

1. PROGRAMA DEL EQUIPO COMO ESPECTROFOTÓMETRO

// Pines para sensibilidad

#define S0 PE_5 //sensibilidad S0

#define S1 PB_4 //sensibilidad S1

// Pines para la escala

#define S2 PE_3 // Pin para la escala S2

#define S3 PE_4 // Pin para la escala S3

#define OUT PF_2 // Salida del sensor

//Pines de colores del LED RGB

#define Rj PD_2 //Pin LED rojo

#define Vd PE_1 //Pi LED verde

#define Az PD_3 //Pin LED azul

///Definiciones y variables necesarias para el programa

#define MainPeriod 100

long previousMillis = 0; // Almacenaremos el tiempo al final del ciclo.

volatile unsigned long duration=0; // acumulamos el ancho de pulso.

volatile unsigned int pulsecount=0;

volatile unsigned long previousMicros=0;

//Estas variables las vamos a usar para permitir calibrar eligiendo escala y sensibilidad

unsigned long nivel_escala;

unsigned long nivel_sensibilidad;

unsigned long sensibilidad;

unsigned long tmp;//temporal

unsigned long tiempo;//temporal

unsigned long time;//temporal

unsigned long timer;//temporal

int tecla;

//// Ahora las variables necesaria para la absorbancia

float frecuencia_rojo;

float frecuencia_verde;

float frecuencia_azul;

float absorbancia_rojo;

float absorbancia_verde;

float absorbancia_azul;

void setup() {

Serial.begin(9600);

// attach interrupt al pin OUT del sensor,

attachInterrupt(OUT, add_pulse, RISING);

pinMode(OUT, INPUT);

pinMode(S0, OUTPUT);

pinMode(S1, OUTPUT);

pinMode(S2, OUTPUT);

