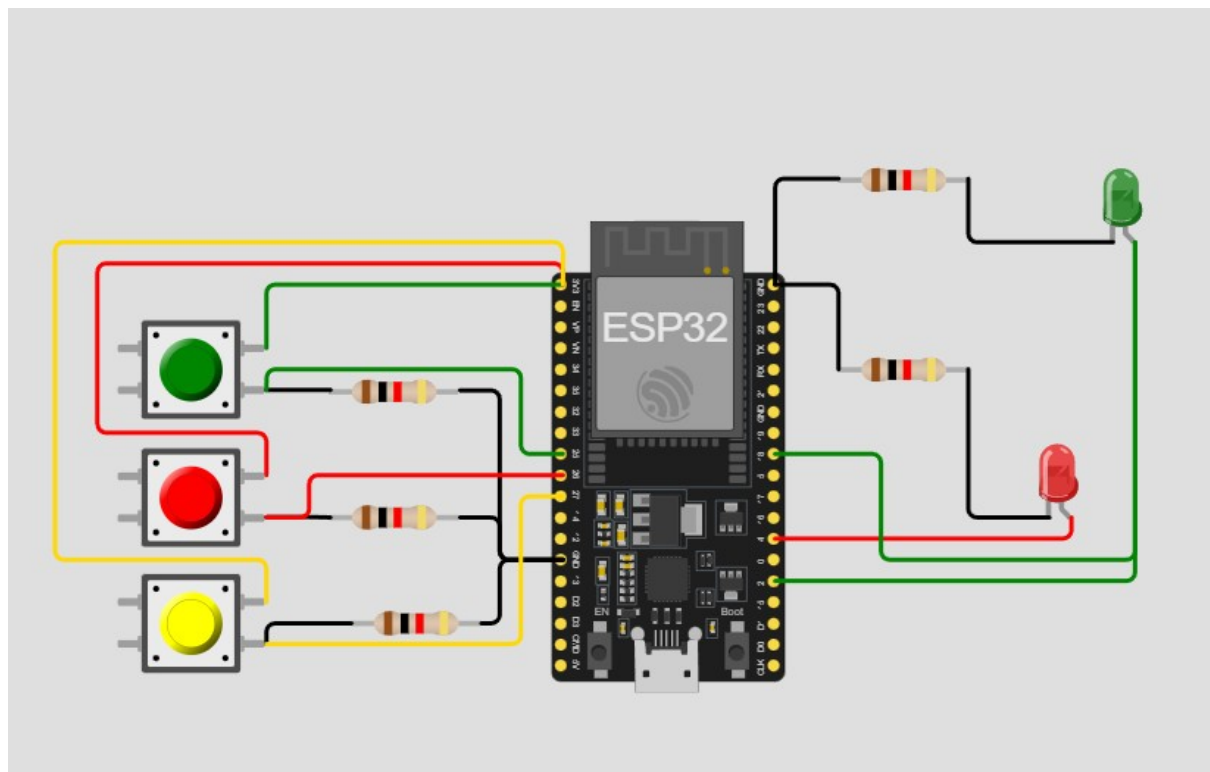
<https://wokwi.com/projects/397336748356052993>

```
5  #include <Arduino.h>
6
7  #define LED1 18
8  #define LED2 4
9  #define btn1 27
10 #define btn2 26
11 #define btn3 25
12
13 int lectura1 = 0;
14 int lectura2 = 0;
15 int lectura3 = 0;
16
17 void setup() {
18     Serial.begin(115200);
19     pinMode(LED1, OUTPUT);
20     pinMode(LED2, OUTPUT);
21     pinMode(btn1, INPUT);
22     pinMode(btn2, INPUT);
23     pinMode(btn3, INPUT);
24 }
25
26 void loop() {
27     if (digitalRead(btn1)==HIGH){
28         lectura1++;
29         Serial.print("*");
30         delay(250);
31     }
32     else if(digitalRead(btn2)==HIGH){
33         lectura2++;
34         Serial.print("*");
35         delay(250);
36     }
37     else if(digitalRead(btn3)==HIGH){
38         lectura3++;
39         Serial.print("*");
40         delay(250);
41     }
42     if (lectura1 == 2 && lectura2 == 1 && lectura3 == 3){
43         digitalWrite(LED1, HIGH);
44         delay(1500);
45         digitalWrite(LED1, LOW);
46         delay(1000);
47         Serial.println(" PUEDE PASAR. ");
48     }
```

```

36     }
37     else if(digitalRead(btn3)==HIGH){
38         lectura3++;
39         Serial.print("*");
40         delay(250);
41     }
42     if (lectura1 == 2 && lectura2 == 1 && lectura3 == 3){
43         digitalWrite(LED1, HIGH);
44         delay(1500);
45         digitalWrite(LED1, LOW);
46         delay(1000);
47         Serial.println(" PUEDE PASAR. ");
48     }
49     else if (lectura1 > 2 || lectura2 > 1 || lectura3 > 3) {
50         digitalWrite(LED2, HIGH);
51         delay(1000);
52         digitalWrite(LED2, LOW);
53         lectura1 = 0;
54         lectura2 = 0;
55         lectura3 = 0;
56         Serial.println(" CONTRASEÑA INCORRECTA. ");
57     }
58
59
60
61 }
62

