/\*

\* Nick Newcomb

\* 02/07/2020

\* Project\_007F

\* This revision of LightFight Mini supports added interrupts (4 interrupts)

\*/

//Control Lights

//NeoPixel Stuff

//create a NeoPixel strip

#include <WS2811Driver.h>

//create a NeoPixel strip

WS2811Driver strip = WS2811Driver(4, P1\_3, NEO\_GRB);

unsigned int randNum = random(1,4);

int count = 0;

int interval = 2000;

unsigned long referenceTime = 0;

unsigned long currentTime = 0;

//Buttons

int buttonInt1 = P2\_0;

int buttonInt2 = P2\_1;

int buttonInt3 = P2\_2;

int buttonInt4 = P2\_3;

int buttonInt5 = P2\_4;

int buttonInt6 = P2\_5;

//Progress Lights1

WS2811Driver prog\_strip = WS2811Driver(10, P1\_4, NEO\_GRB);

int pState = 0;

volatile int winCount=0;

//Progress Lights2

//create a NeoPixel strip

WS2811Driver prog\_strip2 = WS2811Driver(10, P1\_5, NEO\_GRB);

int pState2 = 0;

volatile int winCount2=0;

//Functions

//Player1

//

//

//

void swap1()

{

if (randNum == 1 && pState == 0)

{

winCount = winCount + 1;

pState=1;

}

else

{

if(winCount > 0 && winCount<5)

{

winCount= winCount - 1;

}

else if(winCount == 5)

{

winCount= 5;

}

if(winCount > 5 && winCount<10)

{

winCount= winCount - 1;

}

else if(winCount == 10)

{

winCount= 10;

}

else if(winCount > 10 && winCount < 15)

{

winCount= winCount -1;

}

else if(winCount == 15)

{

winCount= 15;

}

else if(winCount > 15 && winCount < 20)

{

winCount= winCount -1;

}

else if(winCount == 20)

{

winCount= 20;

}

else if(winCount > 20 && winCount < 25)

{

winCount= winCount -1;

}

else if(winCount == 25)

{

winCount= 25;

}

else if(winCount > 25 && winCount < 30)

{

winCount= winCount -1;

}

else if(winCount == 30)

{

winCount= 30;

}

else if(winCount > 35 && winCount < 40)

{

winCount= winCount -1;

}

else if(winCount == 40)

{

winCount= 40;

}

else if(winCount > 40 && winCount < 50)

{

winCount= winCount -1;

}

else if(winCount == 0)

{

winCount= 0;

}

}

}

void swap2()

{

if (randNum == 2 && pState == 0)

{

winCount = winCount + 1;

pState=1;

}

else

{

if(winCount > 0 && winCount<5)

{

winCount= winCount - 1;

}

else if(winCount == 5)

{

winCount= 5;

}

if(winCount > 5 && winCount<10)

{

winCount= winCount - 1;

}

else if(winCount == 10)

{

winCount= 10;

}

else if(winCount > 10 && winCount < 15)

{

winCount= winCount -1;

}

else if(winCount == 15)

{

winCount= 15;

}

else if(winCount > 15 && winCount < 20)

{

winCount= winCount -1;

}

else if(winCount == 20)

{

winCount= 20;

}

else if(winCount > 20 && winCount < 25)

{

winCount= winCount -1;

}

else if(winCount == 25)

{

winCount= 25;

}

else if(winCount > 25 && winCount < 30)

{

winCount= winCount -1;

}

else if(winCount == 30)

{

winCount= 30;

}

else if(winCount > 35 && winCount < 40)

{

winCount= winCount -1;

}

else if(winCount == 40)

{

winCount= 40;

}

else if(winCount > 40 && winCount < 50)

{

winCount= winCount -1;

}

else if(winCount == 0)

{

winCount= 0;

}

}

}

void swap3()

{

if (randNum == 3 && pState == 0)

{

winCount = winCount + 1;

pState=1;

}

else

{

if(winCount > 0 && winCount<5)

{

winCount= winCount - 1;

}

else if(winCount == 5)

{

winCount= 5;

}

if(winCount > 5 && winCount<10)

{

winCount= winCount - 1;

}

else if(winCount == 10)

{

winCount= 10;

}

else if(winCount > 10 && winCount < 15)

{

winCount= winCount -1;

}

else if(winCount == 15)

{

winCount= 15;

}

else if(winCount > 15 && winCount < 20)

{

winCount= winCount -1;

}

else if(winCount == 20)

{

winCount= 20;

}

else if(winCount > 20 && winCount < 25)

{

winCount= winCount -1;

