BTE3262

ELECTRICAL AUTOMATION

Project: Automatic Gate Opener

Prepared by:

No	Name	Matrix No	Section
1	Jehovah Yii Zui Hon	TB20091	01
2	Wan Nor Samsiah Binti Wan Osman	TB19022	01
3	Affrina Sahira binti Azham	TB19044	01

Lecturer:

Ts. Hj. SYAHRULNAIM BIN MOHAMAD NAWI

Table of Contents

Introduction

Automatic gates are extremely useful in businesses, industries, colleges, and schools. Automatic gates are important and very useful because they save time, reduce the amount of human force required to open and close the manual gate, which saves energy, and protect gate operators from health hazards caused by being exposed to harsh weather conditions while operating the manual gate. The gate was designed, built, and installed, and the control and power supply units were also designed and implemented. These various units were incorporated and put into action. The automatic gate was tested for speed and efficiency, and the results show that the gate is 90% efficient.

This project demonstrates how to control an automatic gate using a switch and a relay. The goal is to provide a secure and convenient way to access a property by using a simple switch that triggers the gate to open and close. The distance from the gate to an object is measured by an ultrasonic sensor and used to determine the state of the gate. When the distance is less than 5 cm, the gate is opened, and a green LED is turned on. When the switch is pressed, the relay is activated and the gate is opened, and a blue LED is turned on. The servo motor is used to provide visual feedback to the user, by swinging from 0 to 90 degrees when the switch is pressed and back to 0 degrees when the gate is closed.

Task Distribution

Name	Contribution
Affrina Sahira bt Azham	Introduction, Scope, Objective, Gantt chart, Recommendation for improvement, Task Distribution
Jehovah Yii Zui Hon	Introduction, Methodology, Project Implementation, List of components, Circuit Design and Results
Wan Nor Samsiah	Flowchart, Discussion, Conclusion

Scope

The scope for this project is the introduction with the growing demand for automation in the industrial and residential sectors, one of the primary concerns for end-users is gate automation. Increasing urbanization and infrastructure development, such as shopping malls, hypermarkets, and supermarkets, has also increased demand for automated equipment to reduce overall labor costs. Automatic gate opening systems are gates that have been integrated with electronic components such as ultrasonic sensors, Arduino Uno, push buttons, 5V relays, and servo motor control to reduce the amount of effort required to open and close gates. These systems are primarily used in high-traffic areas with frequent gate opening and closing, such as shopping malls, theatres, industrial warehouses, and electronic toll collection systems.

Objectives

The objective in this project is:

- To design and develop automatic gate system
- To provide a secure and convenient way to access a property by using a simple switch that triggers the gate to open and close.
- To do hardware interfacing

Methodology

In this automatic system, the methodology to design the system requires the development of the circuit using components such as Arduino, electrical and mechanical components. The electrical parts used in this system is resistors, LED, relays, sensor, wires and motor. When the electrical circuit was built according to the circuit diagram designed in proteus software and simulation were run to make sure it is able to operate theoretically. After verification of simulation can be run, the circuit is built on breadboard by connecting the components to the Arduino to control the circuit outcome. Power supply used for the automatic gate is 240V AC and regulated to 24V DC using a power supply regulator.

Gantt Chart

Task	W11	W12	W13	W14	W15
Lecturer give the project task					
Searching for Project automation					
Implementation of hardware					
Hardware testing					
Create report and slide					
Presentation					

