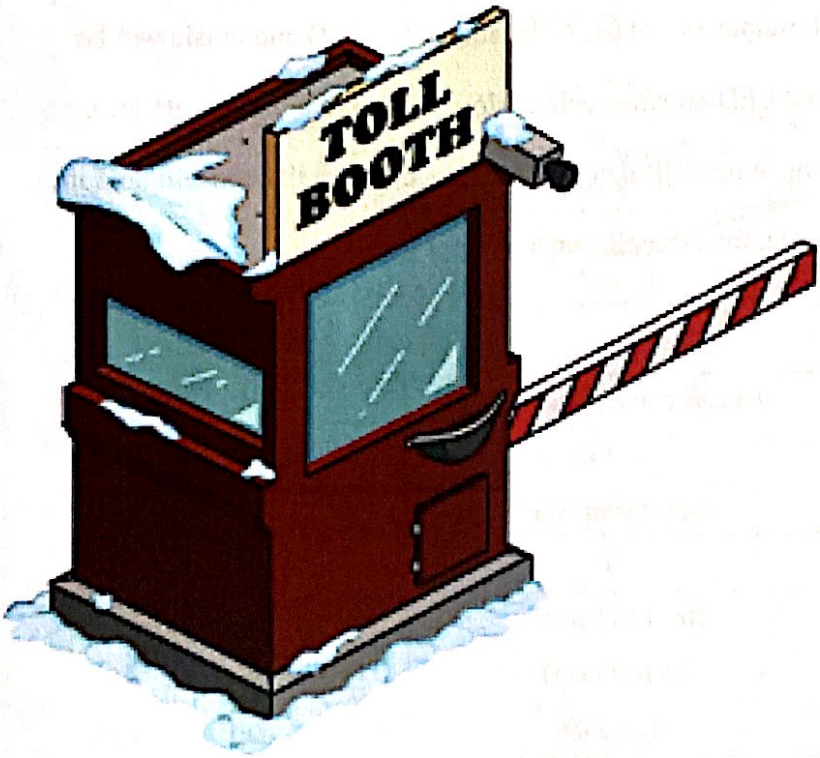
TOLL BOOTH REVISITED PROJECT



Ian Wilkinson

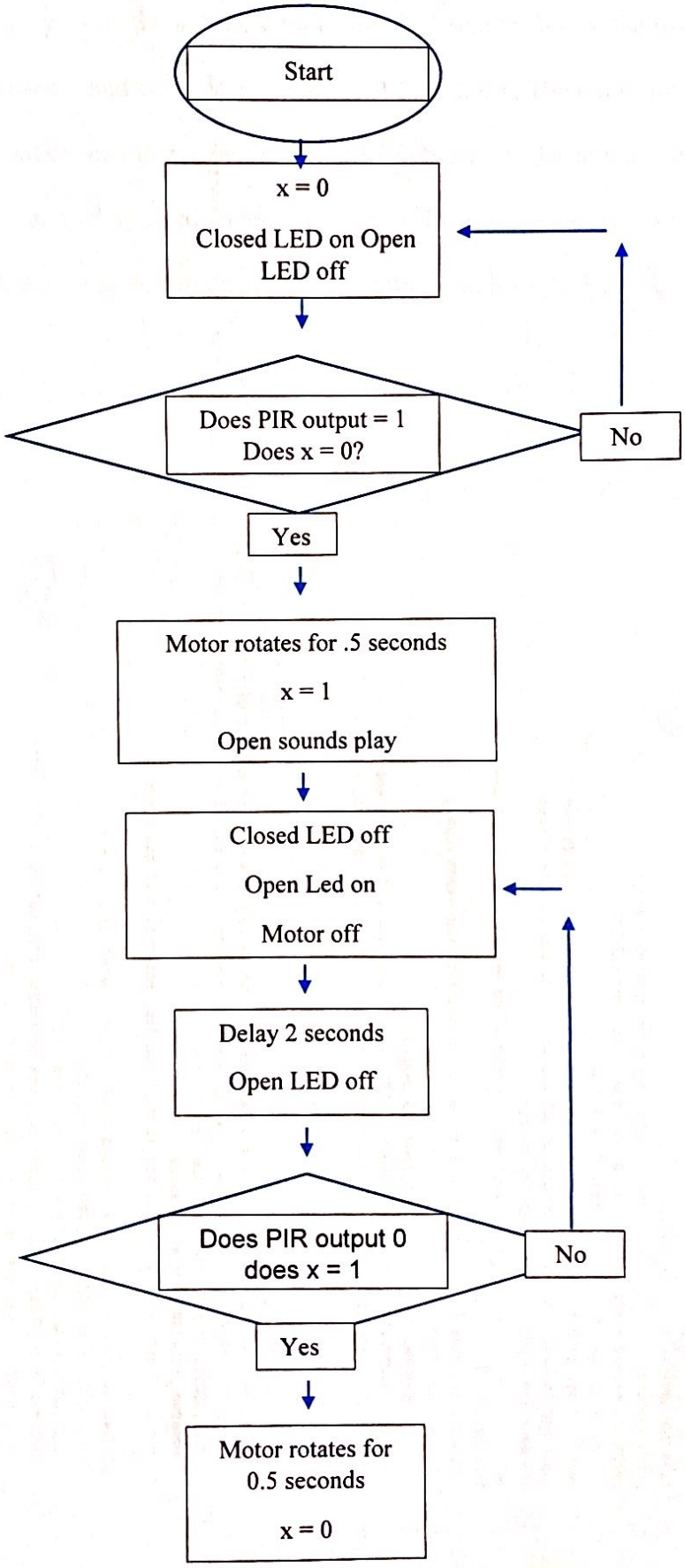
Kholilur Rahman

Keyur Rana

Design Proposal

The previous design of Tollbooth included two limit switches and two user inputs. To innovate the Tollbooth, students will use Arduino instead of state machine with CMOD S6 to make the Tollbooth function without user input. Our plan is to use the PIR sensor to detect disturbance such as a car and will output 1s and 0s. A flashing green LED and music will be played when gate is open and a red LED on when gate is closed. The motor will rotate for 0.5 seconds when opening and closing. When PIR detects disturbance, gate will open and hold its position until there is no disturbance, for example, when there is no car.

Flow chart


Start
x = 0
Closed LED on Open LED off
Does PIR output = 1
Does x = 0?
No
Yes
Motor rotates for .5 seconds
x = 1
Open sounds play
Closed LED off
Open Led on
Motor off
Delay 2 seconds
Open LED off
Does PIR output 0
does x = 1
No
Yes
Motor rotates for 0.5 seconds
x = 0

```
/*Ian Wilkison
 Toll Booth Revisted Circuit
 Students revisted their Toll Booth project and take off the switches and use Arduino to simplify the creation.
 A PIR sensor was used to detect a car moving toward the tool Booth Using the output from the PIR sensor the code will read this INPUT
 and if the PIR sensor senses motion it will output HIGH the TOLL BOOTH gate would open, a green LED would blink  and sounds would play
 If the Motion does not sense anything the gate will close or stay closed and an Red led will turn on
*/
#include <Servo.h>
Servo myservo;              // create servo object to control a servo
const int pirpin = A0;      // analog pin used to connect the PIR
int val = digitalRead(A0); // input from PIR sensor
const int ledOpenPin = 12; // output to LED on when open and off when closed
const int ledClosePin = 8; // output to LED off when open and on when closed
int peizoPin = 10;         // pin for buzzer
int x=0;                  // variable used to make loops happen once and not repeat constant
void setup() {
  myservo.attach(6);       // attaches the servo on pin 6 to the servo object
