#include <Adafruit\_NeoPixel.h>

#define PIN 12 // On Trinket or Gemma, suggest changing this to 1

#define NUMPIXELS 30 // Popular NeoPixel ring size

Adafruit\_NeoPixel pixels(NUMPIXELS, PIN, NEO\_GRB + NEO\_KHZ800);

#define DELAYVAL 500 // Time (in milliseconds) to pause between pixels

int SwitchPin1 = 13;

int SwitchPin1State = 0;

int SwitchPin1StatePrev = 0;

int SwitchPin2 = 2;

int SwitchPin2State = 0;

int SwitchPin2StatePrev = 0;

int SwitchPin3 = 3;

int SwitchPin3State = 0;

int SwitchPin3StatePrev = 0;

int SwitchPin4 = 4;

int SwitchPin4State = 0;

int SwitchPin4StatePrev = 0;

int SwitchPin5 = 5;

int SwitchPin5State = 0;

int SwitchPin5StatePrev = 0;

int SwitchPin6 = 6;

int SwitchPin6State = 0;

int SwitchPin6StatePrev = 0;

int SwitchPin7 = 7;

int SwitchPin7State = 0;

int SwitchPin7StatePrev = 0;

int SwitchPin8 = 8;

int SwitchPin8State = 0;

int SwitchPin8StatePrev = 0;

int SwitchPin9 = 9;

int SwitchPin9State = 0;

int SwitchPin9StatePrev = 0;

int SwitchPin10 = 10;

int SwitchPin10State = 0;

int SwitchPin10StatePrev = 0;

int SwitchPinCount = 0;

const int buttonPin = 11;

int buttonStatePrev = 0;

int state = 1;

int buttonState = 0;

unsigned long changeTime;

unsigned long intervalTime = 5000;

//light show

int RbArrayR[] = {148, 75, 0, 0, 255, 255, 255};

int RbArrayG[] = {0, 0, 0, 255, 255, 127, 0};

int RbArrayB[] = {211, 130, 255, 0, 0, 0, 0};

int pixelIndex = 0;

int colorIndex = 0;

int factor = 0;

void setup() {

Serial.begin(9600);

pinMode(12, OUTPUT);// initialize neopixel pin as output

pinMode(buttonPin, INPUT);// initialize pushbutton pin as input

pinMode(SwitchPin1, INPUT);

pinMode(SwitchPin2, INPUT);

pinMode(SwitchPin3, INPUT);

pinMode(SwitchPin4, INPUT);

pinMode(SwitchPin5, INPUT);

pinMode(SwitchPin6, INPUT);

pinMode(SwitchPin7, INPUT);

pinMode(SwitchPin8, INPUT);

pinMode(SwitchPin9, INPUT);

pinMode(SwitchPin10, INPUT);

changeTime = millis();

pixels.begin(); // Initialize NeoPixel strip

pixels.setBrightness(50);

}

void loop() {

buttonStatePrev = buttonState; //Store the previous button state

buttonState = digitalRead(buttonPin);// read the state of the pushbutton value

SwitchPin1StatePrev = SwitchPin1State;

SwitchPin1State = digitalRead(SwitchPin1);

SwitchPin2StatePrev = SwitchPin2State;

SwitchPin2State = digitalRead(SwitchPin2);

SwitchPin3StatePrev = SwitchPin3State;

SwitchPin3State = digitalRead(SwitchPin3);

SwitchPin4StatePrev = SwitchPin4State;

SwitchPin4State = digitalRead(SwitchPin4);

SwitchPin5StatePrev = SwitchPin5State;

SwitchPin5State = digitalRead(SwitchPin5);

SwitchPin6StatePrev = SwitchPin6State;

SwitchPin6State = digitalRead(SwitchPin6);

SwitchPin7StatePrev = SwitchPin7State;

SwitchPin7State = digitalRead(SwitchPin7);

SwitchPin8StatePrev = SwitchPin8State;

SwitchPin8State = digitalRead(SwitchPin8);

SwitchPin9StatePrev = SwitchPin9State;

SwitchPin9State = digitalRead(SwitchPin9);

