

## **Shoebox DMX Console**

Adam Hoffart, 200338059

December 18, 2017

CS207  
Trevor Tomesh

## **Introduction**

This project was chosen to fill a need. Performing band Small City Blues needed a way to control their stage lights. These stage lights consist of 2 Microh LED Bar II. These lights consist of red, green, and blue LEDs that can be controlled to display a certain colour or a certain colour chase. With the knowledge gained in the class Computer Science 207, Building Interactive Gadgets, it is possible to construct a device to do such a task. Using slide potentiometers, buttons, OLED micro breakout screen, MAX 485 integrated circuit, XLR connector, and a resistor as main electrical components, a console can be assembled. This console can than be utilized by SCB for use in their live performances.

## **Background**

On the market there are devices that can control stage lights that musicians and other performers utilize are called DMX consoles. These consoles can control chase modes, colour selection, and colour intensity. The desired result was to create a device that could handle some of the simple functions that are available on commercial consoles. On more advanced DMX controllers it is possible to control motion. This is not necessary for the application of this console because SCB does not own any lights capable of motion.

Originally, the project was based off using [blog.deskontrol.com](http://blog.deskontrol.com) as a resource for constructing a small and simple dmx controller to control stage lights (Exhibit #1)

This project contained a Fritzing diagram containing 6 slide potentiometers, 12 buttons, one MAX484 integrated circuit, XLR connector, and a 100ohm resistor. When this was assembled and all the parts were connected, It was found that the code provided was giving issues without changing anything. This project test was not successful in controlling the stage lights with the hardware setup suggested by Toni Merino from [deskontrol.com](http://deskontrol.com).

Secondly, the project required a hardware setup suggested by the DMX512 page on the Arduino learning website. this hardware setup responded to a library entitled DmxSimple from Tinker.it! . This library was what allowed this project to continue. Using the Fritzing diagram provided, and a sample code, allows control over the Microh LED bar II. dmxSimple allows you to write commands similar to that of analogwrite commands in basic Arduino language. Usually, this code requires the use of a shield for the hardware setup Tinker.it! . This shield directly plugs into the top of the Arduino. Instead, the basic wiring without the use of the shield was constructed. This allows for more control over the placement of the pins and spare parts can be utilized. Using this method, and the sample “FadeUp” code from [tinker.it!](http://tinker.it/), the lights responded and faded in a specified colour.

## Description the Design Process

For this project a screen was desired so it was possible to see what mode the chase was currently in. This would be the novel contribution to a slightly altered version of deskontrols console. The Micro OLED breakout from Sparkfun was chosen for the

screen. This display required 5 digital pins, leaving only 9 pins left for buttons. This display did not require the use of any analog pins because it was connected via SPI rather than I<sup>2</sup>C. The buttons, slide potentiometers, and the wires required were also purchased from the Sparkfun website.

The LED bar contains red, green, and blue LEDS that need to be controlled via chase or via manual colour selection. Considering these limitations, for the purposes of this project this console would contain only 4 slide potentiometers, 7 buttons, the OLED screen and the XLR output. The first slide potentiometer, or slider, will control the speed of the chases, and the second, third, and fourth slider will control the intensity of the red, green, and blue LEDS, respectively. The buttons are colour coded red, green, blue, white, yellow, and two black. It is desired that the red, green, and blue buttons turn on their lights respectively, the two black buttons will cycle up and down the chase modes, and the yellow will turn on and off all lights.

The four slide potentiometers needed to be wired to the analog in pins on the Arduino, as well as being connected to power and ground. This project used the pins that were called for in the hookup guide for the OLED breakout screen. This screen was therefore hooked up to Arduino digital pins 13, 11, 10, 9, and 8. The buttons were wired to Arduino pins 12, 7, 6, 5, 4, 2, and 1. The MAX485 integrated circuit was hooked up to Arduino pin 3, which is the standard called for in the DmxSimple sample codes. This integrated circuit was also hooked up to power, ground, and the pins of the

XLR connector. A more detailed Fritzing diagram can be seen in exhibit #2. A picture of the final hardware setup in the shoebox is also included in Exhibit #5.