  Serial.begin(9600);
  pinMode(pirpin, INPUT); // input from PIR sensor
  pinMode(ledOpenPin, OUTPUT);// output to LED on when open and off when closed
  pinMode(ledClosePin,OUTPUT); // output to LED off when open and on when closed
}
void loop()
{
  val = digitalRead(A0);
  if (val == HIGH and x==0){ // if the value from the PIR is high x = 0. X is used so that the code is not constantly rising
    myservo.write(60); // myservo.write(x) if x = 90 the motor is neutral if x<90 motor turns a direction if x>90 the motor turns in the opposite direction
    delay(500); // delay to give time for the motor to turn
    myservo.write(90); // stops motor
    x = 1; // sets x to 1 so that the if will not loop and second if can start
    }
  if (val == LOW and x == 1){ // if PIR does not sense anything and x = 1
    digitalWrite(8, HIGH); // RED light turns on
    digitalWrite(12, LOW); // green light turns off
    myservo.write(120); // motor turns opposite direction from before to close gate
    delay(500); // delay to give time for the motor to move
    myservo.write(90);// stops the motor
    x = 0; // sets x to 0 to stop the code from looping this if statement and allows it to go through the first if
    }
     if (val == HIGH){ //if the pir senses something
    digitalWrite(12, HIGH); //Green LED light turns on
    digitalWrite(8, LOW); // Red LED turns off
    tone (peizoPin, 5000, 50); // tone(pim, frequency, duration) the tone function will make the buzzer buzz at a certain frequency for a certain amount of time
    delay(2000); // Delay to make the Green LED blink
    digitalWrite(12, LOW); // turns Greeed LED off
    }
 delay(1000);}
```

Code

Code

The code works by using the input output from a PIR sensor. When disturbance is detected then PIR outputs a 1 and variable x = 0 so the If loop starts. The motor turns in one direction for a set amount of time then it stops. Green LED flashes and a sound is played red LED is off. X is set to 1 and when disturbance is no detected the second if loop begins the motor turns in the opposite direction as before for the same se time then it stops. Red LED turns on and green LED if off