Fun with the Arduino and Electronics – Session B & C

Tech Resort Innovators Sessions #2103BC

In these sessions you are going to experiment with building electronic circuits. You will use an 'Arduino' microcontroller board to interface to your circuits in order to build computer controlled systems.

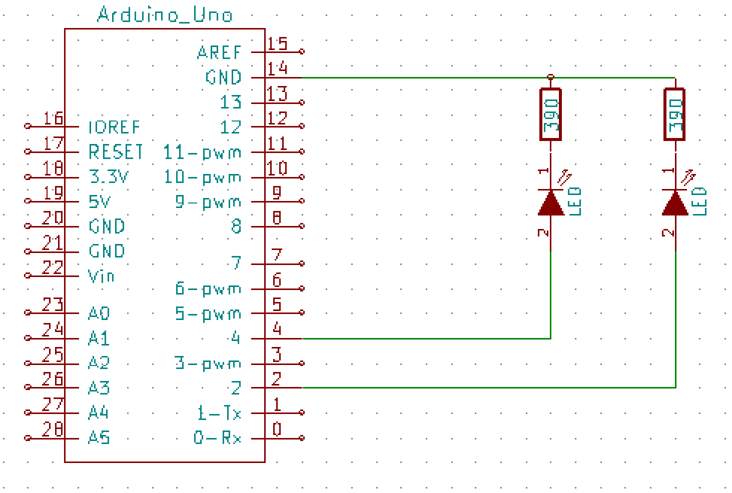
Skills involved:	**Coding**	**√**
	Electronics	**√**
Suitable for ages:	**10-14**	
You will need:	**An Arduino Microcontroller**	
	A PC with the Arduino programming language loaded	
	Some electronics components as set out below	
	A 'Solderless Breadboard'	

Introduction

- Having (hopefully successfully!) completed Session A of *Fun with the Arduino and Electronics,* in the next two sessions we're going to explore more ways of controlling electronic circuits. We'll suggest a few challenges: you don't have to do all of them, just have a look to see which ones you fancy or invent your own!

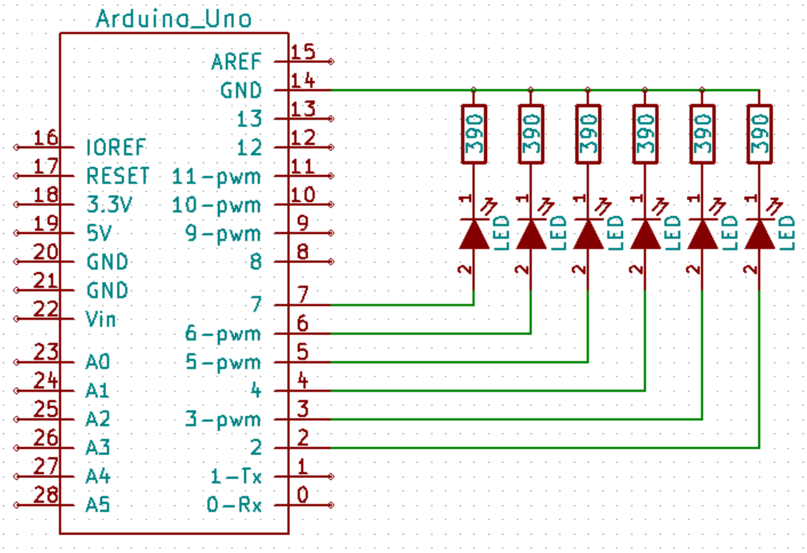
Revision

- Just to get you going, let's revisit what we did last week. Can you put a circuit like this together and get the two LEDs flashing independently of each other?



