**LAB REPORT: Analog**

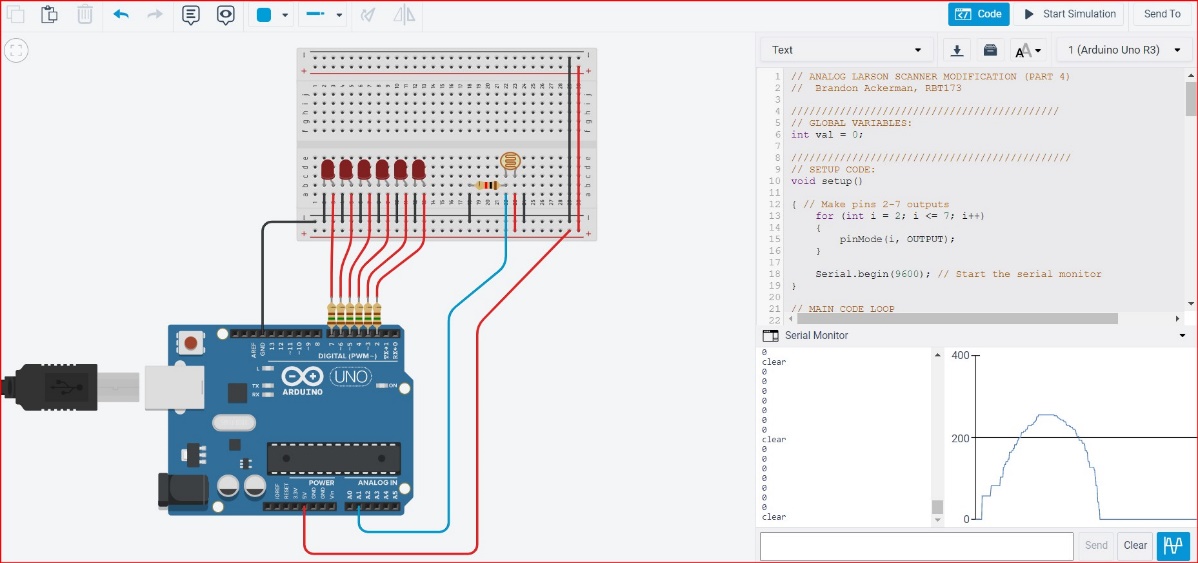
Brandon Ackerman – RBT173

**Introduction**

For this laboratory, we are tasked with a four-part demonstration of the analog features provided within the Arduino IDE. Using the pulse width modulation (PWM) features of analogWrite(), we make an LED fade in and out in a breathing pattern. Part 2 requires the use of analogRead() and the addition of a photoresistor. We modify our code to read the light levels of the photoresistor connected to pin A1, and then print(write) the data to the serial port. It is also requested that a graph be provided that shows the change in light values as it transitions from dim to light. For part 3, the minimum and maximum values printed to the serial console by the photoresistor are mapped to the 0-255 minimum and maximums through the use of the map command to be used with the analogWrite() function. The laboratory concludes with part 4, a modification to the previously built Larson scanner. The modification causes the lights on the scanner to fade in and out, as opposed to digitally switching on and off.

The circuits for these projects are very similar, if not possible to be condensed into one single schematic. The Analog circuit is composed of the Arduino Uno platform, an LED, 100 Ω resistor, 1K Ω resistor, and a photoresistor. We are already familiar with the composition of the Larson scanner circuit, which consists of 6 LED’s, 6 Resistors, and the Arduino UNO platform. The resistor values are calculated using the formula . Since the LEDs for the Larson Scanner are connected in parallel to the 5V pins, a resistor value of 150 Ω is used to provide each LED with 2V 20mA. For the Analog circuit, a resistor value of 100 Ω is used as a minimum value to protect the LED while allowing it to illuminate as bright as possible, and a value of 1K Ω is used for the photoresistor pull-down circuit to decrease voltage drain. I was unable to determine why the voltage of the circuit fluctuates, causing the LED’s to flicker, but I believe it has something to do with the timing of the loop in the code. I attempted to fix it for an entire day, but gave up due to the time constraints and the need to get the other projects finished.

