



UNIVERSIDAD
POLITÉCNICA
DE MADRID



Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingeniería y Diseño Industrial

Grado en Ingeniería Electrónica Industrial y Automática

SISTEMAS ELECTRÓNICOS DIGITALES

Casa domótica

Alejandro del Moral Lapresta	54753
Jacob Sánchez Moreira	54863
Wei Zheng	54911

Grupo:
A404

Profesor:
Pedro Luis Castedo Cepeda

Madrid, enero de 2022

ÍNDICE DE CONTENIDOS

1	Introducción	3
2	Realización del proyecto	3
2.1	Componentes utilizados.....	3
a)	Placa de desarrollo STM32F407G-DISC1	3
b)	Implementación del servomotor Datan S1213	4
c)	Implementación del módulo Bluetooth HC-06.....	4
d)	Implementación del sensor de ultrasonidos HC-SR04	4
e)	Implementación del sensor LDR	6
f)	Implementación del zumbador pasivo.....	6
g)	Implementación del sensor de temperatura	7
2.2	Montaje.....	8
2.3	Funcionamiento	10
a)	Control de la iluminación.....	10
b)	Control de la puerta	10
c)	Control de la temperatura	10
d)	Control de la alarma	11
e)	Aplicación móvil.....	11
3	Mejoras y problemas	12
3.1	Problemas	12
3.2	Soluciones a los problemas encontrados.....	13
3.3	Posibles mejoras	13
4	Anexo	14
4.1	Enlace al proyecto	14

ÍNDICE DE FIGURAS

FIGURA 1. Placa STM32F407G	3
FIGURA 2. Servomotor Datan S1213	4
FIGURA 3. Módulo Bluetooth HC-06	4
FIGURA 4. Sensor de ultrasonidos HC-SR04	6
FIGURA 5. Sensor LDR	6
FIGURA 6. Zumbador pasivo	7
FIGURA 7. Sensor de temperatura	8
FIGURA 8. Esquemático de las conexiones	9
FIGURA 9. Montaje de la maqueta	10
FIGURA 10. Captura de la aplicación móvil	11
FIGURA 11. Programación de la aplicación móvil	12

1 INTRODUCCIÓN

En este proyecto se construirá y programará una casa domótica utilizando la placa de desarrollo STM32F407G-DISC1. Utilizando la tecnología se podrán realizar funciones para aumentar el grado de comodidad de las personas.

La casa constará de un control manual y un control por Bluetooth de la iluminación, temperatura, alarma y apertura de la puerta de la casa.

2 REALIZACIÓN DEL PROYECTO

2.1 COMPONENTES UTILIZADOS

a) Placa de desarrollo STM32F407G-DISC1

La placa utilizada posee un procesador Cortex-M4 de 32 bits RISC, de arquitectura Harvard con pipeline de 3 estados (Fetch, decode y execute), lo que permite una ejecución eficiente del programa.

La documentación de la placa es muy extensa por lo que aquí se señalarán algunas de las características que han sido utilizadas para la realización de este trabajo.

- Frecuencia de reloj de hasta 168 MHz.
- Memoria Flash de 1 Mbyte.
- 3 conversores A/D de 12 bits.
- 2 conversores D/A de 12 bits.
- 12 temporizadores de 16 bits.
- 2 temporizadores de 32 bits.
- Hasta 140 puertos I/O con capacidad de interrupción.
- Comunicación I2C, USART, UART y CAN.



FIGURA 1. Placa STM32F407G

b) Implementación del servomotor Datan S1213

De la hoja de características del fabricante podemos sacar los siguientes datos:

- Voltaje: 4,8 V
- Velocidad: 0,21 s / 60°
- Par: 6,5 kg · cm
- Peso: 46 g



FIGURA 2. Servomotor Datan S1213

c) Implementación del módulo Bluetooth HC-06

Se ha decidido implementar una aplicación Bluetooth para poder elegir la función que se quiera ejecutar en la casa desde un dispositivo móvil.

