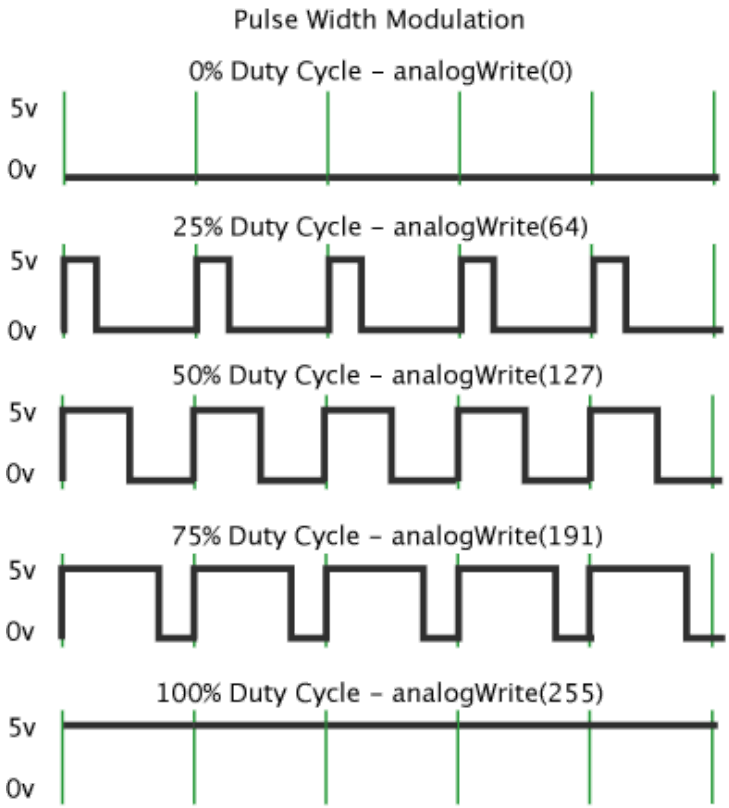
Student Activity Sheet

ACTIVITY 1G: How to Create a Fading LED

Purpose: This activity demonstrates the use of the analogWrite() function in fading an LED off and on. AnalogWrite uses pulse width modulation (PWM), turning a digital pin on and off very quickly with different ratio between on and off, to create a fading effect.

Materials: Computer, Arduino or Genuino board, LED, 220 ohm resistor, hook-up wires, breadboard

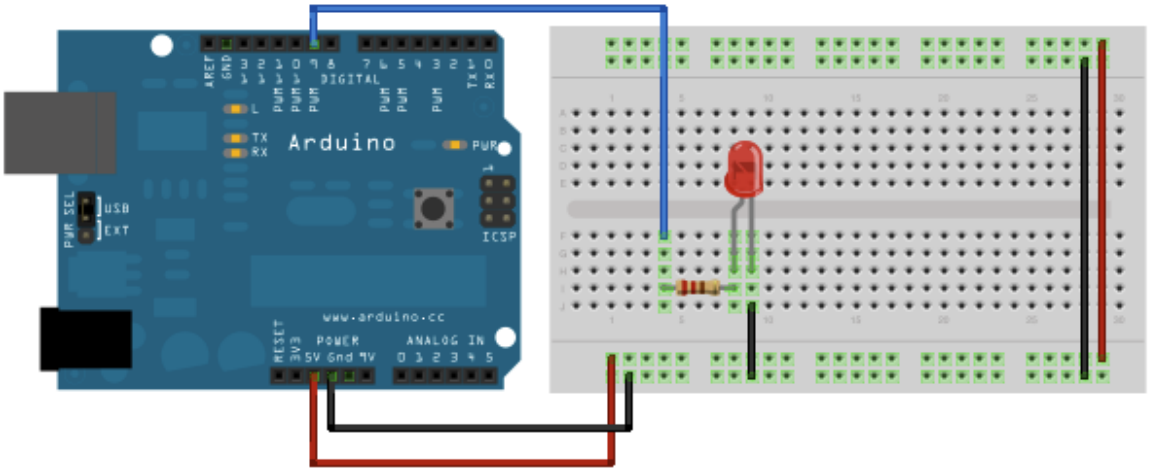
New Ideas: This is the first time we've used integer variables (like X and Y in math) and PWM (Pulse Width Modulation). In order to fade your LED off and on, gradually increase your PWM value from 0 (all the way off) to 255 (all the way on), and then back to 0 once again to complete the cycle. The Arduino controller can flash the LED much faster than we can see. We see slower flashing as dimmer and more constant flashing as brighter.

