

MÓDULO 4: PROYECTO 7 "Teclado musical"

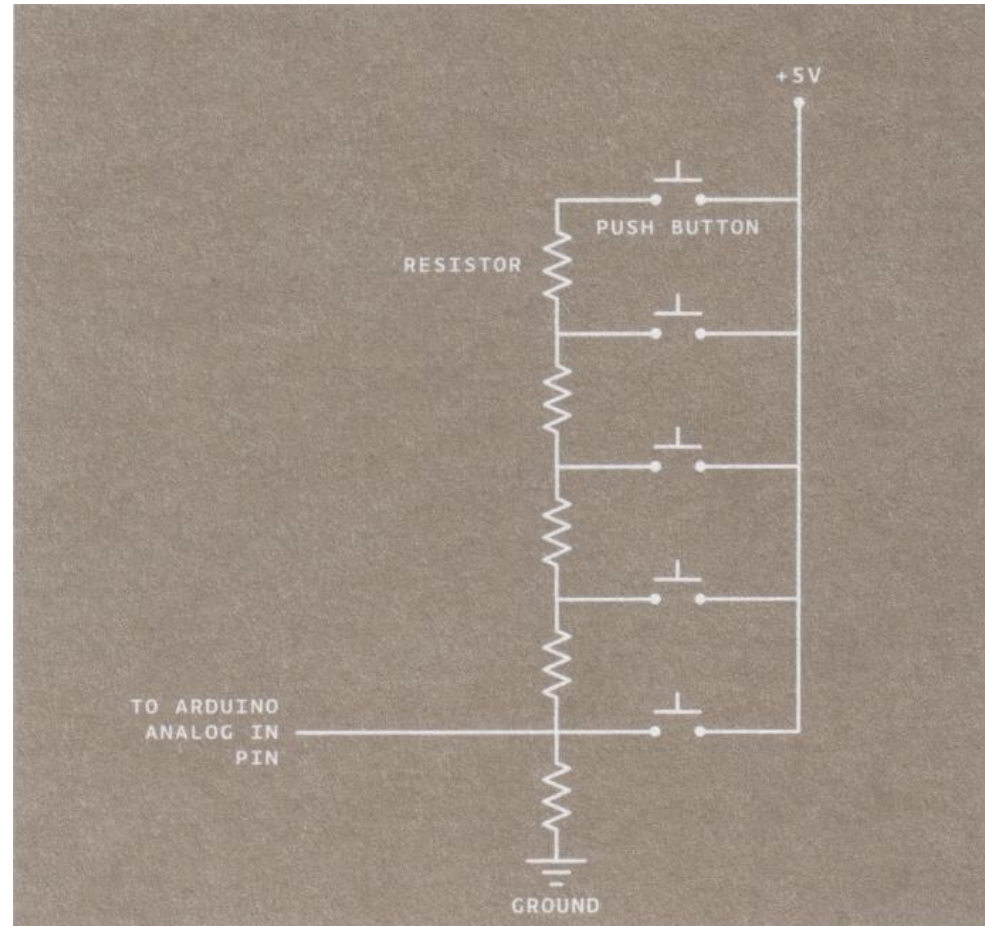
CURSO PROGRAMACIÓN DE PLACAS ROBÓTICAS

A solid orange horizontal bar at the bottom of the slide.

Introducción

- En este proyecto hará un pequeño teclado musical con botones y el zumbador piezoeléctrico. Va a utilizar una nueva técnica llamada “escalera de resistencia. Cada botón que pulsa proporcionará una tensión diferente a un pin de entrada analógica.
- Dicha técnica nos permite leer cierto número de pulsadores utilizando la entrada analógica de nuestro Arduino. Es muy útil cuando nos encontramos con que andamos cortos de entradas digitales.
- Tendremos varios pulsadores encadenados en paralelo a la entrada analógica cero, algunos de ellos conectados a la alimentación mediante una resistencia. Cuando pulsemos cada uno de ellos, llegará un nivel diferente de tensión al pin de entrada.