SwitchPin10StatePrev = SwitchPin10State;

SwitchPin10State = digitalRead(SwitchPin10);

switch (state) {

case 1: //game begins with all neopixels yellow, until satisfied with cups

if(SwitchPin1State == LOW){

pixels.setPixelColor(0, pixels.Color(0,255,0)); //green, cup detected

pixels.setPixelColor(1, pixels.Color(0,255,0));

pixels.setPixelColor(2, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(0, pixels.Color(255,255,0));//yellow, no cup detected

pixels.setPixelColor(1, pixels.Color(255,255,0));

pixels.setPixelColor(2, pixels.Color(255,255,0));

}

if(SwitchPin2State == LOW){

pixels.setPixelColor(3, pixels.Color(0,255,0));

pixels.setPixelColor(4, pixels.Color(0,255,0));

pixels.setPixelColor(5, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(3, pixels.Color(255,255,0));

pixels.setPixelColor(4, pixels.Color(255,255,0));

pixels.setPixelColor(5, pixels.Color(255,255,0));

}

if(SwitchPin3State == LOW){

pixels.setPixelColor(6, pixels.Color(0,255,0));

pixels.setPixelColor(7, pixels.Color(0,255,0));

pixels.setPixelColor(8, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(6, pixels.Color(255,255,0));

pixels.setPixelColor(7, pixels.Color(255,255,0));

pixels.setPixelColor(8, pixels.Color(255,255,0));

}

if(SwitchPin4State == LOW){

pixels.setPixelColor(9, pixels.Color(0,255,0));

pixels.setPixelColor(10, pixels.Color(0,255,0));

pixels.setPixelColor(11, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(9, pixels.Color(255,255,0));

pixels.setPixelColor(10, pixels.Color(255,255,0));

pixels.setPixelColor(11, pixels.Color(255,255,0));

}

if(SwitchPin5State == LOW){

pixels.setPixelColor(12, pixels.Color(0,255,0));

pixels.setPixelColor(13, pixels.Color(0,255,0));

pixels.setPixelColor(14, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(12, pixels.Color(255,255,0));

pixels.setPixelColor(13, pixels.Color(255,255,0));

pixels.setPixelColor(14, pixels.Color(255,255,0));

}

if(SwitchPin6State == LOW){

pixels.setPixelColor(15, pixels.Color(0,255,0));

pixels.setPixelColor(16, pixels.Color(0,255,0));

pixels.setPixelColor(17, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(15, pixels.Color(255,255,0));

pixels.setPixelColor(16, pixels.Color(255,255,0));

pixels.setPixelColor(17, pixels.Color(255,255,0));

}

if(SwitchPin7State == LOW){

pixels.setPixelColor(18, pixels.Color(0,255,0));

pixels.setPixelColor(19, pixels.Color(0,255,0));

pixels.setPixelColor(20, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(18, pixels.Color(255,255,0));

pixels.setPixelColor(19, pixels.Color(255,255,0));

pixels.setPixelColor(20, pixels.Color(255,255,0));

}

if(SwitchPin8State == LOW){

pixels.setPixelColor(21, pixels.Color(0,255,0));

pixels.setPixelColor(22, pixels.Color(0,255,0));

pixels.setPixelColor(23, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(21, pixels.Color(255,255,0));

pixels.setPixelColor(22, pixels.Color(255,255,0));

pixels.setPixelColor(23, pixels.Color(255,255,0));

}

if(SwitchPin9State == LOW){

pixels.setPixelColor(24, pixels.Color(0,255,0));

pixels.setPixelColor(25, pixels.Color(0,255,0));

pixels.setPixelColor(26, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(24, pixels.Color(255,255,0));

pixels.setPixelColor(25, pixels.Color(255,255,0));

pixels.setPixelColor(26, pixels.Color(255,255,0));

}

if(SwitchPin10State == LOW){

pixels.setPixelColor(27, pixels.Color(0,255,0));

pixels.setPixelColor(28, pixels.Color(0,255,0));

pixels.setPixelColor(29, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(27, pixels.Color(255,255,0));

pixels.setPixelColor(28, pixels.Color(255,255,0));

pixels.setPixelColor(29, pixels.Color(255,255,0));