pinMode(S3, OUTPUT);

```
pinMode(Rj,OUTPUT);
pinMode(Az,OUTPUT);
pinMode(Vd,OUTPUT);
digitalWrite(Rj,LOW);
digitalWrite(Vd,LOW);
digitalWrite(Vd,LOW);

// La orden para limpiar el puerto serie es dependiente del terminal serie a usar.
//Putty no deja esta que es la generica:
// Serial.clear();
//En putty el comando para limpiar la pantalla es ESC[2J
Serial.write(27); //ESC comand
Serial.print("[2J");//clear screen comand
Serial.write(27); // Y que comience al principio de la pantalla
Serial.print("[H");
nivel_escala=ponerescala();
nivel_sensibilidad=ponersensibilidad();
}

//Inicializamos el puerto de serie
void loop() {
unsigned long currentMillis = millis();
if (currentMillis - previousMillis >= MainPeriod)
{
previousMillis = currentMillis;
// necesario para reducir errores
unsigned long _duration = duration;
unsigned long _pulsecount = pulsecount;
duration = 0; // reiniciando contadores
pulsecount = 0;
float Freq = 1e6 / float(_duration);
```

```
Freq *= _pulsecount; // calculando frecuencia

Serial.println(""); // Un separador

Serial.print("Frecuencia (Hz) ");

Serial.println(Freq);

//Ahora que me diga la absorbancia de cada color

Serial.println(frecuencia_ajustada);

normalizar(Freq);

delay(1000);

//// Aquí implementamos el algoritmo que se incluye al final del documento.

//// imprimimos por puerto serie los valores de la absorbancia de cada color.

Serial.print("La absorbancia del rojo es ");

Serial.println(absorbancia_rojo);

Serial.print("La absorbancia del verde es ");

Serial.println(absorbancia_verde);

Serial.print("La absorbancia del azul es ");

Serial.println(absorbancia_azul);

delay(1000);

}

}

///Leyendo los pulsos

void add_pulse() {

    unsigned long currentMicros = micros();

    duration += currentMicros - previousMicros;

    previousMicros = currentMicros;

    pulsecount++;

}

// Creamos un menú para poder seleccionar el nivel de escala que queramos.

int ponerescala() // Va de 0,1,2,3

{

    Serial.println("Introduzce el nivel de escala que desees : \n");
```

```
Serial.println(" \n");

Serial.println(" Si quieres escala 1 , pulsa 0 \n");

Serial.println(" Si quieres escala 2 , pulsa 1 \n");

Serial.println(" Si quieres escala 10 , pulsa 2 \n");

Serial.println(" Si quieres escala 100 , pulsa 3 \n");

time=millis();//15 seg arrancado

delay(10);//15+10

while (millis()-time < 10000 ) // Espera 10 segundos ,si no x defecto
{

    //Comienza la lectura del teclado dándole el valor vía puerto serie

    if (Serial.available() > 0) {

        // read the incoming byte:

        tecla = Serial.read();

        // ASCII '0' is 48, ASCII '1' is 49,ASCII '2' is 50,ASCII '3' is 51

        if (tecla == 48) {

            nivel_escala=1;

            digitalWrite(S2, LOW);

            digitalWrite(S3, LOW);

            Serial.println("Fija la escala 1x \n");

            timer=millis()-time;

            break;

        }

        if (tecla == 49) {

            nivel_escala=2;

            digitalWrite(S2, HIGH);

            digitalWrite(S3, LOW);

