

Trabajo Practico 1

Conversores AD y DA.

Protocolos de comunicación serial cableados

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 “*a;b;c;d*” .
 - **a**: valor 0 o 1 para encender led del pin 13.
 - **b**: valor 0 o 255 para encender led del pin 9.
 - **c**: valor 0 o 255 para encender led del pin 10.
 - **d**: valor 0 o 255 para encender led del pin 11.
 - El Arduino responde con tramas con la forma “*numero*”. Donde numero es la luz que ha medido el dispositivo.

2. Comunicación Cliente-Servidor

- a) El socket se crea al conectarse a </tp1/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**.

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

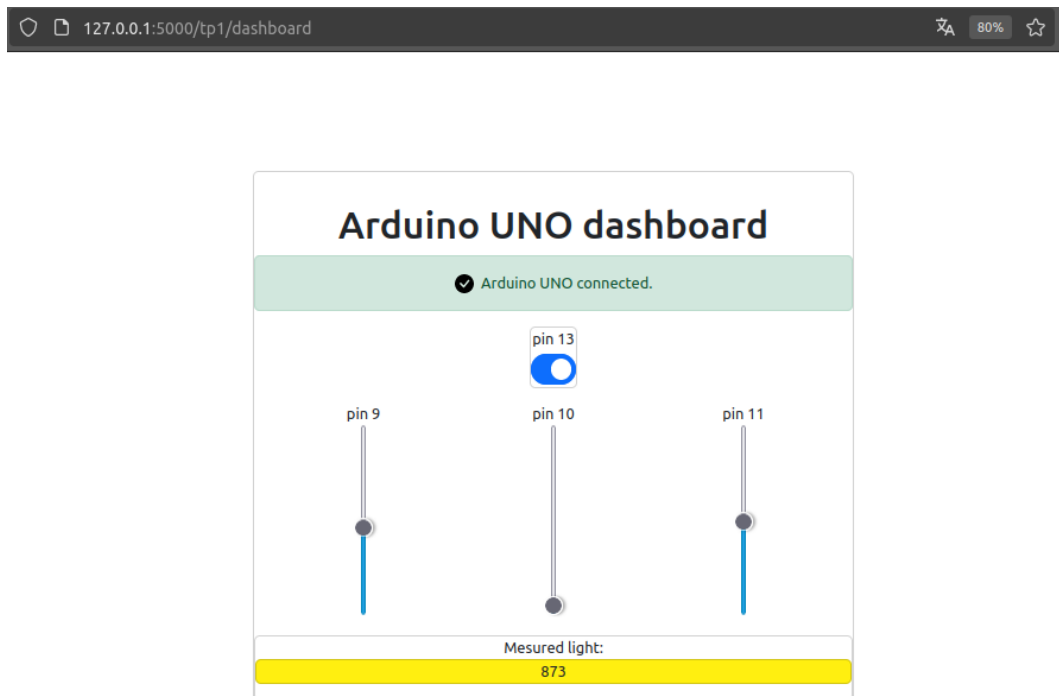


Figura 2: Página web diseñada para controlar el Arduino UNO.