}

pixels.show();

SwitchPinCount = SwitchPin1State + SwitchPin2State +

SwitchPin3State + SwitchPin4State +

SwitchPin5State + SwitchPin6State +

SwitchPin7State + SwitchPin8State +

SwitchPin9State + SwitchPin10State;

Serial.println(" in State 1 ");

if (SwitchPinCount == 0){

Serial.println(" 10 cups detected ");

state = 2;

break;

}

break;

case 2: // game begins once the cups are on the board therefore turning green

if(SwitchPin1State == LOW){

pixels.setPixelColor(0, pixels.Color(0,255,0)); //green, cup detected

pixels.setPixelColor(1, pixels.Color(0,255,0));

pixels.setPixelColor(2, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(0, pixels.Color(255,0,0));//red, no cup detected

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

if(SwitchPin2State == LOW){

pixels.setPixelColor(3, pixels.Color(0,255,0));

pixels.setPixelColor(4, pixels.Color(0,255,0));

pixels.setPixelColor(5, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(3, pixels.Color(255,0,0));

pixels.setPixelColor(4, pixels.Color(255,0,0));

pixels.setPixelColor(5, pixels.Color(255,0,0));

}

if(SwitchPin3State == LOW){

pixels.setPixelColor(6, pixels.Color(0,255,0));

pixels.setPixelColor(7, pixels.Color(0,255,0));

pixels.setPixelColor(8, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(6, pixels.Color(255,0,0));

pixels.setPixelColor(7, pixels.Color(255,0,0));

pixels.setPixelColor(8, pixels.Color(255,0,0));

}

if(SwitchPin4State == LOW){

pixels.setPixelColor(9, pixels.Color(0,255,0));

pixels.setPixelColor(10, pixels.Color(0,255,0));

pixels.setPixelColor(11, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(9, pixels.Color(255,0,0));

pixels.setPixelColor(10, pixels.Color(255,0,0));

pixels.setPixelColor(11, pixels.Color(255,0,0));

}

if(SwitchPin5State == LOW){

pixels.setPixelColor(12, pixels.Color(0,255,0));

pixels.setPixelColor(13, pixels.Color(0,255,0));

pixels.setPixelColor(14, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(12, pixels.Color(255,0,0));

pixels.setPixelColor(13, pixels.Color(255,0,0));

pixels.setPixelColor(14, pixels.Color(255,0,0));

}

if(SwitchPin6State == LOW){

pixels.setPixelColor(15, pixels.Color(0,255,0));

pixels.setPixelColor(16, pixels.Color(0,255,0));

pixels.setPixelColor(17, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(15, pixels.Color(255,0,0));

pixels.setPixelColor(16, pixels.Color(255,0,0));

pixels.setPixelColor(17, pixels.Color(255,0,0));

}

if(SwitchPin7State == LOW){

pixels.setPixelColor(18, pixels.Color(0,255,0));

pixels.setPixelColor(19, pixels.Color(0,255,0));

pixels.setPixelColor(20, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(18, pixels.Color(255,0,0));

pixels.setPixelColor(19, pixels.Color(255,0,0));

pixels.setPixelColor(20, pixels.Color(255,0,0));

}

if(SwitchPin8State == LOW){

pixels.setPixelColor(21, pixels.Color(0,255,0));

pixels.setPixelColor(22, pixels.Color(0,255,0));

pixels.setPixelColor(23, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(21, pixels.Color(255,0,0));

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

if(SwitchPin9State == LOW){

pixels.setPixelColor(24, pixels.Color(0,255,0));

pixels.setPixelColor(25, pixels.Color(0,255,0));

pixels.setPixelColor(26, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(24, pixels.Color(255,0,0));

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

if(SwitchPin10State == LOW){

pixels.setPixelColor(27, pixels.Color(0,255,0));

pixels.setPixelColor(28, pixels.Color(0,255,0));

pixels.setPixelColor(29, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

pixels.show();