## Description of the Build Process

With the Desired functions and components in mind, the project code was based off code supplied in lab as well as from [tinker.it!](#) and sparkfun. This base code helped initialize the components as well as give a direction for the rest of the sketch. All of the code for this project can be seen in Exhibit #7.

First all of the hardware components were initialized and the DmxSimple, SPI, and Wire libraries were included. After all of the hardware was initialized, the modes for all the buttons were initialized. The second, third, and fourth slider from the left were then mapped to adjust the intensity of the LEDs. The values for this intensity could vary from 0 to 255.

if statements were then created to cycle through the modes for the black buttons. One if statement was created to cycle the Chase mode higher and one was to cycle it in the reverse. These modes are commanded using the DmxSimple.write command that functions like an analogwrite command. For example to display these chase modes you need to turn on the channel and input an applicable value for a chase. For these chases Dmx.Simple.write ( 1, 40-255 ) are applicable values. The mode for the chases is entitled ledMode in the code. These modes are created using if

and else statements. Each mode for the black buttons remapped the first slider so this would adjust the speed of the chase. The higher the number in the applicable values the faster the chase would run. By mapping the slider for each different mode, the same slider was able to be used for all the chase modes. Also in each of these chase modes or ledModes the program was to clear the screen and then display the new mode number. This was done for all of these modes. There was a total of 16 modes including an “off” mode to kill the lights.

For the red, green, and blue buttons the code used if and else statements and each mode was labeled appropriately. These if and else statements were done exactly like the chase select modes. There was only three modes for each of these buttons. First mode was to do nothing, or lights off, second mode was to turn on one specified colour, and the third mode was to turn on 2 specified colours to create a new colour. These buttons do not effect the OLED screen. For red the channel is 2, for green the channel is 3, and for blue the channel is 4.

The yellow button had only 2 modes. The first mode was to do nothing, or turn on the lights, and the second mode was to turn off all lights. This yellow button is like a master kill switch. This button does also not effect the OLED Screen. This button will need to control channels 1-4.

## Description of How the Project Works

By simply connecting the USB cord to power and the XLR output to the Microh LED Bar II and placing the light in the “slave mode” setting A001 or P002, you are able to control these lights. Once everything is connected the default mode on A001 light will be the first mode of chases. This is the colour fade chase. To cycle through the 16 different chase modes, press either the top black button to cycle higher or the bottom black button to cycle lower. Each mode is listed in Exhibit #3. These chase speeds can be adjusted using the first slider from the left. These chase modes are only visible when the light is set to “slave mode” A001.

When the light is in “slave mode” P002, the red, green, blue, and white buttons are able to control respective LEDs. When the red button is pressed the red LEDs turn on. If it is pressed a second time both the red and green LEDs will turn on displaying a yellow colour. If it is pressed a third time, the red and the green LEDs will turn off. When the green button is pressed the green LEDs turn on. If it is pressed again, the green and blue LEDs turn on displaying a teal colour. If it is pressed a third time, both the blue and green LEDs will turn off. If the blue button is pressed, the blue LEDs will turn on. If this button is pressed a second time, the blue and red LEDs will turn on displaying a purple colour. If the white button is pressed, all the LEDs will turn on and the intensity of each set of LEDs will be controlled by a slider. The second slider will control the intensity of the red LEDs when the white button has been pressed. The third slider will control the intensity of the green LEDs when the white button has been pressed. The fourth, and last, slider from the left will control the intensity of the blue

LEDs when the white button has been pressed. The default mode of the LEDs will be off.

Lastly, when the yellow button is pressed, all of the lights will turn off. This yellow mode will turn off lights in both “slave mode” A001 and P002. This means there will be nothing displayed from the LED light bars. Once the yellow button is pressed again, the lights will display the mode that was in before the first press on both “slave modes” A001 and P002. A picture of the finished product can be viewed in Exhibit #6.