Introducción

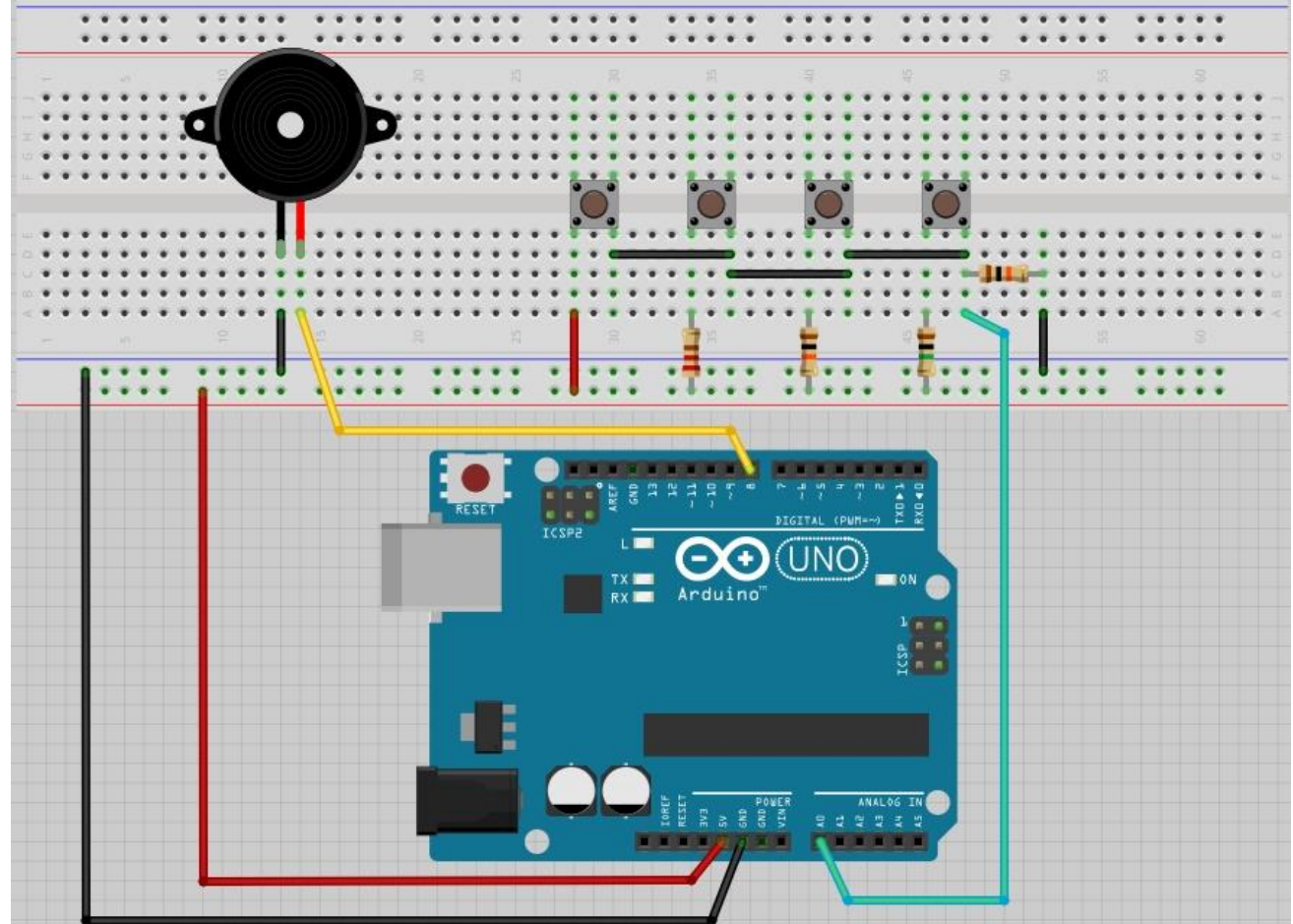


Montando el circuito

1. Conectamos los cables de alimentación a la protoboard.
2. Una patilla del altavoz la unimos a tierra y la otra al pin 8 del Arduino.
3. Colocamos los pulsadores a lo largo de la separación central de la protoboard.
4. Seguidamente, conectaremos la entrada de cada pulsador a la alimentación. El primero estará unido directamente, el segundo mediante una resistencia de 220Ω , el tercero mediante una de $10K\Omega$ y el cuarto con una de $1M\Omega$.
5. Por último, conectaremos la salida de cada pulsador con un cable. La salida del primero irá a la salida del segundo, la del segundo al tercero, la del tercero a la salida del cuarto y por último, la salida del cuarto la conectaremos a la entrada analógica cero de nuestro Arduino y a tierra mediante una resistencia de $10K\Omega$.