SwitchPinCount = SwitchPin1State + SwitchPin2State +

SwitchPin3State + SwitchPin4State +

SwitchPin5State + SwitchPin6State +

SwitchPin7State + SwitchPin8State +

SwitchPin9State + SwitchPin10State;

Serial.println(" in State 2");

if (SwitchPinCount == 4 &&

buttonState == HIGH && buttonStatePrev == LOW){

Serial.println(" button press detected ");

state = 3;

break;

}

if (SwitchPinCount == 6 &&

buttonState == HIGH && buttonStatePrev == LOW){

Serial.println(" button press detected ");

state = 4;

break;

}

if (SwitchPinCount == 7 &&

buttonState == HIGH && buttonStatePrev == LOW){

Serial.println(" button press detected ");

state = 5;

break;

}

if (SwitchPinCount == 8 &&

buttonState == HIGH && buttonStatePrev == LOW){

Serial.println(" button press detected ");

state = 6;

break;

}

if (SwitchPinCount == 10){

Serial.println(" !WINNER! ");

state = 7;

break; //this will leave the switch case

}

break;

case 3: // 6 cup re-rack

if(SwitchPin1State == LOW){

pixels.setPixelColor(0, pixels.Color(255,0,0)); //red

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(0, pixels.Color(255,0,0));

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

if(SwitchPin2State == LOW){

pixels.setPixelColor(3, pixels.Color(0,255,0));

pixels.setPixelColor(4, pixels.Color(0,255,0));

pixels.setPixelColor(5, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(3, pixels.Color(255,255,0));

pixels.setPixelColor(4, pixels.Color(255,255,0));

pixels.setPixelColor(5, pixels.Color(255,255,0));

}

if(SwitchPin3State == LOW){

pixels.setPixelColor(6, pixels.Color(0,255,0));

pixels.setPixelColor(7, pixels.Color(0,255,0));

pixels.setPixelColor(8, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(6, pixels.Color(255,255,0));

pixels.setPixelColor(7, pixels.Color(255,255,0));

pixels.setPixelColor(8, pixels.Color(255,255,0));

}

if(SwitchPin4State == LOW){

pixels.setPixelColor(9, pixels.Color(0,255,0));

pixels.setPixelColor(10, pixels.Color(0,255,0));

pixels.setPixelColor(11, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(9, pixels.Color(255,255,0));

pixels.setPixelColor(10, pixels.Color(255,255,0));

pixels.setPixelColor(11, pixels.Color(255,255,0));

}

if(SwitchPin5State == LOW){

pixels.setPixelColor(12, pixels.Color(0,255,0));

pixels.setPixelColor(13, pixels.Color(0,255,0));

pixels.setPixelColor(14, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(12, pixels.Color(255,255,0));

pixels.setPixelColor(13, pixels.Color(255,255,0));

pixels.setPixelColor(14, pixels.Color(255,255,0));

}

if(SwitchPin6State == LOW){

pixels.setPixelColor(15, pixels.Color(0,255,0));

pixels.setPixelColor(16, pixels.Color(0,255,0));

pixels.setPixelColor(17, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(15, pixels.Color(255,255,0));

pixels.setPixelColor(16, pixels.Color(255,255,0));

pixels.setPixelColor(17, pixels.Color(255,255,0));

}

if(SwitchPin7State == LOW){

pixels.setPixelColor(18, pixels.Color(0,255,0));

pixels.setPixelColor(19, pixels.Color(0,255,0));

pixels.setPixelColor(20, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(18, pixels.Color(255,255,0));

pixels.setPixelColor(19, pixels.Color(255,255,0));

pixels.setPixelColor(20, pixels.Color(255,255,0));

}

if(SwitchPin8State == LOW){

pixels.setPixelColor(21, pixels.Color(255,0,0));

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(21, pixels.Color(255,0,0));

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(24, pixels.Color(255,0,0));

}

if(SwitchPin9State == LOW){

pixels.setPixelColor(24, pixels.Color(255,0,0)); //red

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(24, pixels.Color(255,0,0));