```



## Ejercicio 14: Aplicación de timers para control de LEDs

- Utiliza el temporizador del ESP32 para controlar el parpadeo de led1 a led4 sin usar la función delay(), permitiendo que el programa ejecute otras tareas mientras los LEDs parpadean.

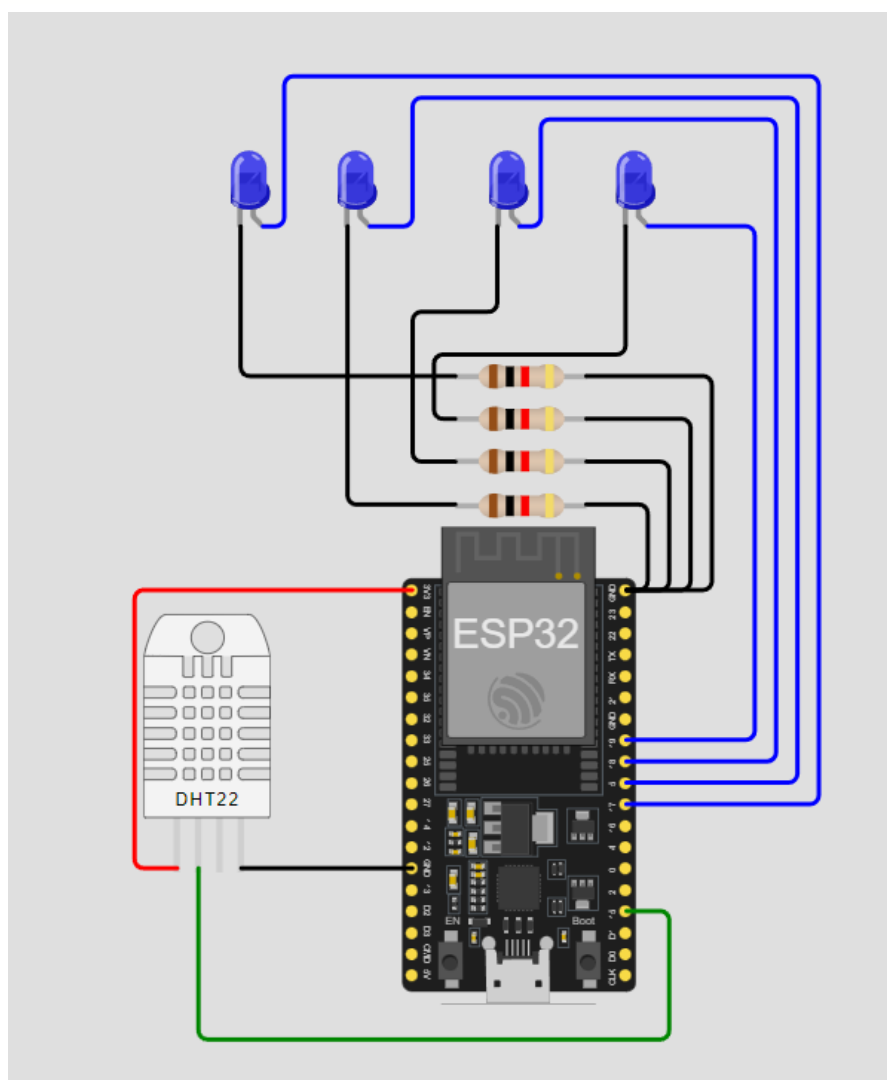
<https://wokwi.com/projects/397517186514343937>

```
5  #include <Arduino.h>
6  #include <DHT.h>
7
8
9  #define LED1 5
10 #define LED2 18
11 #define LED3 19
12 #define LED4 17
13 #define parpadeotime 500
14 #define DHTsensor 15
15
16 bool estado1 = LOW;
17 bool estado2 = LOW;
18 bool estado3 = LOW;
19 bool estado4 = LOW;
20
21 DHT dht(DHTsensor, DHT22);
22
23 // Función de interrupción para el temporizador
24 void IRAM_ATTR interrupcion() {
25     estado1 = !estado1;
26     estado2 = !estado2;
27     estado3 = !estado3;
28     estado4 = !estado4;
29
30     digitalWrite(LED1, estado1);
31     digitalWrite(LED2, estado2);
32     digitalWrite(LED3, estado3);
33     digitalWrite(LED4, estado4);
34 }
35
36 void setup() {
37     Serial.begin(115200);
38     dht.begin();
39     pinMode(LED1, OUTPUT);
40     pinMode(LED2, OUTPUT);
41     pinMode(LED3, OUTPUT);
42     pinMode(LED4, OUTPUT);
43
44     // Inicializar el temporizador
45     hw_timer_t *Temporizador = timerBegin(0, 80, true);
46     timerAttachInterrupt(Temporizador, &interrupcion, true);
47     timerAlarmWrite(Temporizador, parpadeotime * 1000, true);
48     timerAlarmEnable(Temporizador);
49 }
```

```

44 // Inicializar el temporizador
45 hw_timer_t *Temporizador = timerBegin(0, 80, true);
46 timerAttachInterrupt(Temporizador, &interrupcion, true);
47 timerAlarmWrite(Temporizador, parpadeotime * 1000, true);
48 timerAlarmEnable(Temporizador);
49 }
50
51 void loop() {
52 // Leer la temperatura y humedad del sensor DHT22
53 float temperatura = dht.readTemperature();
54 float humedad = dht.readHumidity();
55
56 Serial.print("Temperatura: ");
57 Serial.print(temperatura);
58 Serial.println(" °C");
59 Serial.print("Humedad: ");
60 Serial.print(humedad);
61 Serial.println(" %");
62 delay(2000);
63 }