Challenge 1 – Knightrider LEDs

- In the 1980's there was a TV series called Knightrider that starred a guy named Michael (great name) who had many adventures in a special car called Kit. (Why does a car need a name?). This car had a row of red lamps along the top of the radiator grill that were illuminated one at a time to create a "moving light" effect. You can see this car in a YouTube clip - https://www.youtube.com/watch?v=Mo8Qls0HnWo
- Now your challenge is to create a display like Kit's. Mount all the red LED's on your Breadboard and create a program in the Arduino that switches them on sequentially to create a "Knight Rider" effect.
- Hint – You will need to repeat the flashing LED circuit from Session A several times.
- Connect the LED's to Pins 2 to 9 on the Arduino board.
- Does your circuit look a little bit like the one shown below?

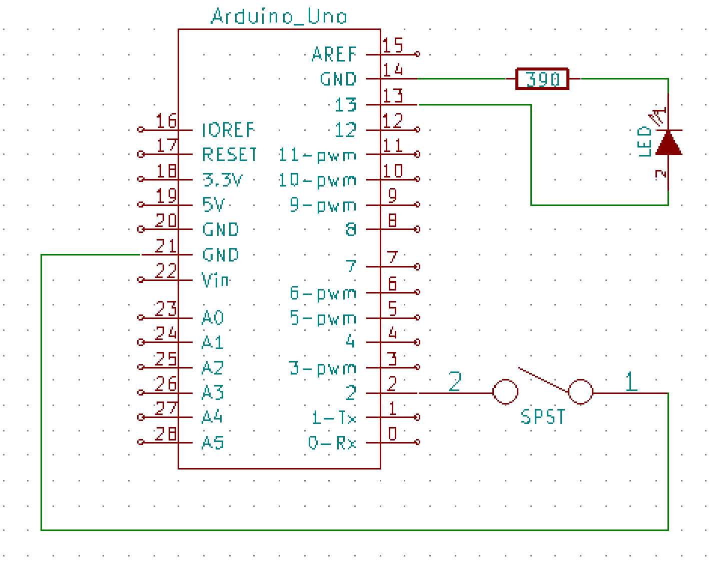


Challenge 2 – Using a Switch as an Input

- First load this program from the examples: **File > Examples > 02. Digital > Button**
- It uses a button to turn an LED on and off, not directly but by using the Arduino to sense whether the button has been pressed or not.
- You'll need to make a change to the program. Where it says:

  pinMode(buttonPin, **INPUT**) change this to:

  pinMode(buttonPin, **INPUT_PULLUP**)
- Now have a go at building the circuit:

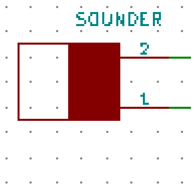


- Does the program do what you thought? If you haven't changed anything else, pressing the button turns the LED off and not the other way round!
- Can you change it so that pressing the button turns the LED ON?