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

if(SwitchPin10State == LOW){

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

pixels.show();

SwitchPinCount = SwitchPin1State + SwitchPin10State +

SwitchPin9State + SwitchPin8State;

Serial.println(" in State 3 ");

if (SwitchPinCount == 4){

Serial.println(" 6 cups detected ");

state = 2;

break; //this will leave the switch case

}

break;

case 4: // 4 cup re-rack

if(SwitchPin1State == LOW){

pixels.setPixelColor(0, pixels.Color(255,0,0)); //red

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(0, pixels.Color(255,0,0));

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

if(SwitchPin2State == LOW){

pixels.setPixelColor(3, pixels.Color(255,0,0));

pixels.setPixelColor(4, pixels.Color(255,0,0));

pixels.setPixelColor(5, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(3, pixels.Color(255,0,0));

pixels.setPixelColor(4, pixels.Color(255,0,0));

pixels.setPixelColor(5, pixels.Color(255,0,0));

}

if(SwitchPin3State == LOW){

pixels.setPixelColor(6, pixels.Color(0,255,0));

pixels.setPixelColor(7, pixels.Color(0,255,0));

pixels.setPixelColor(8, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(6, pixels.Color(255,255,0));//yellow

pixels.setPixelColor(7, pixels.Color(255,255,0));

pixels.setPixelColor(8, pixels.Color(255,255,0));

}

if(SwitchPin4State == LOW){

pixels.setPixelColor(9, pixels.Color(0,255,0));

pixels.setPixelColor(10, pixels.Color(0,255,0));

pixels.setPixelColor(11, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(9, pixels.Color(255,255,0));

pixels.setPixelColor(10, pixels.Color(255,255,0));

pixels.setPixelColor(11, pixels.Color(255,255,0));

}

if(SwitchPin5State == LOW){

pixels.setPixelColor(12, pixels.Color(0,255,0));

pixels.setPixelColor(13, pixels.Color(0,255,0));

pixels.setPixelColor(14, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(12, pixels.Color(255,255,0));

pixels.setPixelColor(13, pixels.Color(255,255,0));

pixels.setPixelColor(14, pixels.Color(255,255,0));

}

if(SwitchPin6State == LOW){

pixels.setPixelColor(15, pixels.Color(255,0,0));

pixels.setPixelColor(16, pixels.Color(255,0,0));

pixels.setPixelColor(17, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(15, pixels.Color(255,0,0));//red

pixels.setPixelColor(16, pixels.Color(255,0,0));

pixels.setPixelColor(17, pixels.Color(255,0,0));

}

if(SwitchPin7State == LOW){

pixels.setPixelColor(18, pixels.Color(0,255,0));

pixels.setPixelColor(19, pixels.Color(0,255,0));

pixels.setPixelColor(20, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(18, pixels.Color(255,255,0));//yellow

pixels.setPixelColor(19, pixels.Color(255,255,0));

pixels.setPixelColor(20, pixels.Color(255,255,0));

}

if(SwitchPin8State == LOW){

pixels.setPixelColor(21, pixels.Color(255,0,0));

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(21, pixels.Color(255,0,0));//red

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

if(SwitchPin9State == LOW){

pixels.setPixelColor(24, pixels.Color(255,0,0));

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(24, pixels.Color(255,0,0));//red

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

if(SwitchPin10State == LOW){

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

pixels.show();

SwitchPinCount = SwitchPin1State + SwitchPin2State +

SwitchPin6State + SwitchPin8State +

SwitchPin9State + SwitchPin10State;

Serial.println(" in State 4 ");

if (SwitchPinCount == 6){

Serial.println(" 4 cups detected ");

state = 2;

break; //this will leave the switch case

}

break;

case 5: // 3 cup re-rack

if(SwitchPin1State == LOW){

pixels.setPixelColor(0, pixels.Color(255,0,0));

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(0, pixels.Color(255,0,0));//red, no cup detected

pixels.setPixelColor(2, pixels.Color(255,0,0));

pixels.setPixelColor(3, pixels.Color(255,0,0));