}

else if(winCount == 25)

{

winCount= 25;

}

else if(winCount > 25 && winCount < 30)

{

winCount= winCount -1;

}

else if(winCount == 30)

{

winCount= 30;

}

else if(winCount > 35 && winCount < 40)

{

winCount= winCount -1;

}

else if(winCount == 40)

{

winCount= 40;

}

else if(winCount > 40 && winCount < 50)

{

winCount= winCount -1;

}

else if(winCount == 0)

{

winCount= 0;

}

}

}

//Player Two

//

//

void swap4()

{

if (randNum == 1 && pState2 == 0)

{

winCount2 = winCount2 + 1;

pState2=1;

}

else

{

if(winCount2 > 0 && winCount2<5)

{

winCount2= winCount2 - 1;

}

else if(winCount2 == 5)

{

winCount2= 5;

}

if(winCount2 > 5 && winCount2<10)

{

winCount2= winCount2 - 1;

}

else if(winCount2 == 10)

{

winCount2= 10;

}

else if(winCount2 > 10 && winCount2 < 15)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 15)

{

winCount2= 15;

}

else if(winCount2 > 15 && winCount2 < 20)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 20)

{

winCount2= 20;

}

else if(winCount2 > 20 && winCount2 < 25)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 25)

{

winCount2= 25;

}

else if(winCount2 > 25 && winCount2 < 30)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 30)

{

winCount2= 30;

}

else if(winCount2 > 35 && winCount2 < 40)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 40)

{

winCount2= 40;

}

else if(winCount2 > 40 && winCount2 < 50)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 0)

{

winCount2= 0;

}

}

}

void swap5()

{

if (randNum == 2 && pState2 == 0)

{

winCount2 = winCount2 + 1;

pState2=1;

}

else

{

if(winCount2 > 0 && winCount2<5)

{

winCount2= winCount2 - 1;

}

else if(winCount2 == 5)

{

winCount2= 5;

}

if(winCount2 > 5 && winCount2<10)

{

winCount2= winCount2 - 1;

}

else if(winCount2 == 10)

{

winCount2= 10;

}

else if(winCount2 > 10 && winCount2 < 15)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 15)

{

winCount2= 15;

}

else if(winCount2 > 15 && winCount2 < 20)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 20)

{

winCount2= 20;

}

else if(winCount2 > 20 && winCount2 < 25)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 25)

{

winCount2= 25;

}

else if(winCount2 > 25 && winCount2 < 30)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 30)

{

winCount2= 30;

}

else if(winCount2 > 35 && winCount2 < 40)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 40)

{

winCount2= 40;

}

else if(winCount2 > 40 && winCount2 < 50)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 0)

{

winCount2= 0;

}

}

}

void swap6()

{

if (randNum == 3 && pState2 == 0)

{

winCount2 = winCount2 + 1;

pState2=1;

}

else

{

if(winCount2 > 0 && winCount2<5)

{

winCount2= winCount2 - 1;

}

else if(winCount2 == 5)

{

winCount2= 5;

}

if(winCount2 > 5 && winCount2<10)

{

winCount2= winCount2 - 1;

}

else if(winCount2 == 10)

{

winCount2= 10;

}

else if(winCount2 > 10 && winCount2 < 15)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 15)

{

winCount2= 15;

}

else if(winCount2 > 15 && winCount2 < 20)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 20)

{

winCount2= 20;

}

else if(winCount2 > 20 && winCount2 < 25)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 25)

{

winCount2= 25;

}

else if(winCount2 > 25 && winCount2 < 30)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 30)

{

winCount2= 30;

}

else if(winCount2 > 35 && winCount2 < 40)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 40)

{

winCount2= 40;

}

else if(winCount2 > 40 && winCount2 < 50)

{

winCount2= winCount2 -1;

}

else if(winCount2 == 0)

{

winCount2= 0;

}

}

}

void setup() {

//NEOPIXEL

// start the strip and blank it out

strip.begin();

strip.show();

prog\_strip.begin();

prog\_strip.show();

prog\_strip2.begin();

prog\_strip2.show();

//Make it more random

Serial.begin(9600);

randomSeed(analogRead(P2\_7));

//Button Test Setup

attachInterrupt(buttonInt1, swap1, FALLING);

attachInterrupt(buttonInt2, swap2, FALLING);

attachInterrupt(buttonInt3, swap3, FALLING);

attachInterrupt(buttonInt4, swap4, FALLING);

attachInterrupt(buttonInt5, swap5, FALLING);

attachInterrupt(buttonInt6, swap6, FALLING);