## **Discussion of Milestones and the Project’s Future.**

The milestones of this project are described in detail in Exhibit #4. The first 2 milestones were completed successfully and on time. All the components were purchased, delivered, and assembled by Nov 8. Unfortunately, when using the schematic and the code created by Toni Merino from [blog.deskontrol.com](http://blog.deskontrol.com), the hardware was unable to control the light bars. This pushed back all of the subsequent milestones and the project was behind schedule. After using the schematic provided by the Arduino learning site, and the sample code from [tinker.it](http://tinker.it)! The lights were responding to the commands sent by the Arduino. Once this milestone was complete the rebuilding to incorporate the OLED screen was started. The rebuilt hardware was fitted into a shoebox to make it easier to manipulate the inputs of the console. The code was then inputed to manipulate the lights as described earlier in this paper.

Due to the issue with the original schematic and code from [blog.deskontrol.com](http://blog.deskontrol.com), the project was pushed back. Without the extra time, a 3D printed case was not created to house all the console components. In the future this would be beneficial to give the console a more practical and cleaner appearance. Also in the future the use of a piezo speaker to capture sound input using one of the left over analog Arduino pins. This sound input can be used to change the light colour based on intensity of sound and etc. This could also be used as a trigger so these lights can be changed using non conventional ways that will make it more practical for the member of Small City Blues to use the Console while performing.

By connecting the Screen using I<sup>2</sup>C instead of SPI would free up some digital pins that could be utilized by buttons. These buttons could be utilized so each other button has a more specific command. With four more buttons these could be assigned the secondary colours of the first buttons instead of cycling through modes. This would allow for more specific and quicker control over what colours are displayed. This will greatly improve the functionality of the console.

## References

DMX512. (n.d.). Retrieved December 18, 2017, from <http://playground.arduino.cc/Learning/DMX>

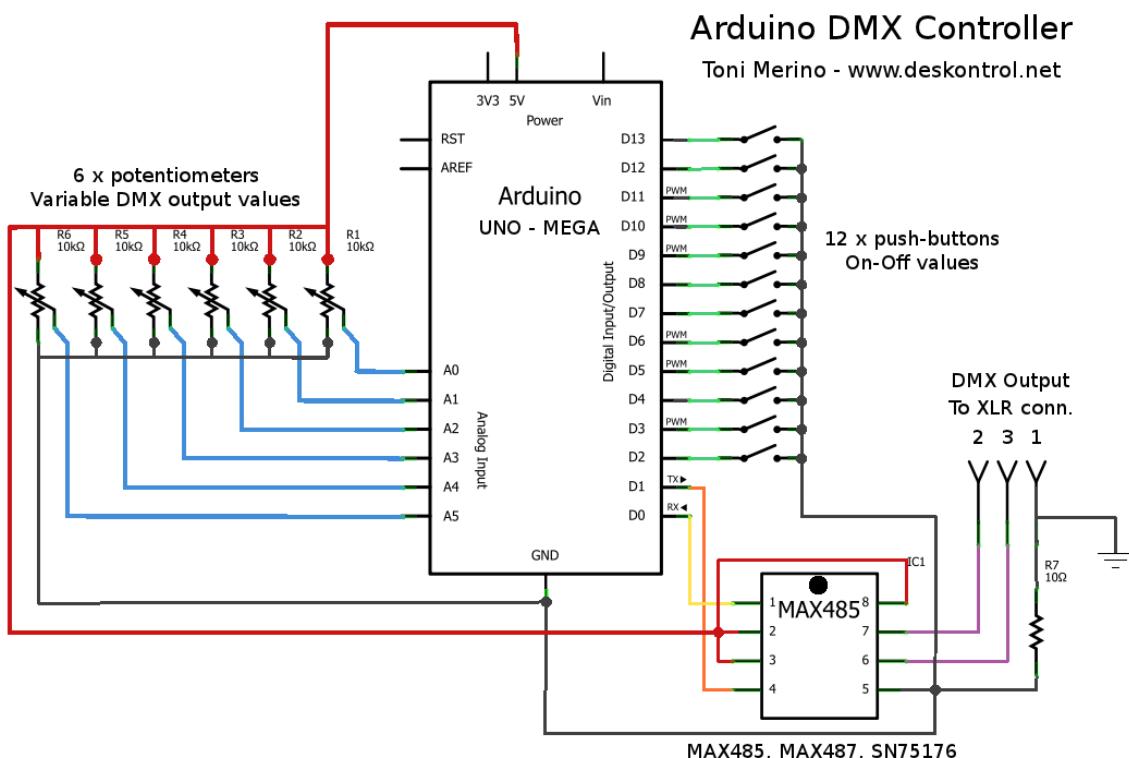
DmxSimple.wiki. (n.d.). Retrieved December 18, 2017, from <https://code.google.com/archive/p/tinkerit/wikis/DmxSimple.wiki>