- Protocolo Bluetooth: protocolo estándar Bluetooth V2.0
- Potencia: Class2 (+6 dBm)
- Banda: 2,40 GHz – 2,48 GHz, ISM Band
- Protocolo USB: USB v1.1/2.0
- Voltaje operativo: de +3,3 V a 6 V
- Rango de temperatura: de –20°C a +55°C
- Intensidad operativa: 40 mA

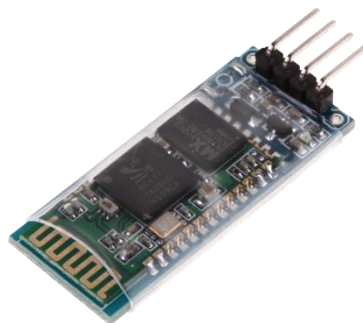


FIGURA 3. Módulo Bluetooth HC-06

d) Implementación del sensor de ultrasonidos HC-SR04

El sensor de ultrasonidos es un dispositivo electrónico usado para la detección de objetos y medición de distancias. Su funcionamiento se basa en el envío de ondas de

alta frecuencia (40 kHz) que chocan con un objeto y se reflejan volviendo de nuevo al sensor.

En la hoja de especificaciones del fabricante nos encontramos con información detallada para el conexionado y parámetros característicos. Los datos más relevantes son los siguientes:

- Voltaje operativo: 5 V
- Intensidad operativa: 15 mA
- Frecuencia de trabajo máxima: 40 Hz
- Distancia de detección mínima: 2 cm
- Distancia de detección máxima: 400 cm
- Ángulo máximo de medición: 30°
- Pulso de señal de entrada (*Trigger*): 10 μs
- Pulso de señal de salida (*Echo*): Proporcional a la distancia, Máx de 23,3 ms

El envío de la señal ultrasónica es producido por el pin *Trigger* al ser excitado por un pulso de 10 μs. Es suficiente con la señal de 3,3 V que provee la placa para activarse.

La recepción de la señal es captada por el pin *Echo*, produciendo un pulso de duración idéntica al tiempo transcurrido entre el envío y la recepción de esa onda. Debido a que el voltaje de trabajo del sensor (5 V) y de la placa (3,3 V) son distintos, es necesario adaptar la amplitud de la señal a un rango que pueda leer la placa. Para ello se ha optado por implementar un divisor de tensión con dos resistencias (1 kΩ y 2 kΩ) para ajustar los valores de 5 a 3,3 V.

La distancia es fácilmente calculable sabiendo que la onda viaja a la velocidad del sonido, 340 m/s, y conociendo el ancho de pulso que nos proporciona el sensor. Sin embargo, el tiempo que nos proporciona el sensor es el transcurrido entre la ida y la vuelta, por lo que habría que dividir ese tiempo entre 2.

$$Distancia = \frac{Tiempo_{Ida-vuelta}}{2} \cdot V_{sonido}$$

Para la obtención del tiempo a través de la lectura de la señal de salida del pin *Echo* se ha optado por el uso de un temporizador en modo de captura de señal. Este modo permite obtener el tiempo en el que se ha producido un flanco, ya sea de subida o de bajada.

La detección de los flancos es capturada con una interrupción del temporizador.

Sabiendo que se va a producir un pulso de duración determinada, es suficiente con detectar un flanco de subida, cambiar el modo de detección a flancos de bajada, y detectar un flanco de bajada. Con esa diferencia de tiempos se puede proceder al cálculo de la distancia.

No es un sensor muy preciso, se obtienen buenas aproximaciones de la distancia con errores de ± 0.5 cm. Suficiente para el propósito de nuestro trabajo.



FIGURA 4. Sensor de ultrasonidos HC-SR04

e) Implementación del sensor LDR

Se trata de un resistor que varía su valor de resistencia eléctrica dependiendo de la cantidad de luz que incide sobre él.



FIGURA 5. Sensor LDR

f) Implementación del zumbador pasivo

Se usa para generar un “chirrido” electromagnético utilizado para generar señales de sonido de diferentes frecuencias.

El principio de funcionamiento del zumbador pasivo es utilizar un pulso PWM que hace vibrar el aire para generar un sonido. Por ejemplo, enviando un pulso de 523 Hz, genera una nota Do, con 587 Hz un Re, y con 659 Hz un Mi.

Algunas de sus características son:

- Voltaje: de +3 V a 5 V
- Potencia: $\geq 85 \text{ dB}$
- Rango de frecuencia: de 50 Hz a 14 kHz
- Frecuencia de resonancia: $\sim 2300 \text{ Hz}$
- Rango de temperatura: de -20°C a 60°C
- Peso: 1,27 g



FIGURA 6. Zumbador pasivo

g) Implementación del sensor de temperatura

El sensor de temperatura externo consta de un termistor NTC, una resistencia dependiente de la temperatura.

El sensor posee dos salidas, una digital, ajustable mediante un potenciómetro que se activa una vez se ha superado cierto valor; y otra analógica, que provee el valor de la caída de potencial en la resistencia.

