

Trabajo Practico 2

FreeRTOS

Martín Cogo Belver

19 de marzo de 2024

1. Detalles de implementación

1. Comunicación Serial

- a) *USBport*: Esta clase es una interfaz que provee los métodos para enviar y leer los datos del bus serie.
- b) *Protocolo de comunicación*:
 - El servidor envía tramas con la forma “*num*”.
 - **num**: si el numero es 1, se encarga de apagar o prender las mediciones del sensor de luz.
 - El Arduino responde con tramas con la forma “*num;alarm*”.
 - **num**: valor 0 o 1024 que indica la luz captada por el sensor sensor.
 - **alarm**: un carácter que indica si el estado de la medición es menor a 800 con símbolo “-” o mayor a 800 con símbolo “!”.

2. Comunicación Cliente-Servidor

- a) El socket se crea al conectarse a </tp2/dashboard>
- b) El script *dashboard.js* en el **cliente** se encarga de detectar eventos y enviar los parámetros desde la vista hasta el servidor utilizando la librería socketIO con javascript.

3. Threads

- a) *ReadThread*: Este hilo se encarga de leer el contenido del bus utilizando la clase **USBport** y almacenándolas tramas en una cola.
- b) *SendThread*: Este hilo se encarga de tomar el contenido de la cola llenada por el **ReadThread**.

4. Tareas de FreeRTOS

- *MesureLightTask*: Tarea que se encarga de leer los valores del sensor LDR.
- *SendSerialTask*: Tarea que se encarga de enviar las mediciones por el bus serial.
- *AlarmTask*: Tarea que se encarga de activar la tarea *BlinkingTask*.
- *ReadSerialTask*: Tarea que se encarga de la lectura del bus serie.
- *BlinkingTask*: Tarea que se encarga de hacer titilar el led cuando el medidor se encuentra activado.
- *AlarmBlinkingTask*: Tarea que se encarga de hacer titilar el led en el pin 12 cuando existe una alarma.

2. Librerías utilizadas

1. Flask framework: Para la creación de la página web.
2. pySerial: Para la comunicación con puerto Serial.
3. socketio: Para la comunicación con websockets.

3. Diagrama

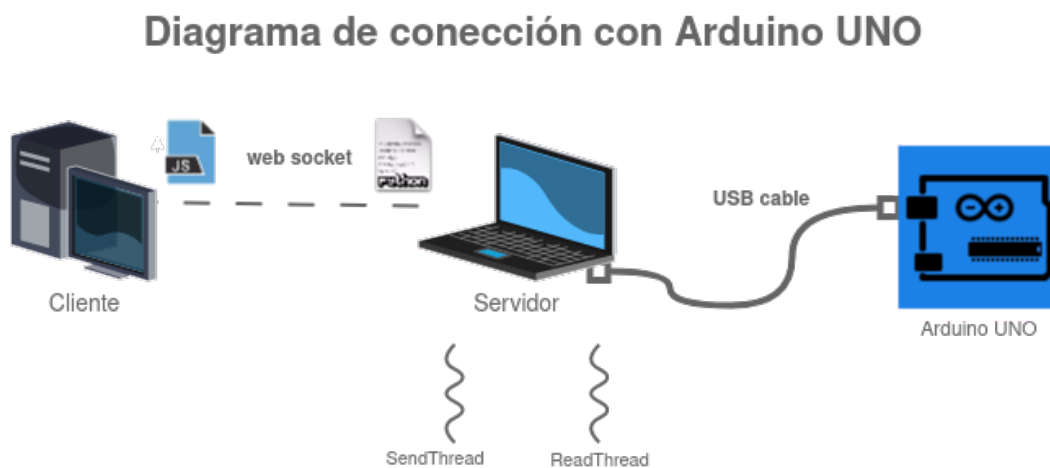


Figura 1: Diagrama representativo y los hilos del programa

4. Página web

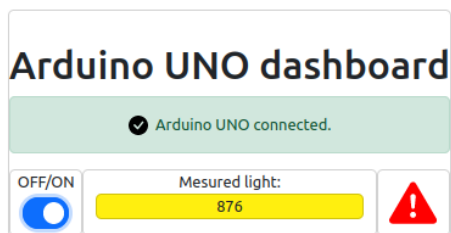


Figura 2: Página web diseñada para controlar el Arduino UNO.