Fritzing. (n.d.). Retrieved December 18, 2017, from <http://fritzing.org/projects/arduino-to-dmx-converter/>

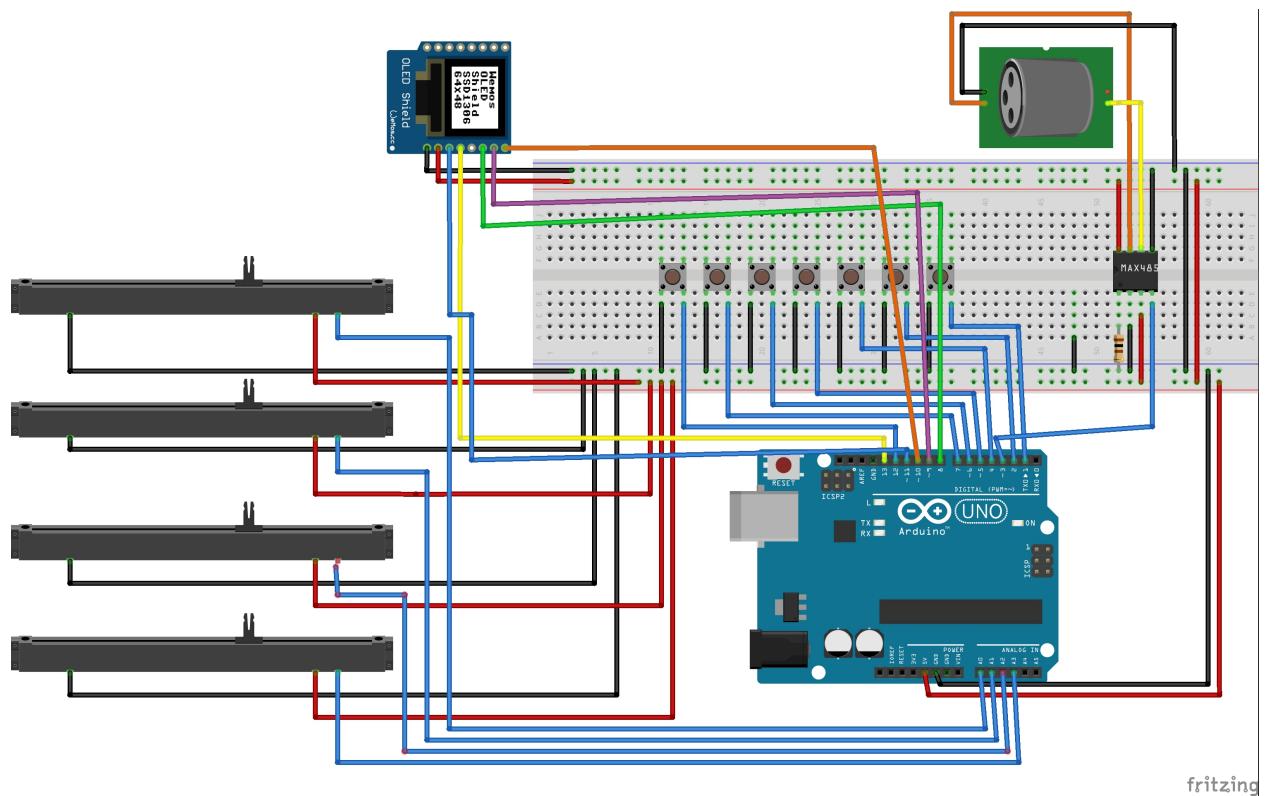
Micro OLED Breakout Hookup Guide . (n.d.). Retrieved December 18, 2017, from [https://learn.sparkfun.com/tutorials/micro-oled-breakout-hookup-guide?\\_ga=2.34733154.2147007682.1513586703-985330741.1510197581&\\_gac=1.219336171.1513212801.CjwKCAiAvMPRBRBIEiwABuO6qcwNu5XCb2tLtuwHKEvbfsQ4PIzmR4OMIHKVwDj8eIm54Uospj\\_dEhoCL58QAvD\\_BwE](https://learn.sparkfun.com/tutorials/micro-oled-breakout-hookup-guide?_ga=2.34733154.2147007682.1513586703-985330741.1510197581&_gac=1.219336171.1513212801.CjwKCAiAvMPRBRBIEiwABuO6qcwNu5XCb2tLtuwHKEvbfsQ4PIzmR4OMIHKVwDj8eIm54Uospj_dEhoCL58QAvD_BwE)

