

## Tutorial15. Comunicación serie

### Contenido

Contenido teórico .....	3
Puertos serie de Arduino.....	3
Funciones Arduino .....	3
Proyecto15_01. Control de un LED desde el teclado de un PC.....	4
Enunciado .....	4
Código.....	4
Ejercicios.....	5

## Material necesario

- Placa Arduino
- Cable micro USB
- Componentes:
  - LEDs
  - Resistencias
  - Pulsadores
  - Potenciómetro
  - Protoboard
  - Cables

### **IMPORTANTE:**

Comprobaremos que nuestra placa **Arduino** está **desconectada** y sin energía, puesto de no ser así podría dañarse tanto la placa, como el equipo. Una vez hemos realizado esta comprobación, pasaremos a realizar el montaje.

## Contenido teórico

### Puertos serie de Arduino

Todos los microcontroladores de las placas Arduino tienen al menos un puerto serie (también conocido como UART o USART). El puerto serie permite la comunicación entre la placa Arduino y un ordenador u otros dispositivos (Raspberry, p.e.).

El microcontrolador de **Arduino Leonardo** dispone de **dos puertos serie**:

- “Serial”: correspondiente al puerto serie USB.
- “Serial1”: correspondiente al puerto serie localizado en los pines 0 (RX1) y 1 (TX1).
- 

El microcontrolador de **Arduino Uno** solo dispone de **un puerto serie**:

- “Serial”: correspondiente al puerto serie USB y también a los pines 0 (RX0) y 1 (TX1).

En el apartado referencia de la web de Arduino puedes ver que existen varias funciones para trabajar con el puerto serie.

El entorno de trabajo de Arduino integra un “Monitor Serial”, disponible en las herramientas, mediante el cual se monitorizan los valores recibidos por el puerto serie. Es muy importante que la velocidad de transmisión establecida en el “Monitor Serial” coincida con la configurada para el puerto serie de Arduino. Por defecto, este valor es de 9.600 baudios para el “Monitor Serial”.

### Funciones Arduino

<http://arduino.cc/en/Reference/Serial>

## Proyecto15\_01. Control de un LED desde el teclado de un PC

### Enunciado

Realizar un código que cuando escribamos la letra “a” en nuestro PC (consola serial) actúe del siguiente modo, la primera vez que se escriba un LED se encenderá, la segunda vez, el LED se apagará.

### Código

```
//VARIABLES & CONSTANTS//
//-----//
const int led = 11;
char letter; //Variable to store the read key
boolean StateLed=false; //It warns about the LD state

//-----//
void setup()
{
    Serial.begin(9600);

    pinMode(led, OUTPUT);
}
//-----//

void loop()
{
    // It checks when a byte is received

    if (Serial.available()>0) {
        letter=Serial.read();

        // It checks what key is pressed and what is
        // the LED state

        if ((letter=='a') && (StateLed==false))
        {
            digitalWrite(led,HIGH);
            StateLed=true; // Update the LED state
        }

        else if ((letter=='a') && (StateLed==true))
        {
            digitalWrite(led,LOW);
            StateLed=false; // Update the LED state
        }
    }
}
```

## Ejercicios

1. Mostrar por la consola serial un número entre 0 y 10, este número es proporcionado por un potenciómetro conectado a la entrada analógica A0.  
(Sugerencia: usar la función `map()`, (<http://arduino.cc/es/Reference/map>))
2. *Montar un circuito con un LED. En esta ocasión, se trata de usar el puerto serie para recibir información que actúe sobre el diodo LED. Si pulsamos la tecla "Y" en el teclado el LED se encenderá, si pulsamos "N" se apagará. Para el resto de casos el LED mantiene su estado anterior.*

*NOTA: Usar un switch-case en el código.*