**Implementation**

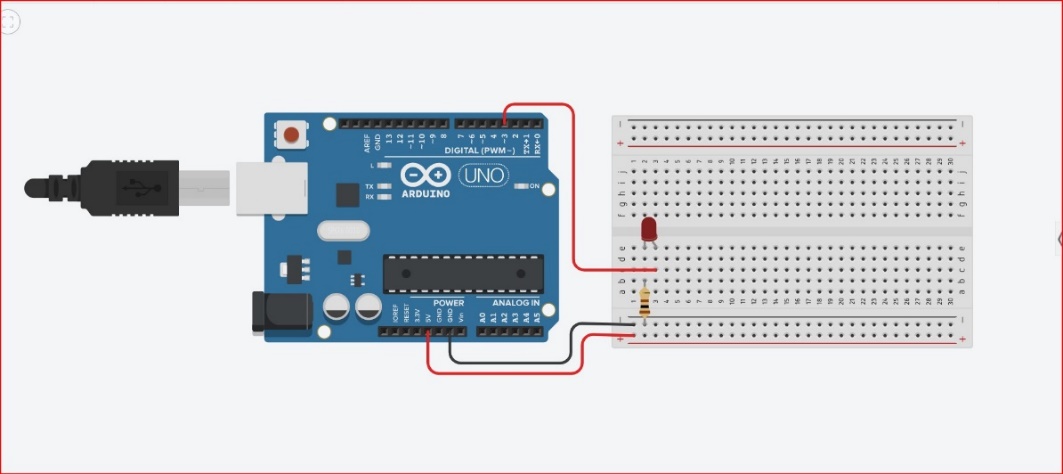


MODIFIED ANALOG LARSON SCANNER

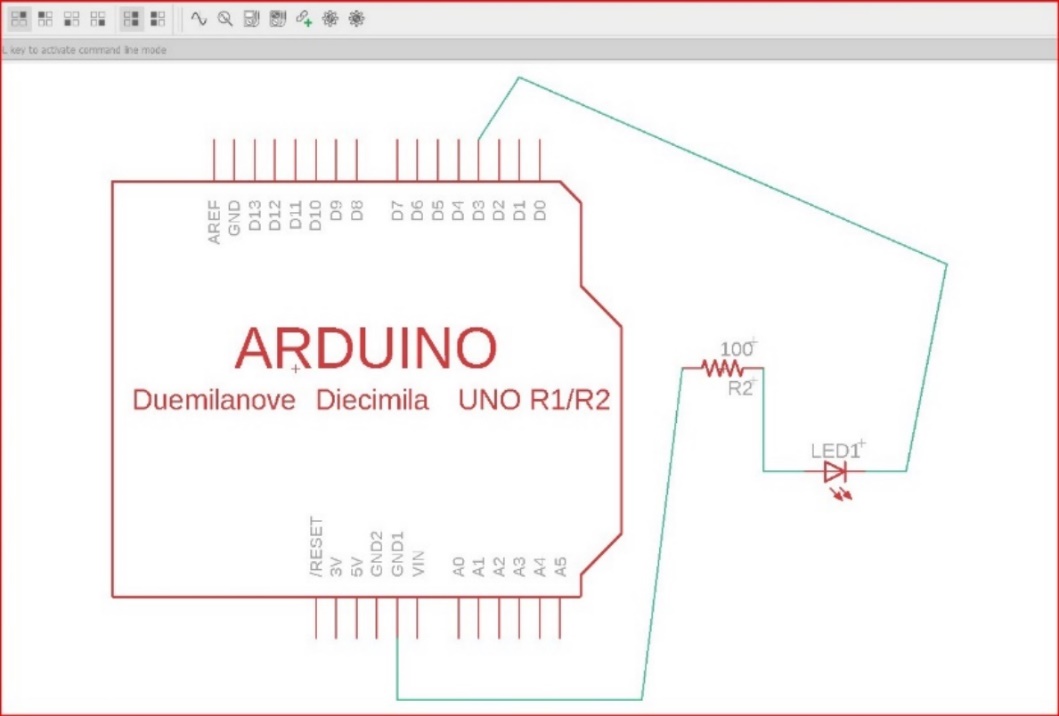
Diagram, schematic

Description automatically generated

MODIFIED ANALOG LARSON SCANNER DIAGRAM



ANALOG WRITE CIRCUIT



ANALOG WRITE DIAGRAM

Diagram

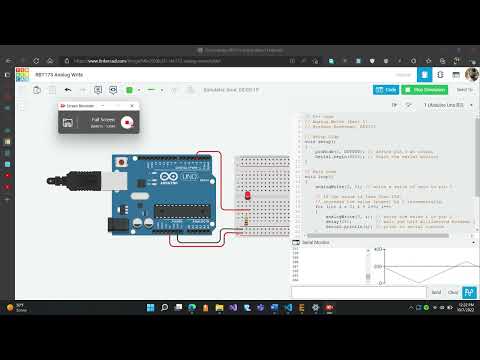
Description automatically generated

ANALOG READ CIRCUIT

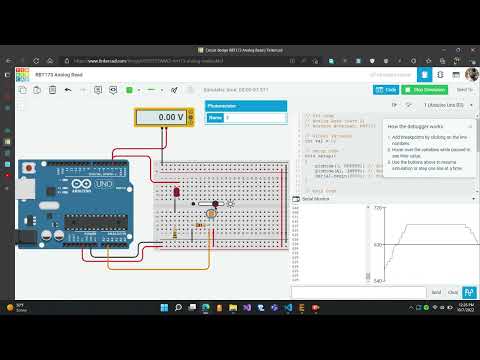
Diagram

Description automatically generated

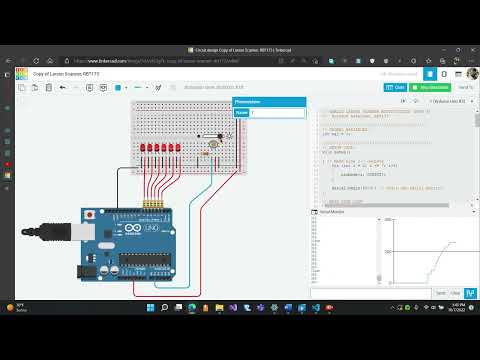
ANALOG READ DIAGRAM

[](https://www.youtube.com/embed/Qz9IE4ZVmo0?feature=oembed)

ANALOG WRITE VIDEO

[](https://www.youtube.com/embed/ng1D3Y8umEY?feature=oembed)

ANALOG READ VIDEO

[](https://www.youtube.com/embed/Gb2L2rDANkw?feature=oembed)

MODIFIED LARSON SCANNER VIDEO

*// Analog Write (Part 1)*

*// Brandon Ackerman, RBT173*

*// Setup Code*

*void* *setup*()

{

*pinMode*(3, OUTPUT); *// define pin 3 as output*

*Serial*.*begin*(9600); *// Start the serial monitor*

}

*// Main Code*

*void* *loop*()

{

*analogWrite*(3, 0); *// write a value of zero to pin 3*

*// if the value is less than 256,*

*// increase the value (power) by 1 incrementally.*

*for* (*int* *i* *=* 0; *i* *<* 256; *i++*)

    {

*analogWrite*(3, *i*); *// write the value i to pin 3*

*delay*(*0*5); *// wait one half millisecond between increments*