## Appendix

### Exhibit #1



## Exhibit #2



## **Exhibit #3**

### Chase Modes

<b>Mode Number</b>	<b>Chase Mode</b>	<b>Applicable Values</b>
1	Colour Fade	40-69
2	Cloud Flash	70-99
3	Purple Chase	100-111
4	Red Chase	112-123
5	Green and Red Chase	124-135
6	Blue and Green Chase	136-147
7	Yellow and Purple Chase	148-159
8	Yellow and Blue Chase	160-171
9	Pink and Green Chase	172-183
10	Red and Green Chase	184-195
11	Blue Chase	196-207
12	Multi Colour Chase	208-219
13	Multi Colour Chase	220-231
14	Multi Colour Chase	231-243
15	Multi Colour Chase	244-255
16	Off	0

## Exhibit #4

### MILESTONES

1	November 6	All Parts Gathered
2	November 8	Assembling the Components
3	November 10	Working DMX Console as described by <a href="http://blog.deskontrol.com">blog.deskontrol.com</a>
4	November 15	Adjust DMX Console to incorporate a screen
5	November 16	Working DMX console with Screen
6	November 22	Stretch Goal: 3D printed casing for DMX console Components instead of shoebox

## Exhibit #5



## Exhibit #6



## Exhibit #7

```
/*
```

```
Shoe box DMX console
```

```
control LED DMX stage lights with buttons and sliders using dmxSimple protocols and max485 intergrated curcuit.  
- displays mode number on screen.
```

```
code modified from lab code created by
```

```
Alex Clarke
```

```
and example code from spark from micro Oled breakout screen
```

```
modified by
```

```
Adam Hoffart
```

```
DEC 2017
```

```
*/
```

```

#include <DmxSimple.h>
#include <SPI.h>
#include <Wire.h>
#include <SFE_MicroOLED.h>
///////////////////////
// MicroOLED Definition //
///////////////////////
#define PIN_RESET 9 // Connect RST to pin 9
#define PIN_DC 8 // Connect DC to pin 8
#define PIN_CS 10 // Connect CS to pin 10
#define DC_JUMPER 0

///////////////////////
// MicroOLED Object Declaration //
///////////////////////
MicroOLED oled(PIN_RESET, PIN_DC, PIN_CS); // SPI declaration
//MicroOLED oled(PIN_RESET, DC_JUMPER); // I2C declaration

int SCREEN_WIDTH = oled.getLCDWidth();
int SCREEN_HEIGHT = oled.getLCDHeight();

float curButton = HIGH; // mode up
float lastButton = HIGH;
const int button = 2;

float curButton2 = HIGH; // mode down
float lastButton2 = HIGH;
const int button2 = 1;

float curButton3 = HIGH; // red button
float lastButton3 = HIGH;
const int button3 = 4;

float curButton4 = HIGH; //green button
float lastButton4 = HIGH;
const int button4 = 12;

float curButton5 = HIGH; // blue button
float lastButton5 = HIGH;
const int button5 = 7;

float curButton6 = HIGH; // white button
float lastButton6 = HIGH;
const int button6 = 5;

float curButton7 = HIGH; // yellow button
float lastButton7 = HIGH;
const int button7 = 6;

int ledMode = 1; // chase mode cycle
int RedMode = 1; // red colour cycle
int GrnMode = 1; // green colour cycle
int BluMode = 1; // blue colour cycle
int WitMode = 1; // turn on all leds - control by slide2,3, and 4.
int YlwMode = 1; // turn off/on all lights

void setup() {
    // put your setup code here, to run once:
    DmxSimple.usePin(3);

    /* DMX devices typically need to receive a complete set of channels
     ** even if you only need to adjust the first channel. You can