El circuito consta de un divisor de tensión compuesto por una resistencia fija de 10 kΩ y el termistor. La lectura será la diferencia de potencial entre el punto común de resistencia-termistor y tierra.

Debido a que el termistor no es lineal es necesario realizar ciertos cálculos para obtener el valor de la temperatura.

El voltaje medido será:

$$V_o = \frac{R_T}{R_T + R} \cdot V_{cc}$$

Como el conversor que tiene el sensor es de 10 bits, el máximo valor que se leerá será de 1023. Por tanto, el valor leído del sensor será de:

$$ADC = \frac{V_o}{V_{cc}} \cdot 1023$$

Sustituyendo en la primera ecuación se obtiene:

$$ADC = \frac{\frac{R_T}{R_T + R} \cdot V_{cc}}{V_{cc}} \cdot 1023 = \frac{R_T}{R_T + R} \cdot 1023$$

Despejando la incógnita del valor de la resistencia del termistor:

$$R_T = \frac{R}{\frac{1023}{ADC} - 1}$$

Se obtiene una aproximación de la temperatura a través de la fórmula del NTC.

$$\frac{1}{T} = \frac{1}{T_0} + \frac{1}{B} \ln\left(\frac{R_T}{R_0}\right)$$

Por lo que, relacionando estas dos ecuaciones se llega a:

$$T = \left(\frac{1}{T_0} + \frac{1}{B} \ln \left(\frac{R \cdot R_0}{\frac{1023}{ADC} - 1} \right) \right)^{-1}$$

De la hoja de características que proporciona el fabricante del módulo se pueden obtener algunos de los valores necesarios para la obtención de la temperatura.

- $T_0 = 298.15 \text{ K (25}^\circ\text{C)}$
- $B = 3950 \text{ K}$
- $R_0 = 10 \text{ k}\Omega$
- $R = 10 \text{ k}\Omega$
- Rango de temperatura: -55°C a 125°C



FIGURA 7. Sensor de temperatura

2.2 MONTAJE

Para la implementación de todo lo descrito anteriormente, se usará una placa de conexionado rápido (*Protoboard*).

Los pulsadores se han montado con una resistencia de $10 \text{ k}\Omega$, de forma que sea una resistencia *pulldown*, es decir, que cuando se pulse el pulsador tendremos un valor alto de voltaje.

El sensor LDR también se ha armado con una resistencia de $1 \text{ k}\Omega$ para evitar dañar el componente, al igual que los leds.