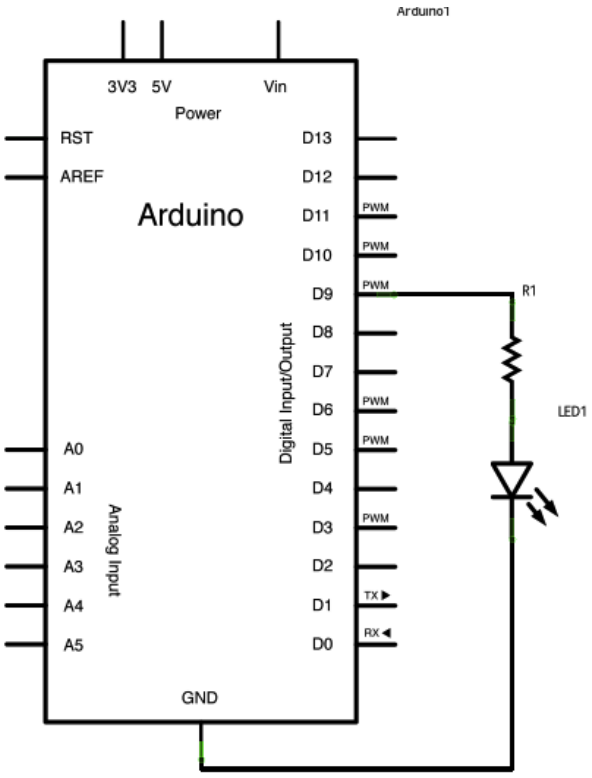


With funding from Canada

SIEC Saskatoon Industry Education Council

Circuit: Connect the anode (the longer, positive leg) of your LED to digital output pin 9 on your board through a 220 ohm resistor. Connect the cathode (the shorter, negative leg) directly to ground.





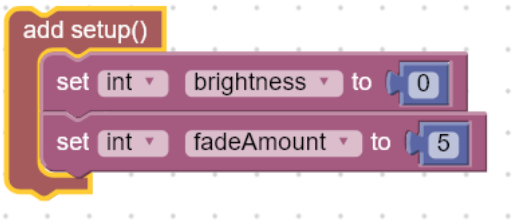
With funding from
Canada
<SaskCode>
SIEC
Saskatoon Industry Education Council

Code

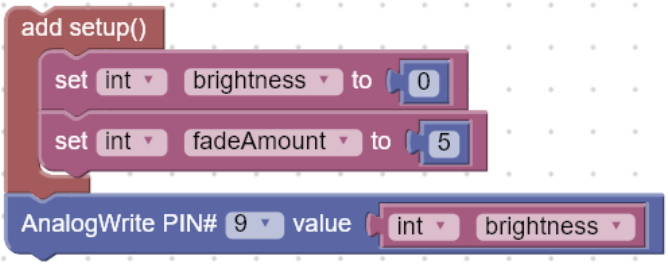
You can create your code in the BlocklyDuino Editor then paste it into the Arduino IDE.

In order to fade your LED off and on, gradually increase your PWM (Pulse Width Modulation) value from 0 (all the way off) to 255 (all the way on), and then back to 0 once again to complete the cycle. In the sketch below, the PWM value is set using a variable called “brightness”. Each time through the loop, it increases by the value of the variable “fadeAmount”.

We first create the variables



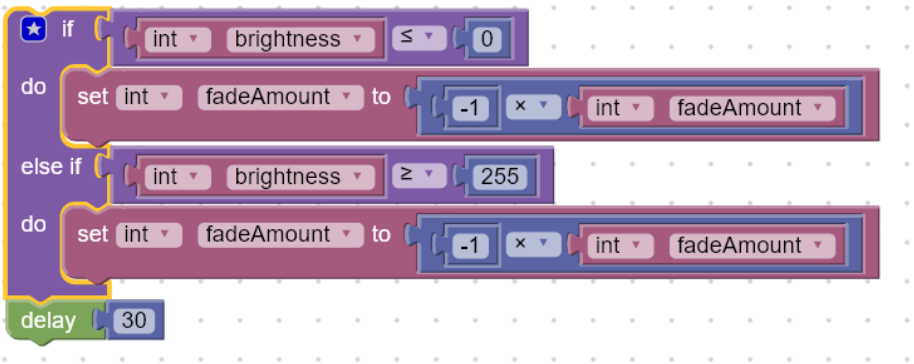
The AnalogWrite statement is added to tell the led in pin 9 how bright it is. Rather then giving a constant high or low value we use the brightness variable.



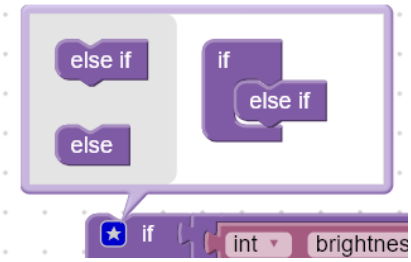
An increment counter is usually used to increment the LED brightness slowly. The set statement is used to add or subtract a value of 5 from the brightness.

