

Lab 8B - Arduino Drum Machine (ARD)

Lab Assignment

1. Demonstrate 3 LEDs flashing at different rates using 3 objects of the Flasher class shown in the section titled "A Classy Solution"
2. Add a piezo speaker to your breadboard. Use a copy of Flasher class to make a Beat class. This class will send brief tone pulses to the piezo, allowing you to create a "Drum Machine" emulator that will produce a steady stream of beats of different pitches and duration and timing patterns.



You can get an idea of what the possibilities are for your drum machine by playing with the online drum machine at <https://drumbit.app/> (Links to an external site.)



Extra Challenge - Add a pushbutton switch to be able to do an additional pulse/drumbeat whenever the button is pressed so you can do a "solo"!

Read Adafruit: Multi-tasking the Arduino Pt 1

<https://learn.adafruit.com/multi-tasking-the-arduino-part-1>

Key Points:

- 1) Using the delay() command is inefficient, there are better ways
- 2) Use millis() and state machines to be more efficient
- 3) This efficiency allows you to do more interesting things such as:
 - Elements with different timings
 - Simpler code design
 - Resulting in adding interest without adding complexity

Three Class Elements

1. Define the class:

```
class Flasher {.....}
```

2. Add a constructor which creates a class instance:

```
public:
```

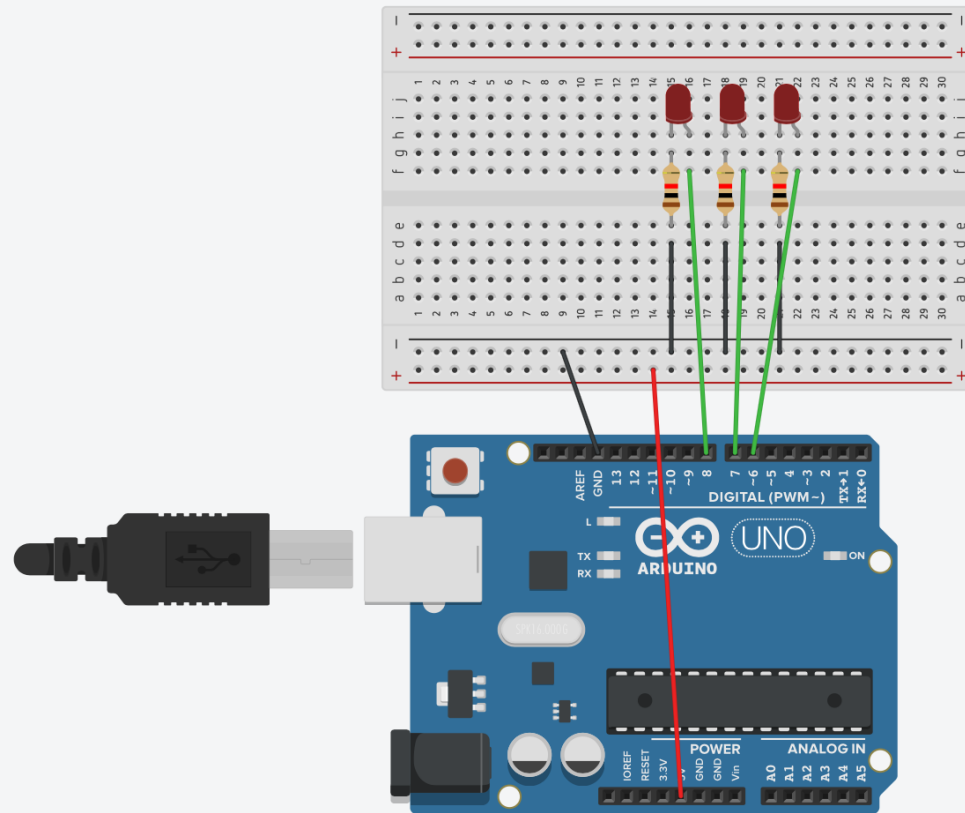
```
Flasher(int pin, long on, long off)
```

```
{ ....}
```