```

```

** easily change the number of channels sent here. If you don't
** do this, DmxSimple will set the maximum channel number to the
** highest channel you DmxSimple.write() to. */
//DmxSimple.maxChannel();

Serial.begin(9600);

pinMode(button, INPUT);
pinMode(button2, INPUT);
pinMode(button3, INPUT);
pinMode(button4, INPUT);
pinMode(button5, INPUT);
pinMode(button6, INPUT);
pinMode(button7, INPUT);
// Trick to make this a pull up button
// Other side of button must be attached to ground
// This trick doesn't work on pin 13.
digitalWrite(button, HIGH);
digitalWrite(button2, HIGH);
digitalWrite(button3, HIGH);
digitalWrite(button4, HIGH);
digitalWrite(button5, HIGH);
digitalWrite(button6, HIGH);
digitalWrite(button7, HIGH);

oled.begin();
oled.clear(ALL);
oled.display();

oled.begin();

}

void loop() {

int slide1 = A0;
int slide2 = A1;
int slide3 = A2;
int slide4 = A3;

lastButton = curButton;
curButton = digitalRead(button);

lastButton2 = curButton2;
curButton2 = digitalRead(button2);

lastButton3 = curButton3;
curButton3 = digitalRead(button3);

lastButton4 = curButton4;
curButton4 = digitalRead(button4);

lastButton5 = curButton5;
curButton5 = digitalRead(button5);

lastButton6 = curButton6;
curButton6 = digitalRead(button6);

lastButton7 = curButton7;
curButton7 = digitalRead(button7);

//int spd= analogRead(slide1);
//spd = map(spd, 0, 1023, 244, 255); // mapping to the speed cycle of chase

```

```

int dim1 = analogRead(slide2);
dim1 = map(dim1, 0, 1023, 0, 255); // mapping the brightness of the red leds

int dim2 = analogRead(slide3);
dim2 = map(dim2, 0, 1023, 0, 255); // mapping the brightness of the green leds

int dim3 = analogRead(slide4);
dim3 = map(dim3, 0, 1023, 0, 255); // mapping the brightness of the blue leds

if (curButton == LOW && lastButton == HIGH)

{
    // if it is, count a press - cycle through chase modes
    ledMode = ledMode + 1;

    // if we've counted to an invalid mode
    if (ledMode == 17)
    {
        // reset to mode 1
        ledMode = 1;
    }
}

if (curButton2 == LOW && lastButton2 == HIGH)

{
    // if it is, count a press
    ledMode = ledMode - 1;

    // if we've counted to an invalid mode
    if (ledMode == 0)
    {
        // reset to mode 1
        ledMode = 16;
    }
}

// Detect the current mode and set the LEDs appropriately
if (ledMode == 1) // Mode 1 - colour fade
{
    int spd = analogRead(slide1);
    spd = map(spd, 0, 1023, 40, 69); // map chase speed - give values for colour fade chase
    DmxSimple.write (1, spd);

    oled.clear(PAGE); // clear screen
    oled.setFontType(1); // smaller font size
    oled.setCursor (0, 0); // set point for text on screen

    oled.print( "MODE #1"); // display mode #1
    //oled.println ("COLOUR FADE");
    oled.display(); // Draw to the screen
    //delay(1000);

}
else if (ledMode == 2)
{
    //Mode 2 - colour flash
    int spd = analogRead(slide1);
    spd = map(spd, 0, 1023, 70, 99); // map chase speed - give values for colour flash chase
}