            Serial.println("Fija la escala 2x \n");
```

```
        timer=millis()-time;

        break;
    }

    if (tecla == 50) {

        nivel_escala=10;

        digitalWrite(S2, LOW);

        digitalWrite(S3, HIGH);

        Serial.println("Fija la escala 10x \n");

        timer=millis()-time;

        break;
    }

    if (tecla == 51) {

        nivel_escala=100;

        digitalWrite(S2, HIGH);

        digitalWrite(S3, HIGH);

        Serial.println("Fija la escala 100x \n");

        timer=millis()-time;

        break;
    }

    }

    } //El de while, terminó la espera :)

    timer=millis()-time;//25-15


    if(timer <= 35000){

        nivel_escala= 1 ;

    }

    Serial.print("Al no elegir nivel de escala queda la escala por defecto :");
```

```
Serial.print(nivel_escala);

Serial.println(" \n");

return(nivel_escala); //Devuelve el valor del nivel por si lo necesitamos

delay(1000);

}

// Creamos un menú para poder seleccionar el nivel de escala que queramos.

/////IMITAR AL ANTERIOR

int ponersensibilidad() { // Va de 0,1,2,3

    Serial.println("Introduzce el nivel de sensibilidad que desees : \n");

    Serial.println(" \n");

    Serial.println(" Si quieres Powerdown , pulsa 0");

    Serial.println(" Si quieres sensibilidad 1 , pulsa 1");

    Serial.println(" Si quieres sensibilidad 10 , pulsa 2");

    Serial.println(" Si quieres sensibilidad 100 , pulsa 3");

    Serial.println(" \n");

    time=millis(); // 15 seg arrancado

    delay(10);

    while (millis()-time < 10000) // Espera 10 segundos ,si no x defecto

    {

        //Comienza la lectura del teclado dandole el valor via puerto serie

        if (Serial.available() > 0) {

            // read the incoming byte:

            tecla = Serial.read();

            // ASCII '0' is 48, ASCII '1' is 49,ASCII '2' is 50,ASCII '3' is 51
```

```
if (tecla == 48) {  
    nivel_sensibilidad=0;  
    digitalWrite(S0, LOW);  
    digitalWrite(S1, LOW);  
    Serial.println("Pone sensibilidad en Powerdown \n");  
    timer=millis()-time;  
    break;  
  
}
```

```
if (tecla == 49) {  
    nivel_sensibilidad=1;  
    digitalWrite(S0, HIGH);  
    digitalWrite(S1, LOW);  
    Serial.println("Pone sensibilidad en 1 \n");  
    timer=millis()-time;  
    break;  
  
}
```

```
if (tecla == 50) {  
    nivel_sensibilidad=10;  
    digitalWrite(S0, LOW);  
    digitalWrite(S1, HIGH);  
    Serial.println("Pone sensibilidad en 10 \n");  
    timer=millis()-time;  
    break;  
  
}
```

```
if (tecla == 51) {  
    nivel_sensibilidad=100;  
    digitalWrite(S0, HIGH);
```



```
digitalWrite(S1, HIGH);

Serial.println("Pone sensibilidad en 100 \n");

timer=millis()-time;

break;

}

    }

}

//El de while,termino la espera :)

    if(millis() >= 10000){

        nivel_sensibilidad = 2 ; //Al pasar tiempo sin escribir,queda por defecto esta

    }

    Serial.print("Al no elegir nivel de sensibilidad queda la sensibilidad por defecto :");

    Serial.print(nivel_escala);

