Identificación sensores, actuadores e indicadores

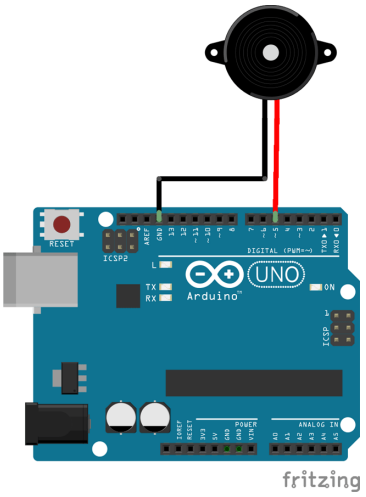
- **Alarma con Arduino → Buzzer (zumbador)**

Un buzzer o zumbador es un dispositivo que es capaz de enviar avisos a través del sonido, puede ser mecánico, electromecánico o piezoeléctrico. Si se aplica una señal eléctrica al material este se deforma y al deformarse produce un sonido. Ahora bien, si desde un pin de Arduino conseguimos suministrar una tensión que varía con el tiempo como una señal PWM, podremos reproducir diferentes sonidos.
A su vez, los zumbadores piezoeléctricos se dividen en dos grandes grupos: activos y pasivos.

La diferencia entre estos dos tipos de zumbadores es su funcionamiento interno.

- Un zumbador activo tiene un circuito adicional que lo hace más fácil de usar. Este circuito es un oscilador que hace que el zumbador suene tan pronto como se conecte a la alimentación.
- El zumbador pasivo, el oscilador interno desaparece y su funcionamiento depende de la una señal externa. Esta señal debe ser una señal cuadrada.

La conexión de un buzzer con Arduino consiste en conectar el positivo a un pin digital PWM y el negativo a GND. En ocasiones, se pone una resistencia entre el pin digital y el buzzer, debido a que la intensidad máxima que puede suministrar una placa Arduino por cada uno de sus pines es de 40 mA.



El terminal positivo se conecta al pin digital PWM 5 y el terminal negativo al pin de GND.

Código Arduino:

Queremos que el buzzer se active como alarma ante un determinado evento. Supongamos que tenemos una cierta función GetSystemState() que realiza las mediciones de sensores y cálculos necesarios para determinar si la alarma tiene que ser encendida.

Simplemente realizamos la llamada a la función, y encendemos la alarma si es necesario, manteniéndola encendida un mínimo de 5 segundos.

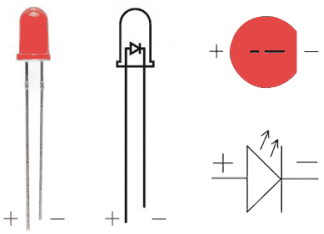
```ino
#include <EasyBuzzer.h>

void sonidoTerminado(){
  Serial.println("Sonido terminado");
}

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

  // Configuración del pin
  EasyBuzzer.setPin(5);

  Serial.println("Comienza el sonido");

  // Configuración del beep
  EasyBuzzer.beep(
    2000,          // Frecuencia en herzios
    100,           // Duración beep en ms
    100,           // Duración silencio en ms
    2,             // Números de beeps por ciclos
    300,           // Duración de la pausa
    1,             // Número de ciclos
    sonidoTerminado// Función callback que es llamada cuando termina
  );
}

void loop() {
  // Función para que funcione la librería
  EasyBuzzer.update();
}
```

```ino
const int pin = 9;

bool isAlarmOn = 0;     //almacena el estado de la alarma

void setup() {
  pinMode(pin, OUTPUT);  //definir pin como salida
}

bool GetSystemState()
{
    return true;  //cambiar en función del sensor usado
}

void loop(){
  isAlarmOn = GetSystemState();

  if(isAlarmOn)
  {
     digitalWrite(pin, HIGH);    // poner el Pin en HIGH
     delay(5000);               // esperar 5 segundos
  }
  else
  {
     digitalWrite(pin, LOW);    // poner el Pin en LOW
  }
  delay(1000);
}
```

- **Diodos leds**

Un LED es un diodo emisor de luz, es decir, un tipo particular de diodo que emite luz al ser atravesado por una corriente eléctrica. Los diodos (emisor de luz, o no) son unos de los dispositivos electrónicos fundamentales.

Un diodo es una unión de dos materiales semiconductores con dopajes distintos, tienen polaridad, es decir, solo dejan pasar la corriente en un sentido. Por tanto, tenemos que conectar correctamente la tensión al dispositivo. La patilla larga debe ser conectada al voltaje positivo (ánodo), y la corta al voltaje negativo (cátodo).



A baja tensión los electrones siguen sin poder atravesar el dispositivo. Esto ocurre hasta alcanzar un cierto valor de tensión que se denomina tensión de polarización directa (Vd), que depende del tipo de diodo, a partir de esta tensión se dice que el diodo está polarizado y la corriente puede atravesarlo libremente con una resistencia casi nula. En el momento que se supera la tensión de polarización, se genera una gran corriente que destruirá el diodo, por ese motivo, necesitamos una resistencia que limite la cantidad de corriente que circula por el diodo.

La tensión que soporta el LED es la diferencia entre la tensión aplicada y la tensión de polarización directa del LED. Aplicando la ley de Ohm, con el valor de la intensidad nominal del LED

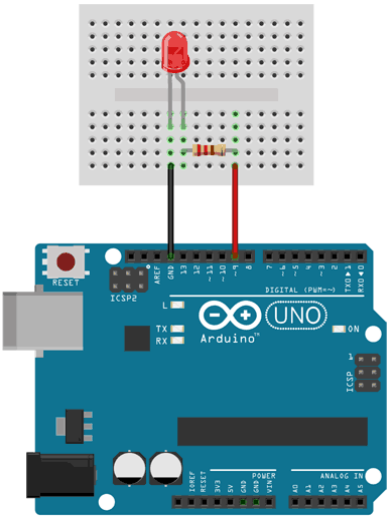
$$V = V_{cc} - V_d = I_{nominal} * R$$

Por lo que lo que el valor de la resistencia resulta: $$R = \frac{V_{cc} - V_d}{I_{nominal}}$$

Respecto a la tensión de polarización y la corriente nominal dependen de los materiales y constitución interna del diodo. En el caso de diodos LED convencionales de 3mm y 5mm, dependen principalmente del color y luminosidad.

Dentro de los LED de pequeña potencia, los más habituales son los encapsulados tradicionales de LED de 3mm o 5mm. También podemos encontrar LED opacos (diffused) o LED transparentes (clear), leds con diferentes ángulos, o que tengan el encapsulado de un color.

Montaje en una protoboard:



Código en Arduino:

Para encender y apagar un LED

```
1.  const int ledPIN = 9;
2.
3.  void setup() {
4.    Serial.begin(9600);    //iniciar puerto serie
5.    pinMode(ledPIN , OUTPUT);  //definir pin como salida
6.  }
7.
8.  void loop(){
9.    digitalWrite(ledPIN , HIGH);    // poner el Pin en HIGH
10.   delay(1000);                    // esperar un segundo
11.   digitalWrite(ledPIN , LOW);     // poner el Pin en LOW
12.   delay(1000);                    // esperar un segundo
13. }
```