Project Implementation

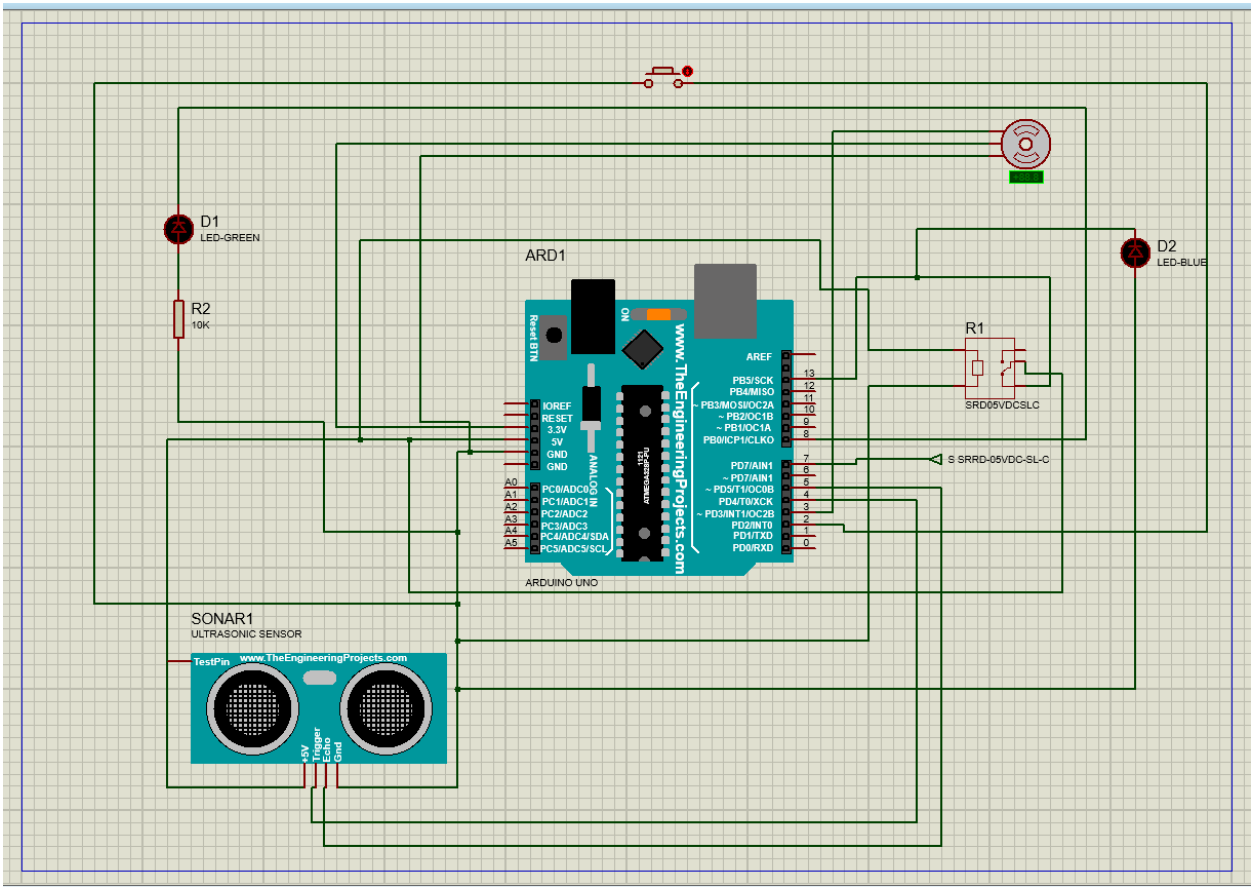
List of Components

- Arduino Uno
- Ultrasonic sensor
- Breadboard
- Jumper wire
- Servo motor
- 5V relay switch
- Push button switch
- Green LED
- Blue LED

Bill of Materials

No	Item	Quantity (Unit)	Price (RM)
1	SG90 Micro Servo	1	6.50
2	LED Super Bright 5mm Blue	1	0.20
3	LED 5mm Green	1	0.10
4	6x6x1 Push Button 2 Pins	1	0.40
5	Breadboard 16.5x5.5cm (830 Holes)	1	3.90
6	40-Way 20cm Dupont Jumper Wire	1	2.50
7	Maker UNO X	1	48.00
8	Single Channel 5V Relay Breakout Board	1	5.00
9	Resistor 0.25W 5% 1K5 - 15K	1	0.05
		Total	66.65