```

```

DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #2"); /// display mode #2
//oled.println ("COLOUR FLASH");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 3)
{
//Mode 3 - purple chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 100, 111); // map chase speed - give values for purple chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #3"); // display mode #3
//oled.println ("PURPLE CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 4)
{
//Mode 4 - red chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 112, 123); // map chase speed - give values for red chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #4"); // display mode #4
//oled.println ("RED CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 5)
{
//Mode 5 - green and red chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 124, 135); // map chase speed - give values for green and red chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #5"); // display mode #5
//oled.println ("GRN+RED CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 6)
{
//Mode 6 - blue and green chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 136, 147); // map chase speed - give values for blue and green chase
}

```

```

DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #6"); /// display mode #6
//oled.println ("BLU+GRN CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 7)
{
//Mode 7 - yellow and purple chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 148, 159); // map chase speed - give values for yellow and purple chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #7"); // display mode #7
//oled.println ("YLW+PRP CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 8)
{
//Mode 8 - yellow and red chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 160, 171); // map chase speed - give values for yellow and red chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #8"); // display mode #8
//oled.println ("YLW+RED CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 9)
{
//Mode 9 - pink and green chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 172, 183); // map chase speed - give values for pink and green chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE #9"); // display mode #9
//oled.println ("PNK+GRN FLASH");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 10)
{
//Mode 10 - red and green chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 184, 195); // map chase speed - give values for red and green chase

```

```

DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE 10"); // display mode #10
//oled.println ("RED+GRN FLASH");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 11)
{
//Mode 11 - blue chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 196, 207); // map chase speed - give values for blue chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE 11"); // display mode #11
//oled.println ("BLUE CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 12)
{
//Mode 12 - multi colour chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 208, 219); // map chase speed - give values for multi colour chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE 12"); // display mode #12
//oled.println ("MLT CLR CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 13)
{
//Mode 13 - multi colour chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 220, 231); // map chase speed - give values for multi colour chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE 13"); // display mode #13
//oled.println ("MLT CLR CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}
else if (ledMode == 14)
{
//Mode 14 - multi colour chase
int spd = analogRead(slide1);

```

```

spd = map(spd, 0, 1023, 232, 243); // map chase speed - give values for multi colour chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE 14"); // display mode #14
//oled.println ("MLT CLR CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}

else if (ledMode == 15)
{
//Mode 15 - multi colour chase
int spd = analogRead(slide1);
spd = map(spd, 0, 1023, 244, 255); // map chase speed - give values for multi colour chase
DmxSimple.write (1, spd);

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (0, 0); // set point for text on screen

oled.print( "MODE 15"); // display mode #15
//oled.println ("MLT CLR CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}

else
{
//Mode 16 - off mode

DmxSimple.write (1, 0); // turn off chase mode

oled.clear(PAGE); // clear screen
oled.setFontType(1); // smaller font size
oled.setCursor (20, 0); // set point for text on screen - displays text in the middle

oled.print( "OFF" ); // display off
//oled.println ("MULTI CLR CHASE");
oled.display(); // Draw to the screen
//delay(1000);
}

if (curButton3 == LOW && lastButton3 == HIGH)

{
// if it is, count a press
RedMode = RedMode + 1;

// if //we've counted to an invalid mode
if (RedMode == 4)
{
//reset to mode 1
RedMode = 1;
}
}

if (RedMode == 1)
{