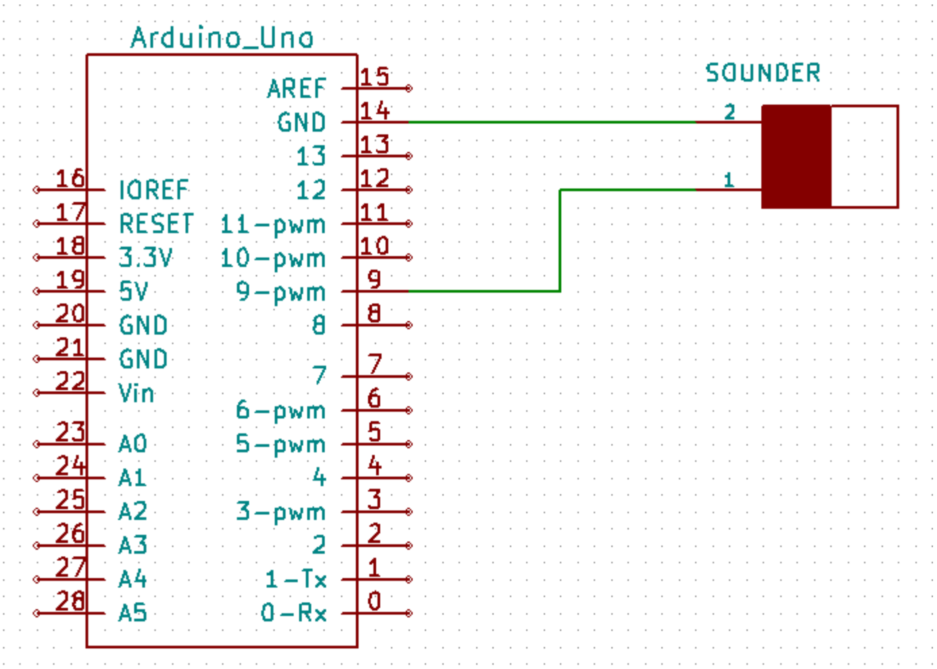
Challenge 3 – Sound

- Let's find one of these: it's a type of miniature loudspeaker called a 'piezoelectic sounder'. We'll just call it a 'Sounder'

- It has 2 pins which you can plug straight into a breadboard. It doesn't matter which way round. Here's what it looks like on a circuit diagram:



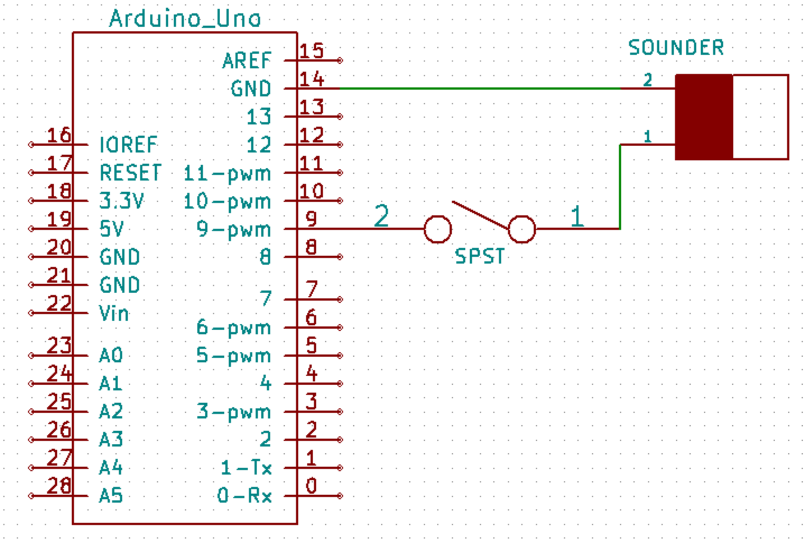
- It’s easy to control this using an Arduino. Connect them together like this:

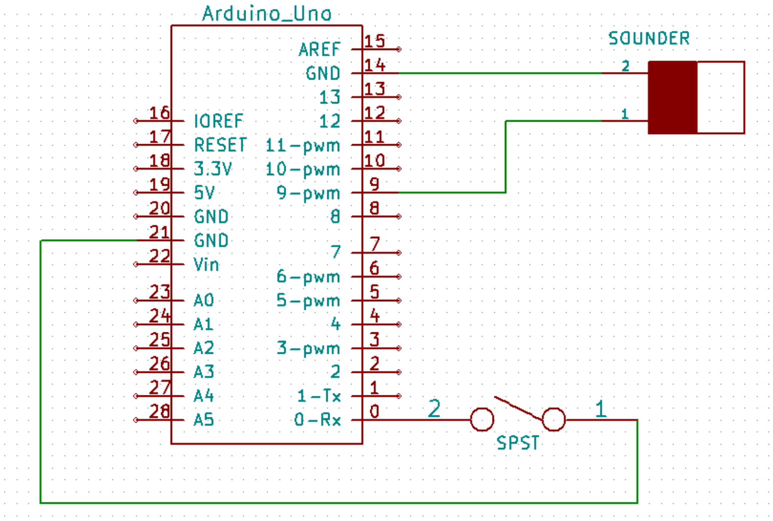


- The **tone()** command is used to send a signal to a sounder. For example the command **tone(9, 262, 1)** will play a note of C (262 Hertz) on a sounder connected to pin 9 for 1 second. See if you can program a sequence of different sounds. Remember to set pin 9 as an output!
- The program **toneMelody** (found in **File->Examples->02. Digital**) is a good example of how to this but it has one important difference in how it expects you to wire up your sounder. Can you spot it?