*Serial*.*println*(*i*); *// print to serial console*

    }

*// if the value is 255*

*// decrease the value (power) by 1 incrementally.*

*for* (*int* *i* *=* 255; *i* *>=* 0; *i--*)

    {

*analogWrite*(3, *i*); *// write the value i to pin 3*

*delay*(*0*5); *// wait one half millisecond between increments*

*Serial*.*println*(*i*); *// print to serial console*

    }

}

*// Analog Read (Part 2)*

*// Brandon Ackerman, RBT173*

*// Global Variable*

*int* *val* *=* 0;

*// Setup Code*

*void* *setup*()

{

*pinMode*(3, OUTPUT); *// define pin 3 as output*

*pinMode*(A1, INPUT); *// define pin A1 as input*

*Serial*.*begin*(9600); *// Start the serial monitor*

}

*// Main Code*

*void* *loop*()

{

*val* *=* *analogRead*(A1); *// assign the value of A1 to val.*

*Serial*.*println*(*val*); *// print to serial console*

*// if the value is greater than or equal to 679*

*// decrease the value (power) by 1 incrementally.*

*if* (*val* *>=* 679)

    {

*analogWrite*(3, 255); *// set pin 3 to 255 (Max Power).*

*delay*(100); *// wait one tenth of a second*

    }

*else*

    {

*// if val is not greater than or equal to 255*

*// set pin 3 to 0 (Power Off)*

*analogWrite*(3, 0);

    }

}

*// Analog Mapping (Part 3)*

*// Brandon Ackerman, RBT173*

*// Setup Code*

*void* *setup*()

