

# PRACTICA 1 - Pol Raich i Víctor Pallás

---

## 1. BLINK

---

El objetivo de la práctica es producir el parpadeo periódico de un led. Se utilizará la salida serie para depurar el programa

Para ello utilizaremos el microcontrolador ESP32-WROOM-32

### 1.1 FUNCIONALIDAD DE LA PRACTICA

1. Iniciar pin de led como salida
2. Iniciar el terminal serie
3. bucle infinito
  - encender led
  - sacar por puerto serie mensaje *ON*
  - espera de 500 milisegundos
  - apagar led
  - sacar por puerto serie mensaje *OFF*
  - espera de 500 milisegundos

### 1.2 CÓDIGO

```
#include <Arduino.h>

int PIN_RED_LED = 22;
int PIN_BLUE_LED = 19;
int PIN_YELLOW_LED = 17;

void setup() {
  Serial.begin(115200);
  pinMode(PIN_RED_LED, OUTPUT);
  pinMode(PIN_BLUE_LED, OUTPUT);
  pinMode(PIN_YELLOW_LED, OUTPUT);
}

void loop() {

  digitalWrite(PIN_RED_LED, HIGH);
  Serial.println("LED RED ON");
  delay(500);

  digitalWrite(PIN_BLUE_LED, HIGH);
  Serial.println("LED BLUE ON");
```

```
    delay(500);

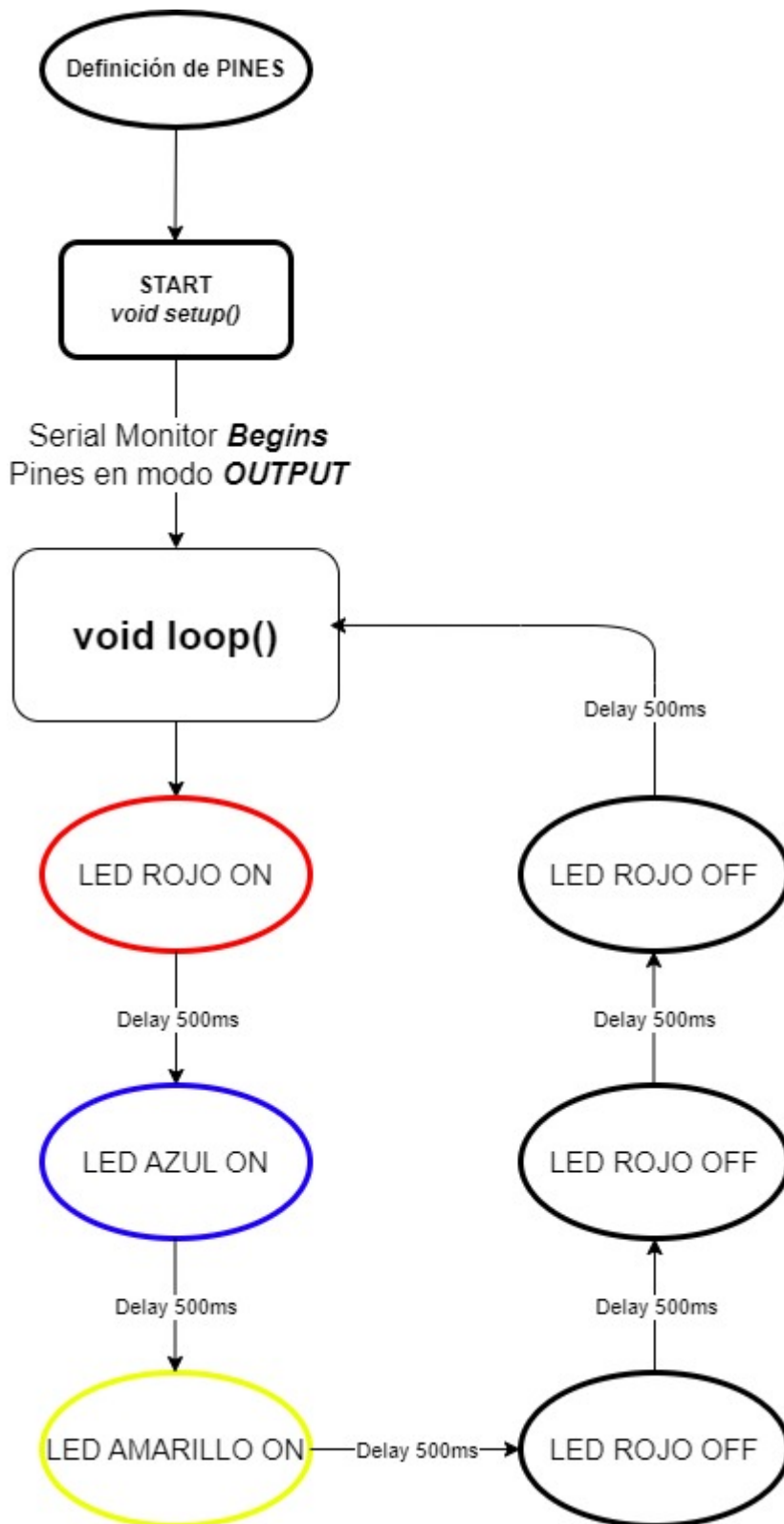
    digitalWrite(PIN_YELLOW_LED, HIGH);
    Serial.println("LED YELLOW ON");
    delay(500);

    digitalWrite(PIN_RED_LED, LOW);
    Serial.println("LED RED OFF");
    delay(500);

    digitalWrite(PIN_BLUE_LED, LOW);
    Serial.println("LED BLUE OFF");
    delay(500);

    digitalWrite(PIN_YELLOW_LED, LOW);
    Serial.println("LED YELLOW OFF");
    delay(500);
}
```