}

if(SwitchPin2State == LOW){

pixels.setPixelColor(3, pixels.Color(0,255,0));

pixels.setPixelColor(4, pixels.Color(0,255,0));

pixels.setPixelColor(5, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(3, pixels.Color(255,0,0));

pixels.setPixelColor(4, pixels.Color(255,0,0));

pixels.setPixelColor(5, pixels.Color(255,0,0));

}

if(SwitchPin3State == LOW){

pixels.setPixelColor(6, pixels.Color(0,255,0));

pixels.setPixelColor(7, pixels.Color(0,255,0));

pixels.setPixelColor(8, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(6, pixels.Color(255,255,0));//yellow

pixels.setPixelColor(7, pixels.Color(255,255,0));

pixels.setPixelColor(8, pixels.Color(255,255,0));

}

if(SwitchPin4State == LOW){

pixels.setPixelColor(9, pixels.Color(0,255,0));

pixels.setPixelColor(10, pixels.Color(0,255,0));

pixels.setPixelColor(11, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(9, pixels.Color(255,255,0));

pixels.setPixelColor(10, pixels.Color(255,255,0));

pixels.setPixelColor(11, pixels.Color(255,255,0));

}

if(SwitchPin5State == LOW){

pixels.setPixelColor(12, pixels.Color(0,255,0));

pixels.setPixelColor(13, pixels.Color(0,255,0));

pixels.setPixelColor(14, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(12, pixels.Color(255,255,0));

pixels.setPixelColor(13, pixels.Color(255,255,0));

pixels.setPixelColor(14, pixels.Color(255,255,0));

}

if(SwitchPin6State == LOW){

pixels.setPixelColor(15, pixels.Color(255,0,0));

pixels.setPixelColor(16, pixels.Color(255,0,0));

pixels.setPixelColor(17, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(15, pixels.Color(255,0,0));//red

pixels.setPixelColor(16, pixels.Color(255,0,0));

pixels.setPixelColor(17, pixels.Color(255,0,0));

}

if(SwitchPin7State == LOW){

pixels.setPixelColor(18, pixels.Color(255,0,0));

pixels.setPixelColor(19, pixels.Color(255,0,0));

pixels.setPixelColor(20, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(18, pixels.Color(255,0,0));//red

pixels.setPixelColor(19, pixels.Color(255,0,0));

pixels.setPixelColor(20, pixels.Color(255,0,0));

}

if(SwitchPin8State == LOW){

pixels.setPixelColor(21, pixels.Color(255,0,0));

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(21, pixels.Color(255,0,0));//red

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

if(SwitchPin9State == LOW){

pixels.setPixelColor(24, pixels.Color(255,0,0));

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(24, pixels.Color(255,0,0));//red

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

if(SwitchPin10State == LOW){

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

pixels.show();

SwitchPinCount = SwitchPin1State + SwitchPin2State +

SwitchPin6State + SwitchPin7State +

SwitchPin8State + SwitchPin9State +

SwitchPin10State;

Serial.println(" in State 5 ");

if (SwitchPinCount == 7){

Serial.println(" 3 cups detected ");

state = 2;

break; //this will leave the switch case

}

break;

case 6: // 2 cup re-rack

if(SwitchPin1State == LOW){

pixels.setPixelColor(0, pixels.Color(255,0,0));

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(0, pixels.Color(255,0,0));//red, no cup detected

pixels.setPixelColor(1, pixels.Color(255,0,0));

pixels.setPixelColor(2, pixels.Color(255,0,0));

}

if(SwitchPin2State == LOW){

pixels.setPixelColor(3, pixels.Color(255,0,0));

pixels.setPixelColor(4, pixels.Color(255,0,0));

pixels.setPixelColor(5, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(3, pixels.Color(255,0,0));

pixels.setPixelColor(4, pixels.Color(255,0,0));

pixels.setPixelColor(5, pixels.Color(255,0,0));

}

if(SwitchPin3State == LOW){

pixels.setPixelColor(6, pixels.Color(255,0,0));

pixels.setPixelColor(7, pixels.Color(255,0,0));

pixels.setPixelColor(8, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(6, pixels.Color(255,0,0));

pixels.setPixelColor(7, pixels.Color(255,0,0));

pixels.setPixelColor(8, pixels.Color(255,0,0));