Challenge 4 – Turning Sound on and Off

- You can add a switch so that so that the sound can be turned on and off manually. There are two ways you could do this:

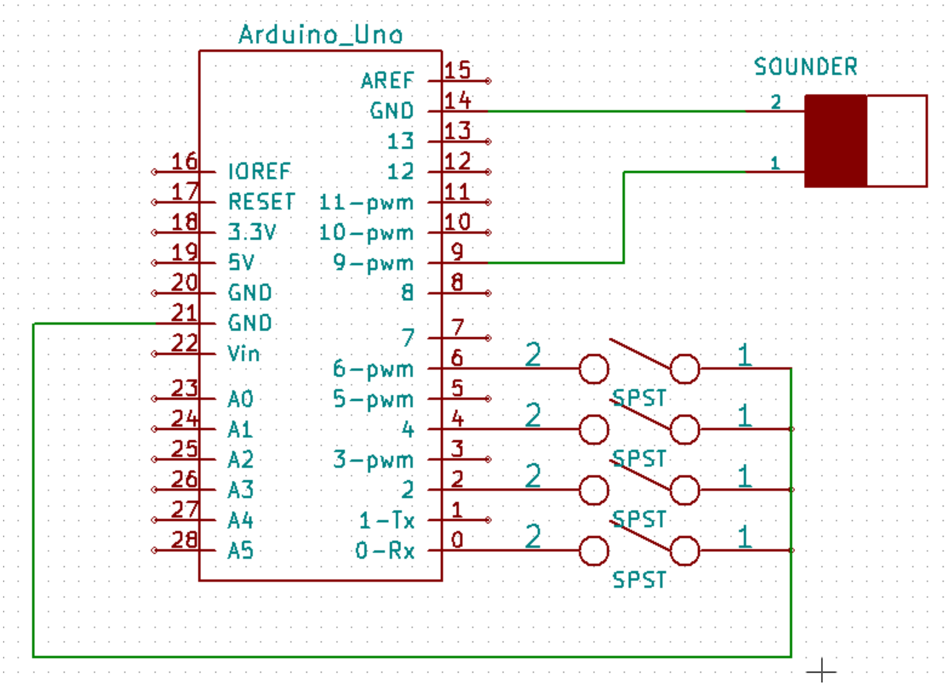




- The first is easier to understand and program but the second is more flexible! The program created in Challenge 2 shows you how to go about programming a button. You should be able to edit this to control a sounder rather than an LED. Note that you may have used different pin numbers in this circuit though!

Challenge 5 – A Musical Instrument

- If you've successfully completed Challenge 3 let's try making a musical instrument. We need to wire up something like this:



- Now can you program each button so that pressing it results in a different note from the sounder?

Challenge 6 – Playing Tunes

- Instead of playing a single note when we press one of the buttons in the above circuit, we could try and program our Arduino to play a tune.
- Each note in a tune is a different frequency so here's a table of what frequency corresponds to a particular note:

Note	Frequency		Note	Frequency	
E3	165		F4	349	
F3	175		F#4	370	
F#3	185		G4	392	
G3	196		G#4	415	
G#3	208		A4	440	
A3	220		A#4	466	
A#3	233		B4	494	
B3	247		C5	523	
C4	262		C#5	554	
C#4	277		D5	587	
D4	294		D#5	622	
D#4	311		E5	659	
E4	330		F5	698	

- Here are the notes that make some Christmas songs. You'll have to figure out how long the duration of each note is when you program it though:
- Jingle Bells:

E4, E4, E4

E4, E4, E4

E4, G4, C4, D4, E4

- White Christmas:

E4, F4, E4, D4, E4, F4, FS4, G4

- Rudolph the Red Nosed Reindeer:

G4, A4, G4, E4, C5, A4, G4

G4, A4, G4, A4, G4, C5, B4

- If you like, have a look on the internet to see what other tunes you might like to program....