Circuit Design



Ultrasonic Sensor

Vcc => 5V

Trig => D4

Echo => D5

Gnd => Gnd

Servo Motor

Yellow Wire => D3

Red Wire => 5V

Chocolate Wire => GND

Relay Switch

S => D7

+ => 5V

– => GND

C => 3.3V

NO => D13

Green LED

+ => D8

– => 10kΩ => GND

Blue LED

+ => D13

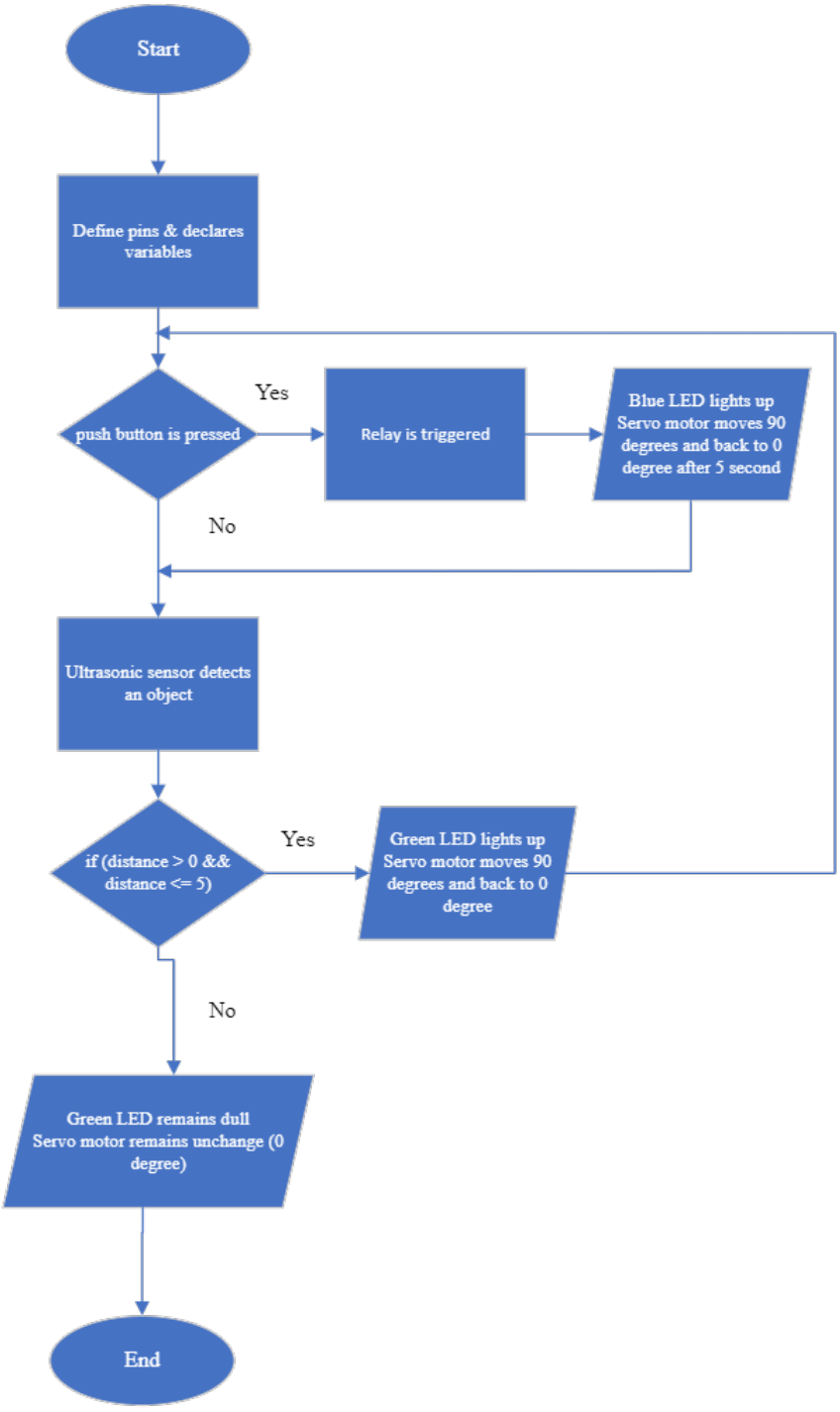
– => GND

Push Button

Pin 1 => D2

Pin 2 => GND

Flowchart


Start
Define pins & declares variables
push button is pressed
Yes
Relay is triggered
Blue LED lights up
Servo motor moves 90 degrees and back to 0 degree after 5 second
No
Ultrasonic sensor detects an object
if (distance > 0 && distance <= 5)
Yes
Green LED lights up
Servo motor moves 90 degrees and back to 0 degree
No
Green LED remains dull
Servo motor remains unchange (0 degree)
End