}

if(SwitchPin4State == LOW){

pixels.setPixelColor(9, pixels.Color(0,255,0));

pixels.setPixelColor(10, pixels.Color(0,255,0));

pixels.setPixelColor(11, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(9, pixels.Color(255,255,0));//yellow

pixels.setPixelColor(10, pixels.Color(255,255,0));

pixels.setPixelColor(11, pixels.Color(255,255,0));

}

if(SwitchPin5State == LOW){

pixels.setPixelColor(12, pixels.Color(255,0,0));

pixels.setPixelColor(13, pixels.Color(255,0,0));

pixels.setPixelColor(14, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(12, pixels.Color(255,0,0));

pixels.setPixelColor(13, pixels.Color(255,0,0));

pixels.setPixelColor(14, pixels.Color(255,0,0));

}

if(SwitchPin6State == LOW){

pixels.setPixelColor(15, pixels.Color(255,0,0));

pixels.setPixelColor(16, pixels.Color(255,0,0));

pixels.setPixelColor(17, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(15, pixels.Color(255,0,0));//red

pixels.setPixelColor(16, pixels.Color(255,0,0));

pixels.setPixelColor(17, pixels.Color(255,0,0));

}

if(SwitchPin7State == LOW){

pixels.setPixelColor(18, pixels.Color(0,255,0));

pixels.setPixelColor(19, pixels.Color(0,255,0));

pixels.setPixelColor(20, pixels.Color(0,255,0));

}

else {

pixels.setPixelColor(18, pixels.Color(255,255,0));//yellow

pixels.setPixelColor(19, pixels.Color(255,255,0));

pixels.setPixelColor(20, pixels.Color(255,255,0));

}

if(SwitchPin8State == LOW){

pixels.setPixelColor(21, pixels.Color(255,0,0));

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(21, pixels.Color(255,0,0));//red

pixels.setPixelColor(22, pixels.Color(255,0,0));

pixels.setPixelColor(23, pixels.Color(255,0,0));

}

if(SwitchPin9State == LOW){

pixels.setPixelColor(24, pixels.Color(255,0,0));

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(24, pixels.Color(255,0,0));//red

pixels.setPixelColor(25, pixels.Color(255,0,0));

pixels.setPixelColor(26, pixels.Color(255,0,0));

}

if(SwitchPin10State == LOW){

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

else {

pixels.setPixelColor(27, pixels.Color(255,0,0));

pixels.setPixelColor(28, pixels.Color(255,0,0));

pixels.setPixelColor(29, pixels.Color(255,0,0));

}

pixels.show();

SwitchPinCount = SwitchPin1State + SwitchPin2State +

SwitchPin3State + SwitchPin5State +

SwitchPin6State + SwitchPin8State +

SwitchPin9State + SwitchPin10State;

Serial.println(" in State 6 ");

if (SwitchPinCount == 8){

Serial.println(" 2 cups detected ");

state = 2;

changeTime = millis();

break; //this will leave the switch case

}

break;

case 7: // !WINNER! light show

for(pixelIndex=0; pixelIndex<NUMPIXELS; pixelIndex++) {//for each pixel...

colorIndex = pixelIndex + factor;

colorIndex = colorIndex%7; //modulud comman to rollover colorINdex

pixels.setPixelColor(pixelIndex,

RbArrayR[colorIndex],

RbArrayG[colorIndex],

RbArrayB[colorIndex]);

}

factor++;

if(factor>= 10)

{

factor = 0;

}

pixels.show(); //send the updated pixel colors to the hardware.

delay(DELAYVAL); //pause before next pass through loop

Serial.println(" in State 7 ");

if (millis() - changeTime > 20000)

{

digitalWrite(12, HIGH);

break;

}

if (buttonState == HIGH && buttonStatePrev == LOW)

{

Serial.println(" game reset ");

state = 1;// this will re-start the game

break; //this will leave the switch case

}

break;

}

}