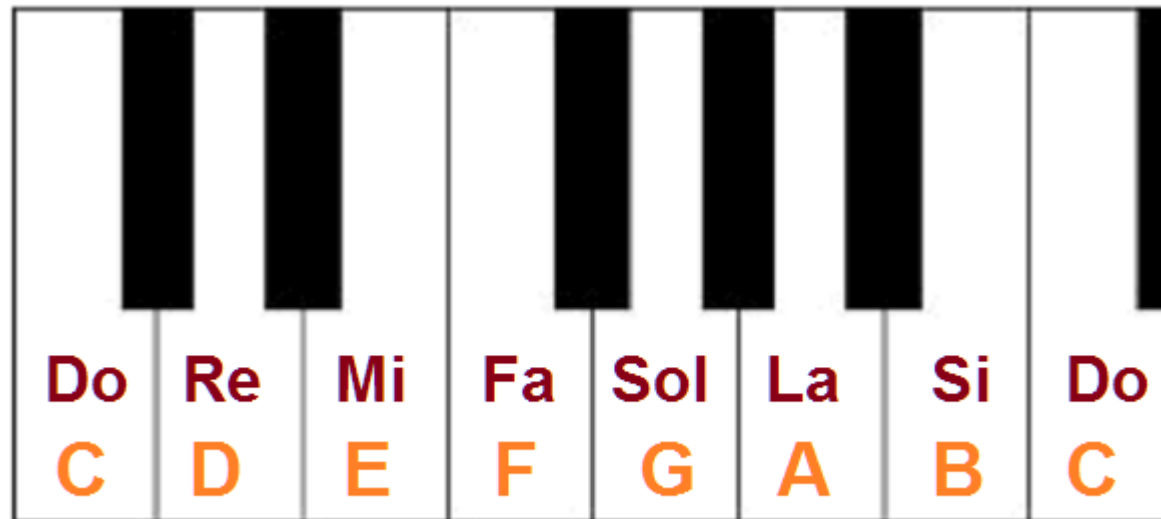
Montando el circuito

Componentes	
Pulsadores	4
Altavoz piezoeléctrico	1
Resistencia 220 Ω	1
Resistencias 10 K Ω	2
Resistencia 1M Ω	1



Comentario del código

- Para empezar, necesitaremos una lista de las frecuencias que vamos a reproducir cada vez que activemos un pulsador. Escogeremos por ejemplo las frecuencias de 262Hz, 294Hz, 330Hz y 349Hz (las frecuencias de medio C, D, E y F). Para conseguir esto, necesitamos un array.



Comentario del código

- Para lograr que este array sea una variable global y que por consiguiente sea visible desde cualquier parte de nuestro programa, lo declararemos antes de la función **setup()**.
- En la función **loop()**, declararemos una variable local para almacenar el valor obtenido del pin A0. Como cada pulsador tiene una resistencia diferente conectada a la alimentación, también cada uno de ellos tendrá asociado un valor diferente al del resto. Para poder ver dichos valores, añadiremos la línea **Serial.println(keyVal)**, así nos enviará la información a nuestro pc.
- Vamos a empezar a reproducir las notas. Tras cada sentencia **if()**, llamaremos a la función **tone()**. El programa hará referencia al array para determinar que frecuencia debe reproducir. Si el valor de A0 coincide con una de las sentencias **if()**, entonces, le diremos a nuestro Arduino que reproduzca un sonido.

Comentario del código

- Es posible que nuestro circuito posea un poco de ruido y esto haga que los valores fluctúen un poco cuando presionamos el pulsador.
- Para adaptarnos a dicha variación, es buena idea disponer de un rango de valores con los que contrastar. Si utilizamos el símbolo de comparación “&&”, podemos comprobar múltiples condiciones y ver cual es cierta.
- Si activamos el primer pulsador, se reproducirá notes[0]. Si es el segundo, será notes[1] y si presionamos el tercero se reproducirá notes[2].
- Para detener la reproducción de las notas cuando no haya ningún pulsador activado, llamaremos a la función **noTone()**, a la que le diremos el número de pin al que debe indicar que pare de reproducir el sonido.