Code

```
#include <Servo.h>

#define trigPin 4
#define echoPin 5
#define buttonPin 2
#define relayPin 7
#define blueLedPin 13
#define greenLedPin 8
#define servoPin 3

Servo myServo;
bool buttonState = 0;

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(buttonPin, INPUT_PULLUP);
  pinMode(relayPin, OUTPUT);
  pinMode(blueLedPin, OUTPUT);
  pinMode(greenLedPin, OUTPUT);
  myServo.attach(servoPin);
}

void loop() {
  long duration, distance=0;
  buttonState = digitalRead(buttonPin);

  digitalWrite(trigPin, LOW);
  delayMicroseconds(10);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1; // Divide by 29.1 to convert the distance to
centimeters

  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  if (!buttonState) {
```

```
    digitalWrite(relayPin, HIGH);
    digitalWrite(blueLedPin, HIGH);
    digitalWrite(greenLedPin, LOW);
    myServo.write(90);
     delay(5000);
    myServo.write(0);
    Serial.print("Button Press: 1\n");

  } else {
    digitalWrite(relayPin, LOW);
    digitalWrite(blueLedPin, LOW);
    Serial.print("Button Press: 0\n");

      if (distance > 0 && distance <= 10) {
        digitalWrite(greenLedPin, HIGH);
        myServo.write(90);
      } else  {
        digitalWrite(greenLedPin, LOW);
        myServo.write(0);
      }


  }



}
```

Result

Function of the project

The purpose of the project is to measure the distance using an ultrasonic sensor, and based on the distance, the system will perform different actions. When the button is not pressed, the ultrasonic sensor continuously measures the distance, and if the distance is less than 5 cm, the green LED lights up. When the button is pressed, the blue LED lights up, the relay is triggered, and the servo moves 180 degrees.

Benefits/Advantages to community/human/environment development

The benefits of this project include:

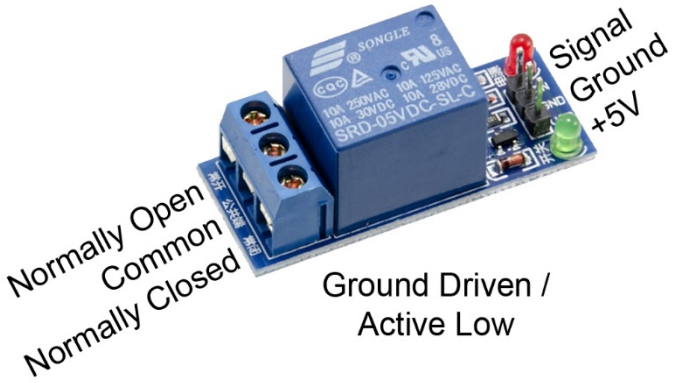
1. Convenience: Automating the gate makes it easier for the user to enter and exit the property without having to get out of the car to manually open and close the gate.
2. Security: Automated gates provide added security as they can be designed to open only when an authorized person or vehicle is detected.
3. Time-saving: With an automated gate, there's no need to stop and manually open the gate, saving the user time and effort.
4. Increased property value: Automated gates can add value to a property as they are seen as a luxury feature.
5. Energy efficiency: Automated gates can reduce energy consumption as they don't require manual effort to open and close, reducing the need for electricity or other power sources.

Overall, an automated gate project can bring added convenience, security, time-saving, increased property value and energy efficiency to a community or individual.

Discussion

Connection

The relay should be connected to the arduino board as follows:



- The NO (Normally Open) pin of the relay is connected to the servo motor so that blue LED is switch off when the button is not pressed, and switch on when the button is pressed.
- The COM (Common) pin of the relay is connected to the arduino 3.3V pin.
- The NC (Normally Closed) pin of the relay can be left unconnected.
- The VCC and GND pins of the relay is connected to 5V pin and GND pin respectively.

It's also important to note that the relay must be connected to a separate power supply from the arduino board, as the servo motor requires a high voltage and current to operate.