## 1.3 DIAGRAMA



## 1.4 PREGUNTA A RESPONDER

*Responder a la siguiente pregunta en el programa que se ha realizado cual es el tiempo libre que tiene el procesador ?*

*Si entendemos el tiempo libre del procesador como el tiempo en el que no ejecuta ninguna instrucción, debido al uso del comando delay() cada loop tenemos al procesador bloqueado 3 segundos en total...*

## 2. EJERCICIO VOLUNTARIO DE MEJORA DE NOTA

El sensor de temperatura interno de la ESP32 no funciona correctamente y los valores que podemos leer son siempre el mismo o varia muy poco y mal.

Hemos decidido a cambio utilizar un sensor de temperatura analógico modelo *KY-013*. El valor de temperatura medido lo mostraremos por la salida serie y además utilizaremos los conocimientos de la parte anterior de la práctica para encender unos leds de colores dependiendo de la temperatura que lea el sensor.

## 2.1 CÓDIGO

```
#include <Arduino.h>
#include <math.h>
//SENSOR DE TEMPERATURA ANALÓGICO KY-013

int sensorPin = 12; //PIN DEL SENSOR DE TEMP ANALÓGICO
int redLED = 27; //PIN LED ROJO
int yellLED = 26; //PIN LED AMARILLO
int blueLED = 25; //PIN LED AZUL
bool red = false, yell = false, blue = false;
//Función que transforma la lectura analógica a temperatura
double Thermistor(int RawADC)
{
    double Temp;
    //Conversión de la lectura RawADC a un valor de Vout
    Temp = log (((1024000/RawADC) - 10000));
    //Ecuación de Steinhart-Hart que nos convierte el Vout a Temperatura en °K
    //los coeficientes en el mismo orden son el A, B, C y se han precalculado para este sensor en
    Temp = 1 / (0.001129148 + (0.000234125 + (0.0000000876741 * Temp * Temp)) * Temp);
    //Conversión a Celsius
    Temp = Temp - 273.15;
    return Temp;
}

void setup()
{
    Serial.begin(115200);
    pinMode(sensorPin, INPUT);
    pinMode(redLED, OUTPUT);
    pinMode(yellLED, OUTPUT);
    pinMode(blueLED, OUTPUT);
    //necesario para que el rango sea entre 0-1024 y sea coherente con la fórmula
    analogReadResolution(10);
}

void loop()
{
    //Leemos la señal analógica en el PIN definido
    int readVal = analogRead(sensorPin);
    //Serial.println(readVal); //Chivato para ver la resolución del sensor
    //Llamada a la función para obtener la temperatura
    double temp = Thermistor(readVal);
```

```
//Output Temperatura en el Serial
Serial.print("Current temperature is:");
Serial.print(temp);
Serial.println("°C");
Serial.println("-----");
//Condiciones en las que se encienden los LEDs
if(temp < 21.5 )
{
    if(yell)
    {
        digitalWrite(yellLED, LOW);
        Serial.println("YELLOW LED OFF");
        yell = false;
    }
    if(!blue)
    {
        digitalWrite(blueLED, HIGH);
        Serial.println("BLUE LED ON");
        blue = true;
    }
}
else if (temp >= 21.5 && temp <= 28.0)
{
    if(blue)
    {
        digitalWrite(blueLED, LOW);
        Serial.println("BLUE LED OFF");
        blue = false;
    }
    if(red)
    {
        digitalWrite(redLED, LOW);
        Serial.println("RED LED OFF");
        red = false;
    }
    if(!yell)
    {
        digitalWrite(yellLED, HIGH);
        Serial.println("YELLOW LED ON");
        yell = true;
    }
}
else
{
    if(!red)
    {
        digitalWrite(redLED, HIGH);
        Serial.println("RED LED ON");
        red = true;
    }
    if(yell)
    {
        digitalWrite(yellLED, LOW);
```

```
        Serial.println("YELLOW LED OFF");  
        yell = false;  
    }  
}  
delay(500);  
}
```

## 2.2 VÍDEO DEL FUNCIONAMIENTO

[Link Vídeo](#)