// C++ code

//

/\*

=======================

Changes Made:

=======================

Swap Yellow and Green LED pins to be uniform with a normal traffic light

Configured Button to use internal pullup resistor to remove 10k resistor (in Setup)

Added a global int variable "buttonState" for digitalRead for the switch, to be uniform.

Button will remain HIGH (5V) when not pressed, and change to LOW (0V) when pressed.

Added serial functions, to see buttonState (sw input).

=======================

Current Issues:

=======================

Button is not as responsive, possibly due to using delay instead of timer.

Does not Stop rotation after 10 minutes no traffic.

\*/

//const variables will not change.

const int buzzer =  9; //Set buzzer to pin 9

const int led\_r =  7; //Set red LED to pin 7

const int led\_y =  6; //Set yellow LED to pin 6

const int led\_g =  5; //Set green LED to pin 5

const int sw =  2; //Set switch to pin 2

//variables will change.

int buttonState; //Will be used to check if button is pushed.

void setup() {

  //Output setup for pins.

  pinMode(led\_r, OUTPUT);

  pinMode(led\_y, OUTPUT);

  pinMode(led\_g, OUTPUT);

  pinMode(buzzer, OUTPUT);

  pinMode(sw, INPUT\_PULLUP); //Using internal pullup resistor to replace 10k resistor.

  Serial.begin(9600); //For debugging, serial monitor 9600 baud rate

}

void loop() {

  buttonState = digitalRead(sw); //detects input of the switch.

  //Debugging Output

  Serial.print(buttonState); //Output buttonState in serial monitor.

  Serial.println(); //Print carriage (new line).

  //If the button is pressed, it will be LOW (0V).

  //Else if the button is not pressed, it will remain HIGH (5V).

  if(buttonState == LOW) {

    while(1) {

      digitalWrite(led\_r, HIGH);

      delay(17000);

      digitalWrite(buzzer, HIGH);

      delay(3000);

      digitalWrite(buzzer, LOW);

      digitalWrite(led\_r, LOW);

      digitalWrite(led\_g, HIGH);

      delay(17000);

      digitalWrite(buzzer, HIGH);

      delay(3000);

      digitalWrite(buzzer, LOW);

      digitalWrite(led\_g, LOW);

      delay(1000);

      digitalWrite(led\_y, HIGH);

      digitalWrite(buzzer, HIGH);

      delay(3000);

      digitalWrite(led\_y, LOW);

      digitalWrite(buzzer, LOW);

    }

  }

 else {

    buttonState = digitalRead(sw);

    digitalWrite(led\_r, HIGH);

    delay(1000);

    buttonState = digitalRead(sw);

    digitalWrite(led\_r, LOW);

    delay(1000); // Wait for 1000 millisecond(s)

 }

}