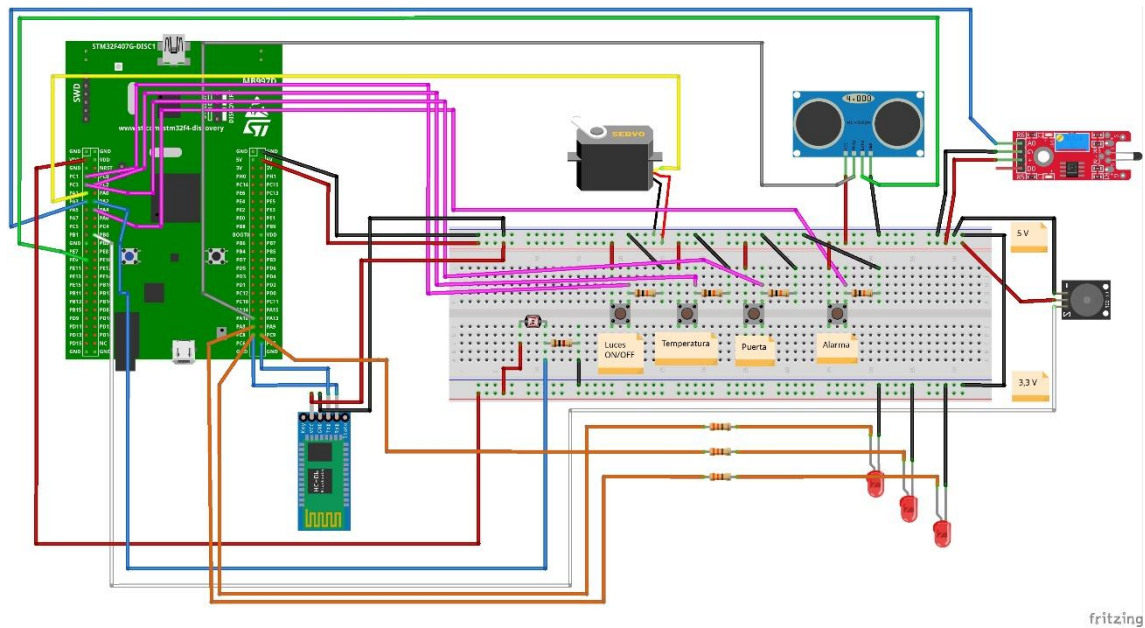
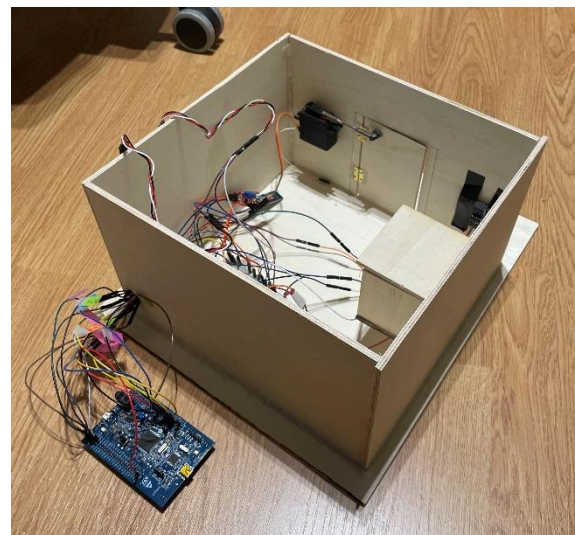
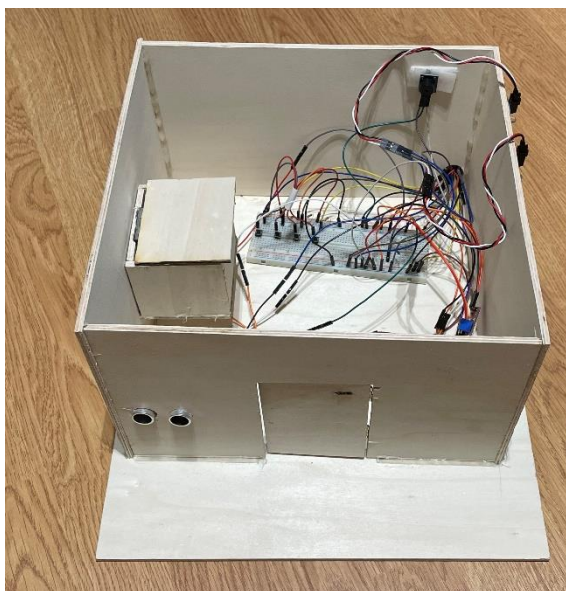


FIGURA 8. Esquemático de las conexiones

Todo esto ha sido montado en una maqueta que simula una casa. La maqueta se ha hecho con madera y se han usado bisagras para la puerta.

Además, para simular un ambiente poco luminoso, se ha escondido el sensor LDR en una cajita que se puede abrir y cerrar.

Este es el resultado:



