Lights and Analog Dial

Code Walkthrough

Code Color Guide

Keyword Function Built-In Function Comments

# Define Pins

Start by defining your pins so that they will not get mixed up. This also allows for us to use this definition in the remainder of the code.

#define blue 10 // blue LED is pin 10

#define yellow 12 // yellow LED is pin 12

#define green 11 // green LED is pin 11

#define red 13 // red LED is pin 13

# Set Integers

*The keyword int is a declaration of a variable whose datatype is an integer.*

int dial = 0; // initial dial speed is 0

int speed = 50; // sets the delay time

const int buttonPin = 9; // dial will always be pin 9

# Define Bool

*Bool is a declaration of true or false.*

bool buttonState = 0; // sets the buttonState to false

bool lastState = 0; // sets the lastState to false

# Void Setup

The void setup runs one time as soon as the program starts

void setup() {

Serial.begin(9600); // port for the serial monitor

pinMode(blue, OUTPUT); // sets LED to an output

pinMode(red, OUTPUT);

pinMode(green, OUTPUT);

pinMode(yellow, OUTPUT);

pinMode(buttonPin, INPUT); // sets dial to an input

}

# Void Loop

Runs in a constant loop

void loop() {

// buttonState reads to see if button pin is set to true or false

buttonState = digitalRead(buttonPin);

dial = analogRead(0); // reads the value of the dial on analog pin 0

Serial.println(dial); // prints the analogRead value to serial monitor

if (dial < 256) {// when the dial value is less the 256 it runLights case 0

runLights(0);

} else if (dial < 512) {// when the dial value is less the 512 it runLights case 1

runLights(1);

}

else if (dial < 768) {// when the dial value is less the 768 it runLights case 2

runLights(2);

}

else {// when the dial value is anything else it runLights case 3

runLights(3);

}

}

# Void runLights

A function that holds the cases

void runLights(int c) {

switch (c) {

case 0: // one pattern of blinking lights

digitalWrite(blue, HIGH); // turns light off

delay(speed); // reads the int speed to change the delay

digitalWrite(blue, LOW); // turns light off

Serial.println("blue"); // prints blue to the serial monitor when done

digitalWrite(yellow, HIGH);

delay(speed);

digitalWrite(yellow, LOW);

Serial.println("yellow");

digitalWrite(green, HIGH);

delay(speed);

digitalWrite(green, LOW);

Serial.println("green");

digitalWrite(red, HIGH);

delay(speed);

digitalWrite(red, LOW);

Serial.println("red");

break; // changes to next pattern

case 1: // second pattern of blinking lights

digitalWrite(green, HIGH);

delay(speed);

digitalWrite(green, LOW);

digitalWrite(yellow, HIGH);

delay(speed);

digitalWrite(yellow, LOW);

digitalWrite(blue, HIGH);

delay(speed);

digitalWrite(blue, LOW);

digitalWrite(red, HIGH);

delay(speed);

digitalWrite(red, LOW);

break; // changes to next pattern

case 2: // third pattern of blinking lights

digitalWrite(yellow, HIGH);

delay(speed);

digitalWrite(yellow, LOW);

digitalWrite(red, HIGH);

delay(speed);

digitalWrite(red, LOW);

digitalWrite(green, HIGH);

delay(speed);

digitalWrite(green, LOW);

digitalWrite(blue, HIGH);

delay(speed);

digitalWrite(blue, LOW);

break; // changes to next pattern

case 3:// turns off

break;

}

}