```



## Ejercicio 15: Control de LEDs mediante comunicación serial

- Escribe un programa que reciba comandos a través del puerto serie para controlar los LEDs. Por ejemplo, enviar '1' podría encender led1, '2' apagar led2, etc.

<https://wokwi.com/projects/397519982591769601>

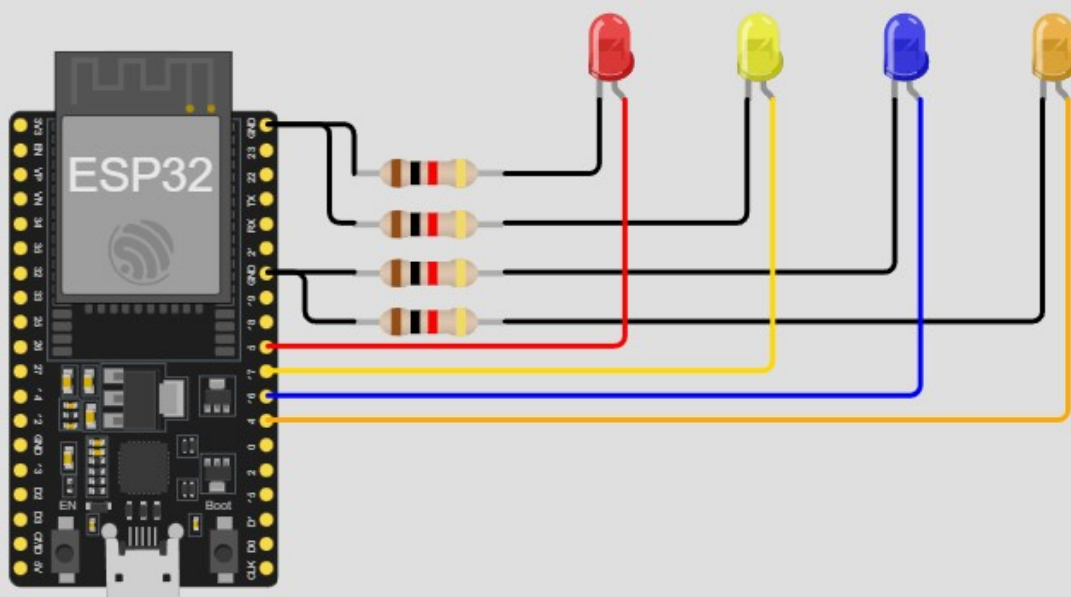
```
5  #include <Arduino.h>
6
7  #define LED1 5
8  #define LED2 17
9  #define LED3 16
10 #define LED4 4
11 char comando;
12
13 void setup() {
14     Serial.begin(115200);
15
16     pinMode(LED1, OUTPUT);
17     pinMode(LED2, OUTPUT);
18     pinMode(LED3, OUTPUT);
19     pinMode(LED4, OUTPUT);
20
21     digitalWrite(LED1, LOW);
22     digitalWrite(LED2, LOW);
23     digitalWrite(LED3, LOW);
24     digitalWrite(LED4, LOW);
25     Serial.println("Ingrese un número del 1 al 8 para controlar los LEDs:");
26     Serial.println("1: Encender LED1");
27     Serial.println("2: Apagar LED1");
28     Serial.println("3: Encender LED2");
29     Serial.println("4: Apagar LED2");
30     Serial.println("5: Encender LED3");
31     Serial.println("6: Apagar LED3");
32     Serial.println("7: Encender LED4");
33     Serial.println("8: Apagar LED4");
34 }
35
36 void loop() {
37     if (Serial.available() > 0) {
38         comando = Serial.read();
39         Serial.print(comando);
40         switch (comando) {
41             case '1':
42                 digitalWrite(LED1, HIGH);
43                 break;
44             case '2':
45                 digitalWrite(LED1, LOW);
46                 break;
47             case '3':
48                 digitalWrite(LED2, HIGH);
```



```

36 void loop() {
37   if (Serial.available() > 0) {
38     comando = Serial.read();
39     Serial.print(comando);
40     switch (comando) {
41       case '1':
42         digitalWrite(LED1, HIGH);
43         break;
44       case '2':
45         digitalWrite(LED1, LOW);
46         break;
47       case '3':
48         digitalWrite(LED2, HIGH);
49         break;
50       case '4':
51         digitalWrite(LED2, LOW);
52         break;
53       case '5':
54         digitalWrite(LED3, HIGH);
55         break;
56       case '6':
57         digitalWrite(LED3, LOW);
58         break;
59       case '7':
60         digitalWrite(LED4, HIGH);
61         break;
62       case '8':
63         digitalWrite(LED4, LOW);
64         break;
65     }
66   }
67 }

```



## Ejercicio 16: Secuencia de luces de emergencia

- Simula luces de emergencia con los LEDs, donde led1 y led2 parpadean alternativamente en un patrón rápido, mientras que led3 y led4 lo hacen en un patrón más lento.