3. Add an action which creates a class behavior:

```
void Update() {....}
```

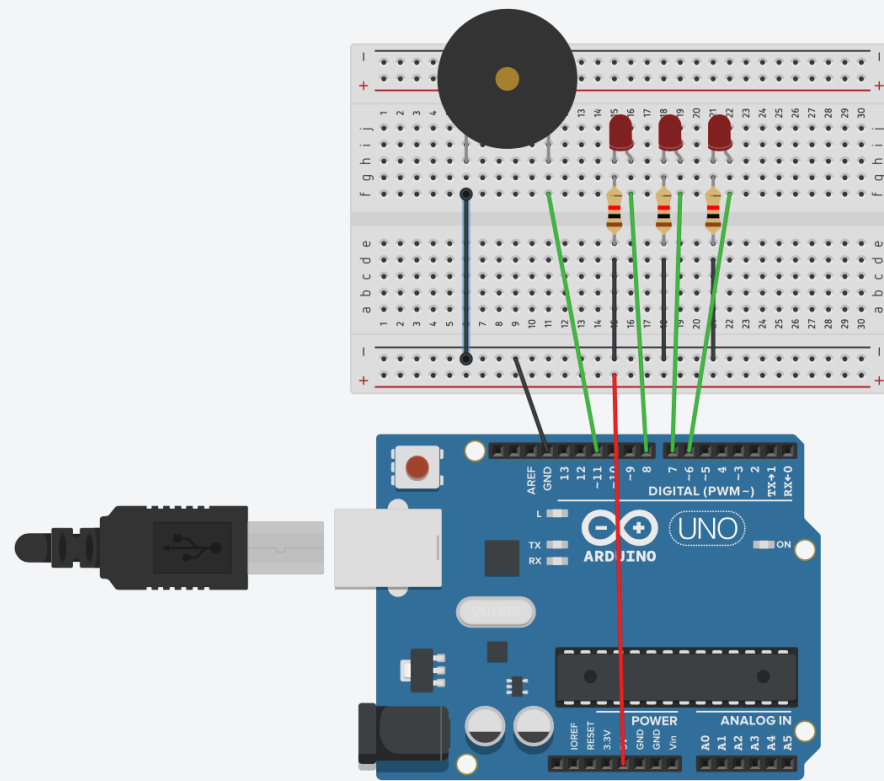
Build the circuit below



Copy the code from Tutorial page

- Use the **complete** code from the page “a classy solution” (be sure to get all three elements)
- Note that there are three LED’s in our example and the pins are different
- **For Problem 1, have all three flashing at different frequencies**

Add sound by adding the piezo speaker



Create a Beat Class identical to the Flasher

Duplicate the Flasher class and rename it Beat then make changes to account for the piezo speaker.

Hints:

- When adding the Beat class, add it immediately after the Flasher class and **before** the void setup and void loop functions.
- Remember in C++, class definitions end with a }; so both Flasher and Beat will need to close properly.

More Hints

1. Just as the LED uses, `digitalWrite(pin, state)`, the speaker will require the use of `tone(pin, frequency)` or `noTone(pin)` to play a note or not.
2. In the Flasher definition, there were three parameters, `pin`, `OnTime` and `OffTime`. What are the right three parameters for the speaker?
 1. Does the pin # need to be a parameter? (Hint: there is only one speaker.)
 2. Clearly Off time and On time, are needed.
 3. What else is needed? (think about how the piezo makes sound)
3. What needs to be changed in the Update function?

Possible Issue

- Tinkercad's simulation of the Arduino isn't great and the piezo speaker isn't perfect.
- In actual comparisons, sound isn't as nice as a voice coil speaker.
- Links to hear the difference:
- [Piezo Speaker](#) *(Links to an external site.)*
- [Voice Coil Speaker](#) *(Links to an external site.)*
- Sometimes the sound doesn't work in Tinkercad simulator.
- It helps to stop simulation and restart if no sound from the speaker.



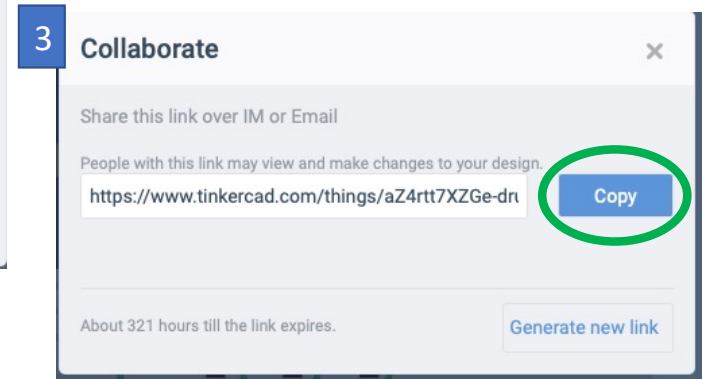
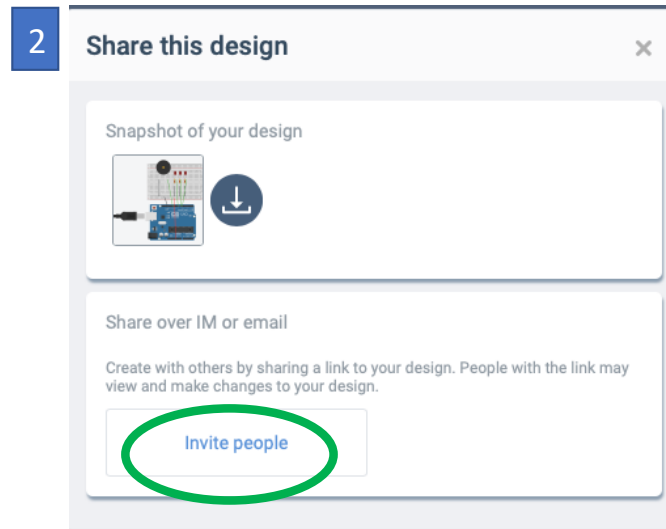
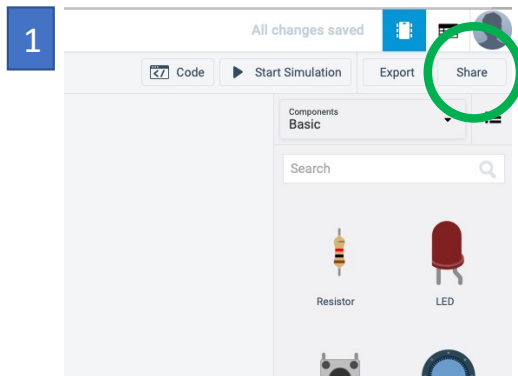
Piezo Speaker



Voice Coil Speaker

For Problem 2

- Create a link for your TinkerCad project and paste in your lab.



Remember the Challenge!

- **Extra Challenge** - Add a pushbutton switch to be able to do an additional pulse/drumbeat whenever the button is pressed so you can do a "solo"!