



Introducción a Arduino

TIC I

4. Pulsador actuando sobre un LED

4.1 Objetivo

Construir un circuito con un pulsador que ha de controlar el funcionamiento de otro circuito con un LED.

Cuando el pulsador esté activado el LED debe lucir. Al liberar el pulsador el LED debe de apagarse y permanecer en ese estado hasta que se actúe de nuevo sobre el pulsador.

4.2 Material

- ✗ 1 placa Arduino
- ✗ 1 placa protoboard
- ✗ 1 LED
- ✗ 1 botón
- ✗ 1 Resistencia 1k Ω
- ✗ 1 Resistencia 10 k Ω
- ✗ 5 cables

4.3 Resistencias Pull-Down y Pull-Up

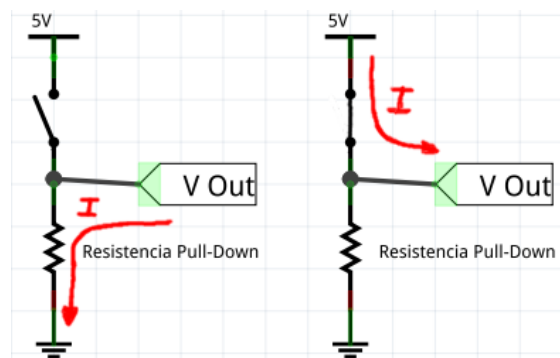
En el proyecto tenemos componentes que funcionan en dos estados de tensión, HIGH o LOW. Es posible que debido a ruido eléctrico o variaciones en la fuente de alimentación el valor caiga a un rango indefinido y sea imposible determinar si estamos HIGH o LOW. Para solucionar esto se usan las resistencias Pull-Down y Pull-Up.

Esa tensión residual podría ser tomada por Arduino como señal y dar lectura cuando en realidad no esté activado el pulsador. La idea es enviar esa corriente residual a tierra cuando nos pueda molestar.

4.3.1 Resistencias Pull-Down

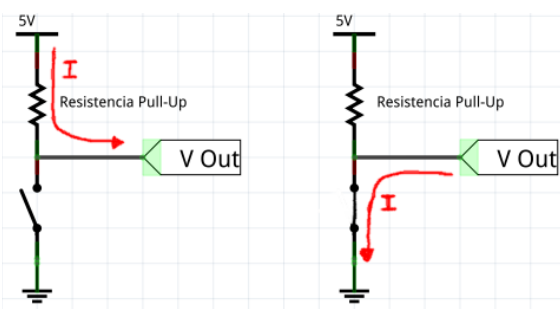
La resistencia se conecta a tierra, así cuando el interruptor este abierto la corriente residual se dirigida hacia la resistencia dejando un valor 0 en Vout y si el interruptor esta cerrado la corriente se moverá hacia Vout dejando un valor lógico HIGH.

Utilizaremos este sistema cuando queramos enviar señal a Arduino al actuar sobre el pulsador.



4.3.2 Resistencias Pull-Up

Cuando el interruptor esta abierto la corriente va desde la fuente de alimentación al Vout dando un valor lógico HIGH y cuando el interruptor esta cerrado la corriente se mueve hacia tierra dejando un 0 en Vout.



Utilizaremos este sistema cuando queramos enviar señal a Arduino cuando el pulsador esté abierto.

4.3.3 Valor de la resistencia

Se suelen usar resistencias de $10\text{k}\Omega$, valor suficientemente alto para no influir en el circuito..

4.4 Elementos de código necesarios

- #define
- digitalWrite(pin, estado) estado (HIGH, LOW) (1,0)
- digitalRead(pin) posibles lecturas (HIGH, LOW) (1,0)
- if(condición){
 bloque_ordenes;
 }else{
 bloque_ordenes II
 }

4.5 Conexiones

- ✗ Establezco el pin 8 como salida para poder mandar tensión al LED.
- ✗ Establezco el pin 9 como entrada para detectar cuando está pulsado el botón.
- ✗ El circuito del LED es el de siempre. Del pin 8 y en serie resistencia de $1\text{k}\Omega$ con LED y a tierra. Recordar que la pata del larga del LED es la que debe estar conectada a mayor tensión.
- ✗ El circuito del pulsador esta montado como resistencia pull-down. Con ello consigo que cuando no se pulse el botón el pin 7 está conectado a tierra y se descarga cualquier tensión que pudiera llegar a el y que diera dar una falsa lectura HIGH. Al pulsar el botón se cierra el circuito. La tensión de 5V llega al botón desde la placa protoboard. La intensidad circula hacia 9 y no por la resistencia (es decir el camino que ofrece menos oposición al paso de corriente).