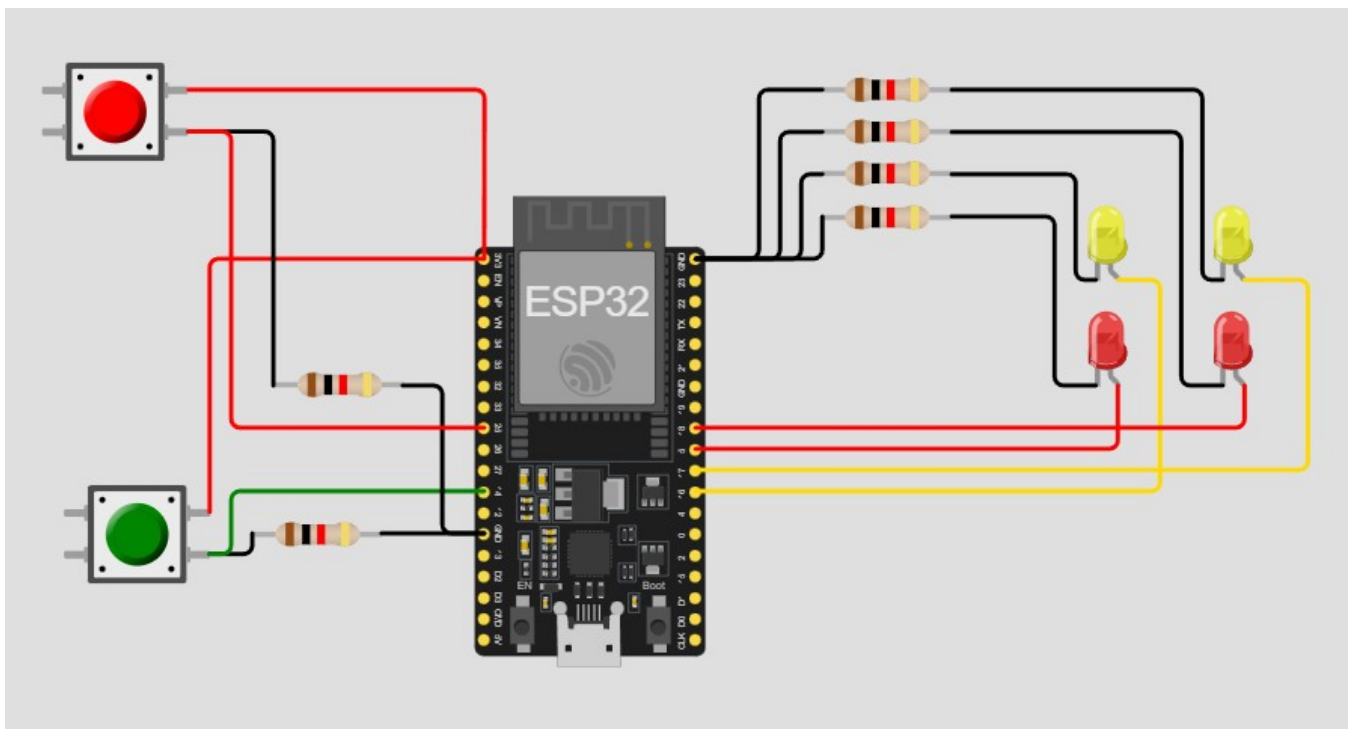
<https://wokwi.com/projects/397611720885951489>

```
5  #include <Arduino.h>
6
7  #define LED1 16
8  #define LED2 17
9  #define LED3 5
10 #define LED4 18
11
12 #define btn1 25
13 #define btn2 14
14
15 void setup() {
16     Serial.begin(115200);
17
18     pinMode(LED1, OUTPUT);
19     pinMode(LED2, OUTPUT);
20     pinMode(LED3, OUTPUT);
21     pinMode(LED4, OUTPUT);
22
23     pinMode(btn1, INPUT);
24     pinMode(btn2, INPUT);
25     Serial.println("Mantenga presionado el boton ROJO en caso de emergencias.");
26     Serial.println("BOTON VERDE terminar con llamado de emergencia.");
27 }
28
29 void loop() {
30     if(digitalRead(btn1) == HIGH) {
31         while (digitalRead(btn1) == HIGH) {
32             digitalWrite(LED1, HIGH);
33             digitalWrite(LED2, LOW);
34             digitalWrite(LED3, HIGH);
35             digitalWrite(LED4, LOW);
36             delay(100);
37             digitalWrite(LED1, LOW);
38             digitalWrite(LED2, HIGH);
39             digitalWrite(LED3, LOW);
40             digitalWrite(LED4, HIGH);
41             delay(300);
42             Serial.println(" EMERGENCIA !! ");
43         }
44     }
```

```

29 void loop() {
30     if(digitalRead(btn1) == HIGH) {
31         while (digitalRead(btn1) == HIGH) {
32             digitalWrite(LED1, HIGH);
33             digitalWrite(LED2, LOW);
34             digitalWrite(LED3, HIGH);
35             digitalWrite(LED4, LOW);
36             delay(100);
37             digitalWrite(LED1, LOW);
38             digitalWrite(LED2, HIGH);
39             digitalWrite(LED3, LOW);
40             digitalWrite(LED4, HIGH);
41             delay(300);
42             Serial.println(" EMERGENCIA !! ");
43         }
44     }
45     else if(digitalRead(btn2) == HIGH) {
46         delay(500);
47         Serial.println(" LLAMADO CANCELADO... ");
48         digitalWrite(LED1, LOW);
49         digitalWrite(LED2, LOW);
50         digitalWrite(LED3, LOW);
51         digitalWrite(LED4, LOW);
52     }
53 }
54
55

```





## TECNICATURA SUPERIOR EN **Telecomunicaciones**

Profesor → Gonzalo Vera.

Alumno → Nicolás Barrionuevo.