```

```

DmxSimple.write (3, 0); //turn off green LEDs
DmxSimple.write (4, 0); //turn off blue LEDs
DmxSimple.write (2, 0); //turn off red LEDs
}

else if (RedMode == 2)
{
    DmxSimple.write (2, 255); // turn on red LEDs
    DmxSimple.write (4, 0); // turn off blue LEDs
    DmxSimple.write (3, 0); // turn off green LEDs
}

else

{
    DmxSimple.write (2, 255); // turn on red LEDs
    DmxSimple.write (3, 255); // turn on green LEDs to make color yellow
    DmxSimple.write (4, 0); // turn off blue LEDs
}

if (curButton4 == LOW && lastButton4 == HIGH)

{
    // if it is, count a press
    GrnMode = GrnMode + 1;

    // if //we've counted to an invalid mode
    if (GrnMode == 4)
    {
        //reset to mode 1
        GrnMode = 1;
    }
}

if (GrnMode == 1)
{
    //DmxSimple.write (3, 0); // do nothing
    //DmxSimple.write (4, 0);
    //DmxSimple.write (2, 0);
}

else if (GrnMode == 2)

{
    DmxSimple.write (3, 255); // turn on green LEDs
    DmxSimple.write (4, 0); // turn off blue LEDs
    DmxSimple.write (2, 0); // turn off red LEDs
}

else

{
    DmxSimple.write (3, 255); // turn on green LEDs
    DmxSimple.write (4, 255); // turn on blue LEDs to make the colour teal
    DmxSimple.write (2, 0); // turn off red LEDs
}

if (curButton5 == LOW && lastButton5 == HIGH)

{
    // if it is, count a press
}

```

```

BluMode = BluMode + 1;

// if we've counted to an invalid mode
if (BluMode == 4)
{
    //reset to mode 1
    BluMode = 1;
}
}

if (BluMode == 1)
{
    //DmxSimple.write (3, 0); // do nothing
    // DmxSimple.write (4, 0);
    // DmxSimple.write (2, 0);

}

else if (BluMode == 2)
{
    DmxSimple.write (4, 255); // turn on blue LEDs
    DmxSimple.write (2, 0); // turn off red LEDs
    DmxSimple.write (3, 0); // turn off green LEDs
}

else
{
    DmxSimple.write (4, 255); // turn on blue LEDs
    DmxSimple.write (2, 255); // turn on red LEDs to make colour purple
    DmxSimple.write (3, 0); // turn off green LEDs
}

if (curButton6 == LOW && lastButton6 == HIGH)

{
    // if it is, count a press
    WitMode = WitMode + 1;

    // if we've counted to an invalid mode
    if (WitMode == 3)
    {
        //reset to mode 1
        WitMode = 1;
    }
}

if (WitMode == 1)

{
    //DmxSimple.write (3, 0); // do nothing
    // DmxSimple.write (4, 0);
    // DmxSimple.write (2, 0);
}

else // if (WitMode == 2)

{
    DmxSimple.write (2, dim1); // turn on red LEDs and fade intensity with slide 2
    DmxSimple.write (3, dim2); // turn on red LEDs and fade intensity with slide 2
    DmxSimple.write (4, dim3); // turn on red LEDs and fade intensity with slide 2
    delay(10);
}

```

```

// else
{
    // DmxSimple.write (3, dim2); // do nothing
    // DmxSimple.write (4, dim3);
    // DmxSimple.write (2, dim1);
}

if (curButton7 == LOW && lastButton7 == HIGH)

{
    // if it is, count a press
    YlwMode = YlwMode + 1;

    // if we've counted to an invalid mode
    if (YlwMode == 3)
    {
        //reset to mode 1
        YlwMode = 1;
    }
}

if (YlwMode == 1) // do nothing
{
    //DmxSimple.write (3, dim2);
    //DmxSimple.write (4, dim3 );
    //DmxSimple.write (2, dim1);
}

else

{
    DmxSimple.write (3, 0); // turn off green LEDs
    DmxSimple.write (4, 0); // turn off blue LEDs
    DmxSimple.write (2, 0); // turn off red LEDs
    DmxSimple.write (1, 0); // turn off chase modes
    delay(10);
}
}

```