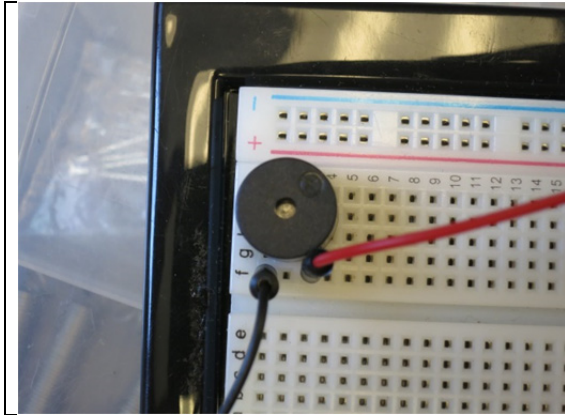


Arduino Lab 2—Music, Switches, Analog Input

1. Making Music!

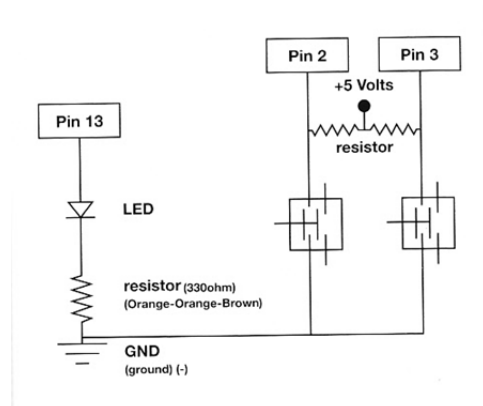


Get a piezobuzzer. Notice that one pin is marked positive. Put it into your breadboard, and connect the positive pin to pin 9 of your Arduino. Connect the other pin to Ground.

We can try 2 different music programs. Both are in the Resources/Arduino lab folder of our COSMOS Smartsite. Each plays a different melody. Open each program in turn, upload each, and listen to the melody it produces. Now try to modify one of the programs to play a different melody!

- A. First try Song1. This program has 2 arrays to keep track of the notes. The notes array lists the names of the notes, with a space to indicate a rest (i.e., no tone). The array beats lists the durations for each note, with 1 meaning quarter note, 2 meaning half note, etc. The duration of the rests must be specified also. The subroutine frequency takes the name of the note as an argument and returns the frequency. The song only plays once, but you can play it again by pushing the Reset button on the Arduino.
- B. Then try Melody. Note that this program requires the header file pitches.h to be in the same folder (/home/Sketchbook/Melody) as the Melody.ino program. This program works in a completely different way. The note frequencies are specified in the file “pitches.h” A preprocessor substitutes the frequencies for the names of the notes before the program is compiled. Then the compiler takes the code and makes the executable program which you upload. Pitches.h defines 8 octaves of note frequencies, together with sharps, so you can play any note which would occur on a piano keyboard. You specify the sequence of notes in the array melody, and the lengths of the notes in the array noteDurations, where 4 = quarter note, 8 = eighth note, etc.

2. Push buttons.



Connect the push buttons, the LED, and the resistors as shown above. Connect a resistor around 330Ω to the LED. Use about 10kΩ for the “pull-up” resistors that connect the switches to +5V. When the button is pushed, the voltage to the Arduino goes LOW it is connected to the other end of the 10kΩ resistor, which is connected to ground when the button is pushed.

Get the program PushButtons.ino from the Smartsite. Read the comments in the program for additional information about the electrical connections of the circuit. Adapt the program to use digital logic when the appropriate buttons are pushed to turn the LED on or off.

Now add additional LEDs, pushbuttons, and logic to turn them the LEDs on and off when you push the buttons.

3. Controlling Intensity from External Inputs

Build the following circuit and connect it to your Arduino.

