

clase-08

martes 03 mayo 2022, presencial

repaso clase anterior y programa hoy (15 min)

la clase pasada aprendimos:

- sonido
- diseño de sintetizadores
- protocolo MIDI

hoy aprenderemos:

- comunicación serial con Processing
- comunicación serial con p5.js

comunicación serial con Processing (60 mins)

gracias a la comunicación serial, podemos crear aplicaciones en Processing que son capaces de recibir información desde nuestro Arduino, y al revés también.

abramos el ejemplo [ej_00_arduino_envia_numeros](#) y subamos el código al microcontrolador Arduino.

este ejemplo envía por puerto serial una línea, con dos valores separados por coma. el primer valor es aleatorio y el segundo es un contador, que parte en 0 y aumenta en 1 por cada iteración del bucle `loop()` de Arduino.

podemos abrir el monitor serial del software Arduino IDE y ver los valores para confirmar que está funcionando.

cerramos el monitor serial y luego abrimos el ejemplo de Processing [ej_00_processing_recibe_numeros](#) que recibe estos números, porque una comunicación serial solamente puede ir desde un lugar a otro, entonces si no cerramos el monitor serial de Arduino, el puerto serial estará ocupado y Processing no podrá acceder a esos valores.

el ejemplo en Processing dibuja una elipse usando los valores recibidos desde Arduino. para lograr hacer la conexión serial entre nuestro hardware microcontrolador Arduino y el software Processing, debemos correr el software en Processing y mirar la consola de Processing e identificar el número de puerto serial ocupado por Arduino, y actualizar ese valor en el código de Processing.

notar que esta configuración puede ser un problema, porque esto hace que nuestro software en Processing no sea "plug and play" o "conectar y listo", debemos configurar esto en cada computador o Arduino que usamos, para asegurar que la conexión serial funcione.

motivación de comunicación serial con Processing

nuestro computador tiene distintas maneras de ingresar información, como ratón, teclado y cámara.

la gran gracia de nuestro microcontrolador Arduino es que tiene pines GPIO (general purpose input and output, en español pines de propósito general para entrada y salida).

comunicación serial con p5.js (60 mins)