{

*Serial*.*begin*(9600); *// Start the serial monitor*

}

*// Main Code*

*void* *loop*()

{

*int* *val* *=* *analogRead*(A1); *// assign the variable analogRead "val" to pin A1*

*// map the min/max values of the photoresistor/100Ohm circuit*

*//  (6-679) to the min/max 8 bit values of 0-255.*

*val* *=* *map*(*val*, 6, 679, 0, 255);

*Serial*.*println*(*val*); *// print to serial console*

*analogWrite*(3, *val*); *// write the value of A1 to pin 3.*

}

*// ANALOG LARSON SCANNER MODIFICATION (PART 4)*

*//  Brandon Ackerman, RBT173*

*////////////////////////////////////////////*

*// GLOBAL VARIABLES:*

*int* val *=* 0;

*//////////////////////////////////////////////*

*// SETUP CODE:*

*void* *setup*()

{ *// Make pins 2-7 outputs*

*for* (*int* i *=* 2; i *<=* 7; i*++*)

    {

*pinMode*(i, OUTPUT);

    }

*Serial*.*begin*(9600); *// Start the serial monitor*

}

*// MAIN CODE LOOP*

*void* *loop*()

{

*clear*(); *// sets all LED's to LOW*

*anlg*();

*if* (val *>=* 244)

    {

*digitalWrite*(2, HIGH);

*digitalWrite*(3, HIGH);

*digitalWrite*(4, HIGH);

*digitalWrite*(5, HIGH);

*digitalWrite*(6, HIGH);

*digitalWrite*(7, HIGH);

    }

*anlg*();

*if* (val *>=* 219)

    {

*digitalWrite*(2, HIGH);

*digitalWrite*(3, HIGH);

*digitalWrite*(4, HIGH);

*digitalWrite*(5, HIGH);

*digitalWrite*(6, HIGH);

*digitalWrite*(7, LOW);

    }

*anlg*();

*if* (val *>=* 183)

    {

*digitalWrite*(2, HIGH);

*digitalWrite*(3, HIGH);

*digitalWrite*(4, HIGH);

*digitalWrite*(5, HIGH);

*digitalWrite*(6, LOW);

*digitalWrite*(7, LOW);

    }

*anlg*();

*if* (val *>=* 147)

    {

*digitalWrite*(2, HIGH);

*digitalWrite*(3, HIGH);

*digitalWrite*(4, HIGH);

*digitalWrite*(5, LOW);

*digitalWrite*(6, LOW);

*digitalWrite*(7, LOW);

    }

*anlg*();

*if* (val *>=* 111)

    {

*digitalWrite*(2, HIGH);

*digitalWrite*(3, HIGH);

*digitalWrite*(4, LOW);

*digitalWrite*(5, LOW);

*digitalWrite*(6, LOW);

*digitalWrite*(7, LOW);

    }

*anlg*();

*if* (val *>=* 75)

    {

*digitalWrite*(2, HIGH);

*digitalWrite*(3, LOW);

*digitalWrite*(4, LOW);

*digitalWrite*(5, LOW);

*digitalWrite*(6, LOW);

*digitalWrite*(7, LOW);

    }

*anlg*();

*if* (val *>=* 39)

    {

*digitalWrite*(2, LOW);

*digitalWrite*(3, LOW);

*digitalWrite*(4, LOW);

*digitalWrite*(5, LOW);

*digitalWrite*(6, LOW);

*digitalWrite*(7, LOW);

    }

*delay*(10);

}

*// END MAIN CODE LOOP*

*// Function for analogRead()*

*void* *anlg*()

{

    val *=* *analogRead*(A1); *// assign the variable analogRead "val" to pin A1*

*// map the min/max values of the photoresistor/100Ohm circuit*

*//(6-679) to the min/max 8 bit values of 0-255.*

    val *=* *map*(val, 6, 679, 0, 255);

*Serial*.*println*(val); *// print to serial console*

}

*// Function to clear*

*void* *clear*()

{

*// set pins 2-7 to LOW (OFF)*

*digitalWrite*(2, LOW);

*digitalWrite*(3, LOW);

*digitalWrite*(4, LOW);

*digitalWrite*(5, LOW);

*digitalWrite*(6, LOW);

*digitalWrite*(7, LOW);

*Serial*.*println*("clear"); *// prints "clear" to the serial console*

}