If brightness is at either extreme of its value (either 0 or 255), then fadeAmount is changed to its negative. In other words, if fadeAmount is 5, then it is set to -5. If it's -5, then it's set to 5. The next time through the loop, this change causes brightness to change direction as well. The “if” statement is used to set the fadeAmount to positive or

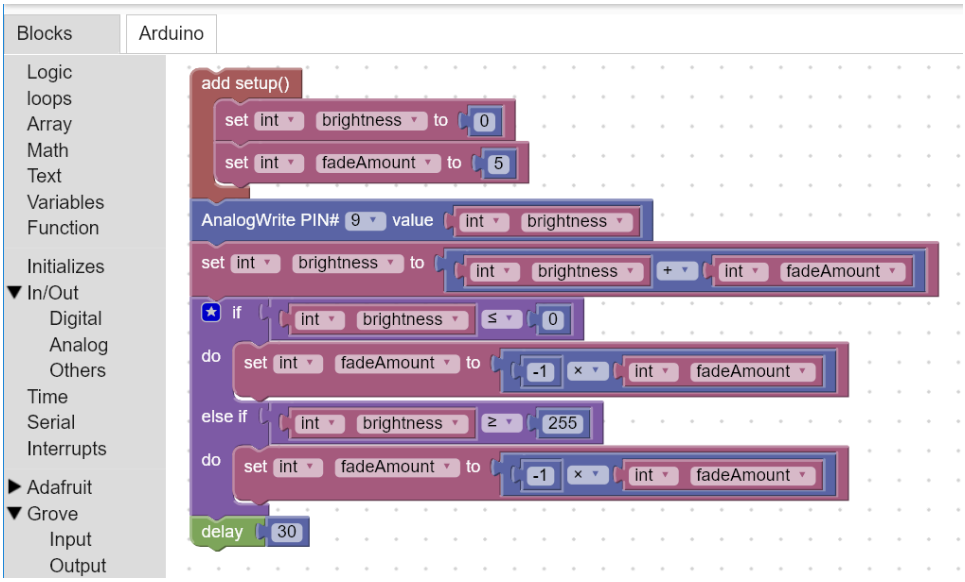
negative 5 if the value gets to 255 to then decrease and once at zero to increase brightness.



Note the little blue gear in the upper left corner of the if block. This allows you to add conditions to the statement like “else” or “else if”. Just drag and add pieces.



The “delay” at the end controls how fast fading occurs. It measures in milliseconds. Try changing the value of the delay and see how it changes the fading effect.



With funding from
Canada
<SaskCode>
SIEC Saskatoon Industry Education Council

*****SAVE your sketch to your BlocklyDuino Folder that you created*****

Upload your program to the arduino. Make sure the Arduino Uno board is hooked up using the supplied USB cord to your computer and click the green “upload” button.

Your program is now loaded. Observe your LED and complete your worksheet titled Assignment 1G: How to create a fading LED

With funding from Canada

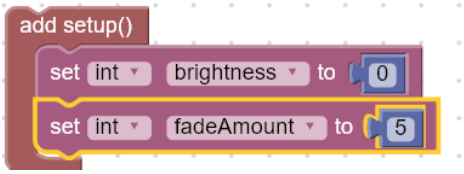
<SaskCode>

SIEC Saskatoon Industry Education Council

Assignment 1G: How to create a fading LED

1. Describe in detail what the LED is doing. (Observe the timing). What happens when you alter the delay from 30 to 50 or 10?

2. **THE CODE.** Code is a set of instructions used to communicate to the arduino.
    a. Setup and Loop: Arduino code that goes in between the curly brackets {} of setup() runs once. It is required to define all our variables. Variables are values that can change in the loop. We have set these values as integers. The same as you're studying in math class! What happens if you change the fadeAmount to 10?



    b. The code in between the loop()curly brackets {} runs over and over. Looking at the code, what steps repeat?

    c. How will the variable change each time you go through the loop?

Extension Challenges:

3. Change your code so that your LED fades at 5 second intervals. Explain how you changed your code to achieve this.

4. Change your code and add a second LED on another pin fading at a different rate. You may have to introduce additional variables to achieve this.

With funding from Canada

5. Connect a button switch to your circuit. Code your Arduino to start the fading when the button is pressed.

6. Code the LED to fade in and out only twice when the button is pressed.

7. Create a “night Light”. This means the LED slowly gets brighter as the room light gets darker. You will need to incorporate the photoresistor that you used in 1E.

CONCLUSION:

9. List two real life applications that this type of fading light might be used.

With funding from Canada