Code

The code provided is a basic simulation of an automated gate system using an arduino board. The code utilizes an ultrasonic sensor to detect the distance of an object and a button to manually open the gate. The servo motor acts as the gate, and the relay is used to control the gate's movement. When the object is 5 cm (in real life would be 500 cm) away from the ultrasonic sensor, the gate (servo motor) will move upwards (90◦). The user can manually control the gate with the switch and relay. The distance of the object detected by the ultrasonic sensor is calculated by the formula:

$$\text{distance} = \frac{speed\ of\ sound \times time\ taken}{2}$$

It is divided by 2 because sound wave goes and hit the surface then comeback to receiver, so waves travelled two times the distance therefore we need to divide by two.

In loop function we will setup two variable in type of long which is duration & distance. Long which is 4 bytes in size and store 32 bit. It can hold very large number.

```
long duration, distance=0;
```

The trigger pin starts in a low state for 2 microseconds, then transitions to a high state for 10 microseconds, before returning to a low state for that iteration. The PulseIn function reads the pulse state (high or low) on a specified pin. It calculates the time difference between the high and low states on the echo pin, which is expressed in microseconds. To convert to milliseconds, 1000 microseconds are equal to one millisecond.

```
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
```

To convert the formula for the speed of sound, which is equal to 343.5m/s at 20 degrees Celsius, to cm/microsecond, we multiply it by 100 and divide it by 1 million. This results in a value of 0.03435 cm/microsecond. To find the duration, we take the reciprocal of this value, which is 29.1. This can be substituted into the original formula in place of time. If you prefer the distance to be in millimeters, simply replace 29.1 with 2.91 and change the "serial.println" to "mm".

```
distance = (duration/2) / 29.1;
```

Error

One potential error in the code is that it doesn't check for errors in the distance readings from the ultrasonic sensor. It is possible that the readings may be incorrect, or the sensor may not function properly, causing the green LED to light up incorrectly or not at all. To solve this, error handling and testing should be added to the code to ensure accurate readings.

Conclusion

In conclusion, this project of an automated gate system using an ultrasonic sensor, a relay, a servo motor and buttons, is a simple yet effective demonstration of how microcontrollers such as the Arduino can be used to control and automate physical systems. The code provided, which uses the Servo library and incorporates the control of a relay, blue LED and green LED, effectively implements the desired functionality of the gate opening when the car comes near it and also allowing manual opening with the button press. The code can be further optimized and improved upon with the implementation of more advanced features such as safety measures and smoother servo movement. The wiring of the relay with the Arduino must be done correctly to avoid any errors or malfunctions in the system. Overall, this project provides a foundation for future development and implementation of automated gate systems in real-world applications.

Recommendations for improvement

There are some recommendations for this project:

- We can architecture the auto gate system so that it opens automatically using a smartphone interface.
- We can attach a limit switch to both sides of the gate frame and install it. Coding can be modified so that the gate can be opened or closed by pressing the interface button once only, rather than repeatedly pressing the designed button. When the gate is opened or closed, each limit switch is activated, and the automatic movement is halted. This can further reduce the time required by the user to open and close the gate, as opposed to repeatedly pressing the button.

References

1. "(PDF) Design and Development of Low Cost Auto Gate System for House Part 5." *ResearchGate*, ResearchGate, 2016, www.researchgate.net/publication/289672125_Design_and_Development_of_Low_Cost_Auto_Gate_System_for_House_Part_5. Accessed 4 Feb. 2023.
2. Zone, Automation. "Gate Automation Trends and Scope." *Automationzone.co.in*, 2023, www.automationzone.co.in/blog_desc.php?ser=Nw==. Accessed 4 Feb. 2023.
3. Masum Vlog. "How to Make a Automatic Gate Opener Using Arduino and Ultrasonic Sensor || Arduino Projects ||." *YouTube*, 19 Feb. 2021, www.youtube.com/watch?v=fbRSgNNMADc. Accessed 4 Feb. 2023.
4. Robotica DIY. (2019, July 21). *How to measure distance using ultrasonic sensor with Arduino*. Robotica DIY. Retrieved February 9, 2023, from https://roboticadiy.com/how-to-measure-distance-using-ultrasonic-sensor-with-arduino/
5. YouTube. (2017, April 1). *Using 5V 1 channel relay module for Arduino*. YouTube. Retrieved February 5, 2023, from https://www.youtube.com/watch?v=7tUGUXyloXQ&ab_channel=Robojax

Appendixes

Github

https://github.com/hovahyii/BTE3262-Final-Project-Automated-Gate

Group Photo