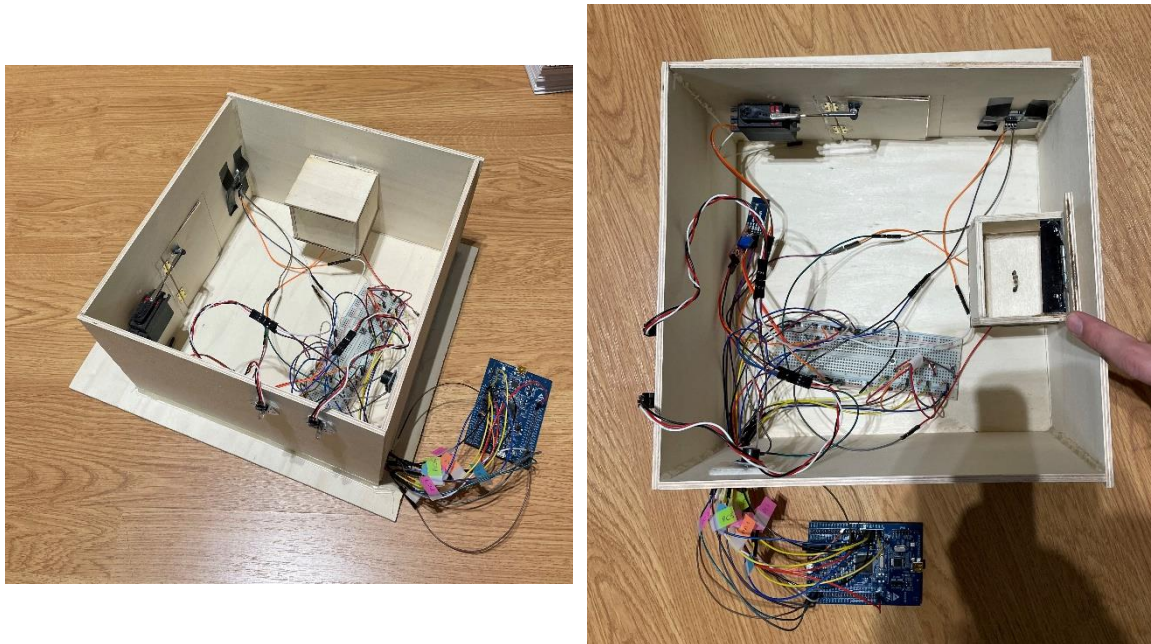


FIGURA 9. Montaje de la maqueta

2.3 FUNCIONAMIENTO

Las funciones que se han implementado en este proyecto son las siguientes:

a) Control de la iluminación

Se controlará un led a través del pin PC8 en función de la intensidad de luz que incida en la fotorresistencia (LDR). Es decir, si el valor que capta el sensor está por debajo de un cierto límite se encenderá el led, y si está por encima se apaga.

Además del sensor, se puede encender y apagar mediante un pulsador externo a la placa conectado al pin PC1 o mediante la aplicación móvil.

b) Control de la puerta

Esta funcionalidad permite abrir la puerta de entrada gracias al control de un servomotor, cuyo pin de señal se conecta al pin PA1.

Para abrir o cerrar, se usará un pulsador conectado al pin PC3 o mediante la aplicación móvil.

Una vez abierta la puerta, permanecerá en ese estado durante 5 s que, transcurrido ese tiempo, se cerrará automáticamente.

c) Control de la temperatura

Mediante la interacción de un pulsador conectado al pin PC2 o con la aplicación móvil, se puede medir la temperatura durante 10 s, que se mostrará por pantalla gracias al

Live Expressions del programa STM32CubeIDE. Transcurrido el tiempo, el sensor deja de medir.

Además, se encenderá un led conectado al pin PC9 que indica que se está midiendo.

d) Control de la alarma

Por último, se ha implementado una alarma (zumbador pasivo conectado al pin PB0) que suene cuando hay una distancia menor de 20 cm al sensor ultrasonidos.

El pin ECHO se conecta al pin PE9 y el TRIG al pin PA10.

La alarma se desconectará cuando pasen 5 s, se pulse un pulsador conectado al pin PA4 o se use la aplicación móvil.

También se encenderá un led conectado al pin PA8 cuando suene la alarma.

e) Aplicación móvil

La aplicación se ha desarrollado con la herramienta MIT App Inventor, una sencilla aplicación donde la programación se hace mediante bloques y la interfaz de forma gráfica, muy intuitiva.

Al pulsar un botón, el móvil envía un carácter mediante Bluetooth al módulo Bluetooth insertado en el proyecto (módulo HC-06).