    Serial.println(" \n");

    return(nivel_escala); //Devuelve el valor del nivel por si lo necesitamos

    delay(1000);

}

// Ahora se normaliza la frecuencia para no tener que calibrar la misma cada vez que
// cambiemos de nivel de escala, de sensibilidad o ambos. Para ello necesitamos una nueva
// variable que llamaremos frecuencia_ajustada.

void normalizar(float frecuencia2){

    if(nivel_escala==100){

        frecuencia_ajustada=frecuencia2 * 100;

    }

}
```

```
if((nivel_escala==10 && nivel_sensibilidad==1) || (nivel_escala==100 &&
nivel_sensibilidad==10))
{
    frecuencia_ajustada=frecuencia2 * 10; // Los valores de frecuencia con escala 10 y
sensibilidad 1 coinciden con los valores escala 100 y sensibilidad 10
}

if(nivel_escala==2 && nivel_sensibilidad==1){
    frecuencia_ajustada=frecuencia2 * 2; //escala 2 y sensibilidad 1
}

if(nivel_escala==2 && nivel_sensibilidad==10){
    frecuencia_ajustada=frecuencia2 / 5; //escala 2 y sensibilidad 10
}

if(nivel_escala==2 && nivel_sensibilidad==100){
    frecuencia_ajustada=frecuencia2 / 50; //escala 2 y sensibilidad 100
}

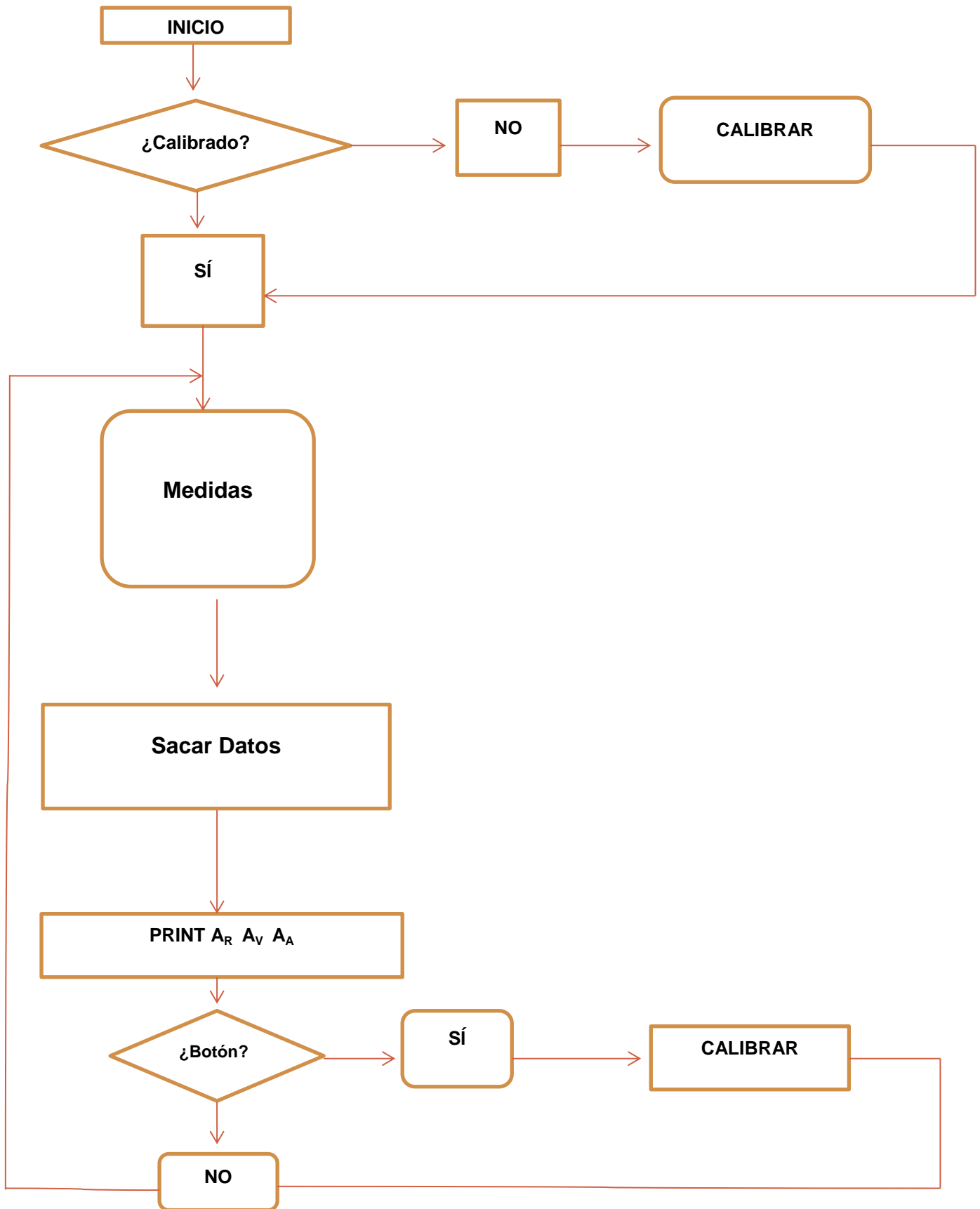
if((nivel_escala==1 && nivel_sensibilidad==1) ||(nivel_escala==10 && nivel_sensibilidad==10) ||
(nivel_escala==100 && nivel_sensibilidad==100) )
{
    frecuencia_ajustada=frecuencia2;      //// los valores de frecuencia con escala 1 y
sensibilidad 1 coinciden con los valores esc diez y sensibilidad 10 y con escala 100 y
sensibilidad 100
}

}
```

```
if((nivel_escala==1 && nivel_sensibilidad==100) )  
{  
    frecuencia_ajustada=frecuencia2 / 100;    /// Escala 1 y sensibilidad 100  
}  
  
if((nivel_escala==1 && nivel_sensibilidad==10) ||(nivel_escala==10 &&  
nivel_sensibilidad==100))  
{  
    frecuencia_ajustada=frecuencia2 / 10;    /// los valores de frecuencia con escala 1 y  
sensibilidad 10 coinciden con los valores esc diez y sensibilidad 100  
  
}  
  
if((nivel_escala==100 && nivel_sensibilidad==1))  
{  
    frecuencia_ajustada=frecuencia2 * 100;    /// los valores de frecuencia con escala 100 y  
sensibilidad 1  
}  
}
```

Con todo esto, ya tenemos nuestro programa prácticamente acabado, ya que sólo tenemos que implementar este sencillo algoritmo:

ANEXO III. ALGORITMO DEL ESPECTROFOTÓMETRO



Los bloques de medidas y de calibrar consisten básicamente en:

Medidas

Apagar LEDS
Medir Frecuencia F_0
Encender LED rojo
Medir Frecuencia F_R
Encender LED verde
Medir Frecuencia F_V
Encender LED azul
Medir Frecuencia F_A

Calibrar

Apagado F_0
Calibrar rojo: $R_{CAL} = (F_R - F_0)$
Calibrar verde: $V_{CAL} = (F_V - F_0)$
Calibrar azul: $A_{CAL} = (F_A - F_0)$
Calculamos las absorbancias

$$A_R = -\log \frac{F_R - F_0}{R_{cal}}$$

$$A_V = -\log \frac{F_V - F_0}{V_{cal}}$$

$$A_A = -\log \frac{F_A - F_0}{A_{cal}}$$

Implementado el algoritmo sólo nos queda sacar por puerto serie el valor de frecuencia_ajustada y de la absorbancia de cada color (rojo, verde y azul).

El proceso seguido en este paso se ha descrito detalladamente dentro del void loop ()