}

void loop() {

currentTime = millis();

Serial.println(currentTime);

if (randNum == 1 && count < interval)

{

for(int i=0; i < 4; i++){

strip.setPixelColor(i, 63, 0, 0);

}

strip.show();

currentTime = millis();

count = currentTime-referenceTime;

if (pState == 1)

{

switch (winCount)

{

case 0 ... 4:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 5 ... 9:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 10 ... 14:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 15 ... 19:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 20 ... 24:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 25 ... 29:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 30 ... 34:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 35 ... 39:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 40 ... 50:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 0, 63);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

}

}

if (pState2 == 1)

{

currentTime = millis();

switch (winCount2)

{

case 0 ... 4:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 5 ... 9:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 10 ... 14:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 15 ... 19:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 20 ... 24:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 25 ... 29:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 30 ... 34:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 35 ... 39:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 40 ... 50:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 0, 63);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

}

}

}

else if (randNum == 2 && count < interval)

{

for(int i=0; i < 4; i++){

strip.setPixelColor(i, 0, 63, 0);

}

strip.show();

currentTime = millis();

count = currentTime-referenceTime;

if (pState == 1)

{

switch (winCount)

{

case 0 ... 4:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 5 ... 9:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 10 ... 14:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 15 ... 19:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 20 ... 24:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 25 ... 29:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 30 ... 34:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 35 ... 39:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 40 ... 50:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 0, 63);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

}

}

if (pState2 == 1)

{

currentTime = millis();

switch (winCount2)

{

case 0 ... 4:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 5 ... 9:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 10 ... 14:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 15 ... 19:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 20 ... 24:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 25 ... 29:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 30 ... 34:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 35 ... 39:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 40 ... 50:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 0, 63);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

}

}

}

else if (randNum == 3 && count < interval)

{

for(int i=0; i < 4; i++){

strip.setPixelColor(i, 0, 0, 63);

}

strip.show();

currentTime = millis();

count = currentTime-referenceTime;

if (pState == 1)

{

switch (winCount)

{

case 0 ... 4:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 5 ... 9:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 10 ... 14:

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 15 ... 19:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 20 ... 24:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.show();

currentTime = millis();

break;

case 25 ... 29:

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 30 ... 34:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 35 ... 39:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

case 40 ... 50:

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 0, 63);

prog\_strip.show();

delay(75);

pState=0;

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

currentTime = millis();

break;

}

}

if (pState2 == 1)

{

switch (winCount2)

{

case 0 ... 4:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 5 ... 9:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 10 ... 14:

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 15 ... 19:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 20 ... 24:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 25 ... 29:

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 30 ... 34:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 35 ... 39:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

case 40 ... 50:

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 0, 63);

prog\_strip2.show();

delay(75);

pState2=0;

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

currentTime = millis();

break;

}

}

}

else if ( count >= interval)

{

count = 0;

referenceTime = currentTime;

randNum = random(1,4);

}

if(winCount == 50)

{

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 0, 63);

prog\_strip.show();

while(1)

{

prog\_strip.setPixelColor(0, 63, 0, 0);

prog\_strip.setPixelColor(1, 63, 0, 0);

prog\_strip.setPixelColor(2, 63, 0, 0);

prog\_strip.show();

delay(75);

prog\_strip.setPixelColor(0, 0, 63, 0);

prog\_strip.setPixelColor(1, 0, 63, 0);

prog\_strip.setPixelColor(2, 0, 63, 0);

prog\_strip.show();

delay(75);

prog\_strip.setPixelColor(0, 0, 0, 63);

prog\_strip.setPixelColor(1, 0, 0, 63);

prog\_strip.setPixelColor(2, 0, 0, 63);

prog\_strip.show();

delay(75);

}

}

if(winCount2 == 50)

{

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 0, 63);

prog\_strip2.show();

while(1)

{

prog\_strip2.setPixelColor(0, 63, 0, 0);

prog\_strip2.setPixelColor(1, 63, 0, 0);

prog\_strip2.setPixelColor(2, 63, 0, 0);

prog\_strip2.show();

delay(75);

prog\_strip2.setPixelColor(0, 0, 63, 0);

prog\_strip2.setPixelColor(1, 0, 63, 0);

prog\_strip2.setPixelColor(2, 0, 63, 0);

prog\_strip2.show();

delay(75);

prog\_strip2.setPixelColor(0, 0, 0, 63);

prog\_strip2.setPixelColor(1, 0, 0, 63);

prog\_strip2.setPixelColor(2, 0, 0, 63);

prog\_strip2.show();

delay(75);

}

}

}