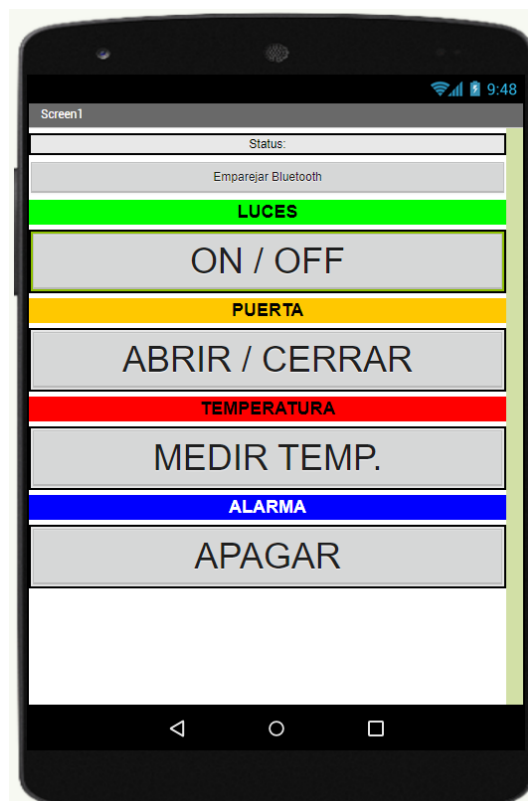


FIGURA 10. Captura de la aplicación móvil

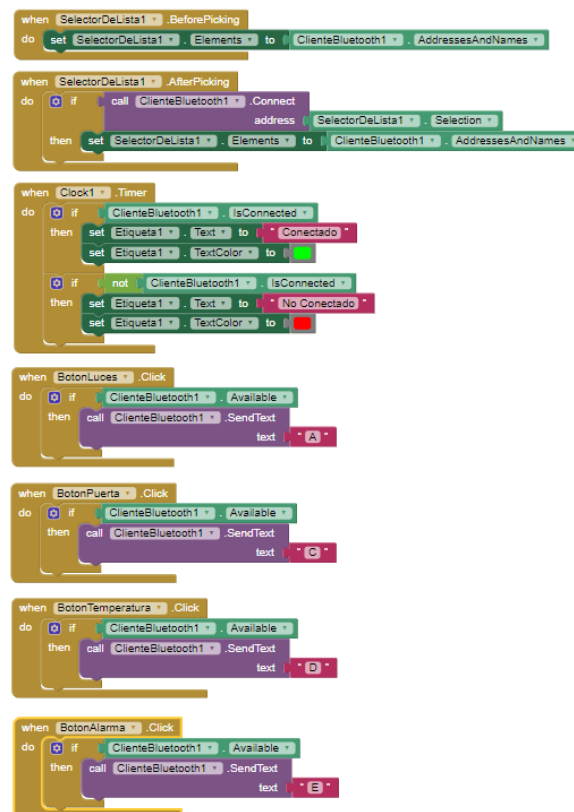


FIGURA 11. Programación de la aplicación móvil

3 MEJORAS Y PROBLEMAS

3.1 PROBLEMAS

Algunos de los problemas que han surgido durante el desarrollo del proyecto han sido:

1. Al querer poner tantos componentes, no hemos podido probarlos todos a la vez de primeras.
2. Lo que más problemas dio fue la programación de las funcionalidades del control de la iluminación, de la puerta y de la temperatura, debido a que tienen varios estados diferentes.
3. Al realizar la maqueta, hubo problemas con el servo que se montó inicialmente (el SG90) debido a que no era capaz de tirar de la puerta.
4. Los cables se desconectan fácilmente de los pines de la placa dificultando el correcto funcionamiento y la fluidez del mismo.
5. El módulo Bluetooth usado al principio no se emparejaba con ninguno de nuestros teléfonos móviles para poder usar la aplicación.

6. Al montar todo en la maqueta hemos utilizado muchos cables, lo que nos confundía bastante y nos ralentizaba a la hora de conectarlos en la placa.

3.2 SOLUCIONES A LOS PROBLEMAS ENCONTRADOS

Soluciones adoptadas a los problemas enunciados anteriormente:

1. Se ha necesitado ir probando componente a componente por separado, para entender su funcionamiento y saber utilizarlo.
2. Se solucionó a base de programar, probar y cambiar. Es lo que más tiempo ha llevado.
3. Se cambió al servo que usamos actualmente (el Datan S1213) ya que ofrece un par mayor, siendo capaz de tirar de la puerta con solvencia.
4. La única solución es tener precaución y estar revisando continuamente las conexiones.
5. Se tuvo que cambiar el módulo por otro que sí empareja correctamente.
6. Se han etiquetado todos los cables para acelerar el proceso de conexionado con la placa.

3.3 POSIBLES MEJORAS

Algunas mejoras que es posible implementar para mejorar el funcionamiento en el futuro:

- Sustituir algunos módulos de sensores por otros que tengan mayor precisión en las medidas, como es el caso del sensor de temperatura implementado con el termistor, únicamente tiene un conversor de 10 bits frente al de la placa de 12 bits. Aumentando el número de bits mejoraría la lectura de la temperatura.
- Añadir un módulo LCD que permita observar los valores de los sensores, simplificaría su uso, ya que para observar algunos valores como la temperatura hemos de acudir al Live Expressions del programa STM32CubeIDE.
- La placa utilizada siempre está conectada mediante un cable USB, esto produce cierta dependencia, ya que si no se dispone de ese conector no funcionaría. Una solución sería conectarlo a unas baterías.
- También se pueden plantear añadir más funcionalidades, como por ejemplo controlar persianas con un motor de continua, poner riego automático, etc.

4 ANEXO

4.1 ENLACE AL PROYECTO

A través del siguiente enlace se podrá acceder a todo el código desarrollado para el funcionamiento de este proyecto:

<https://github.com/alejandrodelmoral/SED-Trabajo-Micros>