BACHELOR OF TECHNOLOGY

IN

ELECTRONICS & TELE-COMMUNICATION ENGINEERING

(2019-23)

SUBMITTED BY

VINAY PATIL

19070123075

Report on

Automatic Toll Collection System Using RFID

Description

RFID based toll collection is used for fast and efficient toll collection at toll booths. Time and efficiency are a matter of priority of present day. To overcome the major issues of vehicle congestion and time consumption RFID technology is used. Handheld tag reads the tag and toll deduction takes place through a prepaid card assigned to the concerned RFID tag that belongs to the owners' account.

The Benefits are:

- Faster and more efficient
- Ability to make payments by keeping a balance on the card itself
- Fuel wastage
- Reduced emissions by reducing deceleration rate
- Waiting time of vehicles in queue.

List of Components Used

1) Arduino Board
2) RFID Card Reader
3) RFID Card
4) Liquid Crystal Display
5) Infrared Sensors (IR Sensors)
6) Servo Motor
7) Buzzer
8) Potentiometer (10k)
9) Resistors (220 ohm, 4.7k ohm, 1k ohm)
10) Breadboard
11) Jumper Wires

Arduino Sketch

```
#include <SPI.h>
#include <MFRC522.h>
#include <OnewireKeypad.h>
#include <LiquidCrystal.h>
#include <Servo.h>

LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

Servo servo;
int servoPos = 0;

#define sensorPin1 A2
#define sensorPin2 A3
#define buzzerPin A4

int senVal1 = 0;
int senVal2 = 0;

#define RST_PIN 8
#define SS_PIN 10
```

```
int card1Balance = 5000;
int card2Balance = 300;

#define num 7
char Data[num];
byte data_count = 0;

String num1, num2, card, card2;
int a, b;
char Key;

bool recharge = true;

MFRC522 mfrc522(SS_PIN, RST_PIN);

int state = 0;

char KEYS[] = {
  '1', '2', '3', 'A',
  '4', '5', '6', 'B',
```

```
  '7', '8', '9', 'C',
  '*', '0', '#', 'D'
};

OnewireKeypad <Print, 16 > KP2(Serial, KEYS, 4, 4, A0, 4700,
1000, ExtremePrec );

void setup () {
  lcd.begin(16, 2);
  Serial.begin(9600);
  servo.attach(9);
  servo.write(30);

  pinMode(sensorPin1, INPUT);
  pinMode(sensorPin2, INPUT);
  pinMode(buzzerPin, OUTPUT);

  KP2.SetKeypadVoltage(5.0);

  SPI.begin();
  mfrc522.PCD_Init();
```

```
  lcd.setCursor(0, 0);
  lcd.print(" Automatic toll");
  lcd.setCursor(0, 1);
  lcd.print("colection system");
  delay(3000);
  lcd.clear();
}
void loop()
{
  if (recharge == 0)
  {
    reCharge();
  }
  else
  {
    lcd.setCursor(0, 0);
    lcd.print("   Welcome!!!");
    sensorRead();
    rfid();
    KeyPad();
    if (senVal1 == 0)
```

```
{
  servoDown();
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Vehicle detected");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Put your card to");
  lcd.setCursor(0, 1);
  lcd.print("the reader......");
  delay(4000);
  lcd.clear();
}
else if (senVal2 == 0 && state == 1)
{
  servoUp();
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Have a safe");
  lcd.setCursor(0, 1);
```

```
      lcd.print("journey");
      delay(3000);
      lcd.clear();
      state = 0;
    }
  }
}

void servoDown()
{
  servo.attach(9);
  for (servoPos = 30; servoPos <= 120; servoPos += 1)
  {
    servo.write(servoPos);
    delay(5);
  }
}

void servoUp()
{
  servo.attach(9);
```

```
  for (servoPos = 120; servoPos >= 30; servoPos -= 1)
  {
    servo.write(servoPos);
    delay(5);
  }
}

void sensorRead()
{
  senVal1 = digitalRead(sensorPin1);
  senVal2 = digitalRead(sensorPin2);
}

void rfid()
{
  if ( ! mfrc522.PICC_IsNewCardPresent())
  {
    return;
  }
  if ( ! mfrc522.PICC_ReadCardSerial())
  {
```

```
    return;
  }

  String content = "";
  for (byte i = 0; i < mfrc522.uid.size; i++)
  {
    content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" :
" "));
    content.concat(String(mfrc522.uid.uidByte[i], HEX));
  }
  content.toUpperCase();

  if (content.substring(1) == "84 E6 E0 1F")
  {
    if (card1Balance >= 500)
    {
      lcdPrint();
      card1Balance = card1Balance - 500;
      lcd.setCursor(9, 1);
      lcd.print(card1Balance);
      delay(2000);
      lcd.clear();
```

```
      state = 1;
    }
    else
    {
      card = content.substring(1);
      LcdPrint();
      lcd.setCursor(9, 1);
      lcd.print(card1Balance);
      lcd.print(" Rs");
      delay(3000);
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Please Recharge");
      delay(2000);
      lcd.clear();
      state = 0;
    }
  }
  else if (content.substring(1) == "44 88 D9 73")
  {
    if (card2Balance >= 500)
```

```
{
  lcdPrint();
  card2Balance = card2Balance - 500;
  lcd.setCursor(9, 1);
  lcd.print(card2Balance);
  delay(2000);
  lcd.clear();
  state = 1;
}
else
{
  card = content.substring(1);
  LcdPrint();
  lcd.setCursor(9, 1);
  lcd.print(card2Balance);
  lcd.print(" Rs");
  delay(3000);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Please Recharge");
  lcd.clear();
```

```
      delay(2000);
      state = 0;
    }
  }
  else   {
    digitalWrite(buzzerPin, HIGH);
    lcd.setCursor(0, 0);
    lcd.print("Unknown Vehicle");
    lcd.setCursor(0, 1);
    lcd.print("Access denied");
    delay(1500);
    lcd.clear();
    digitalWrite(buzzerPin, LOW);
  }
}


void KeyPad()
{
  byte KState = KP2.Key_State();

  if (KState == PRESSED)
```

```
  {
    Key = KP2.Getkey();
    if (Key)
    {
      if (Key == 'A')
      {
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Recharging Mode.");
        lcd.setCursor(0, 1);
        lcd.print("...............");
        delay(1500);
        lcd.clear();
        recharge = 0;
      }
    }
  }
}


void clearData()
{
```

```
  while (data_count != 0)
  {
    Data[data_count--] = 0;
  }
  return;
}

void reCharge()
{
  lcd.setCursor(0, 0);
  lcd.print ("Enter the amount");
  byte KState = KP2.Key_State();
  if (KState == PRESSED)
  {
    Key = KP2.Getkey();
    if (Key)
    {
      if (Key == 'D')
      {
        if (card == "84 E6 E0 1F")
        {
```

```
    num1 = Data;
    card1Balance = num1.toInt() + card1Balance;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Your current");
    lcd.setCursor(0, 1);
    lcd.print("balance: ");
    lcd.setCursor(9, 1);
    lcd.print (card1Balance);
    lcd.print(" Rs.");
    delay(3000);
    clearData();
    lcd.clear();
    recharge = 1;
  }
  else if (card == "44 88 D9 73")
  {
    num2 = Data;
    card2Balance = num2.toInt() + card2Balance;
    lcd.clear();
    lcd.setCursor(0, 0);
```

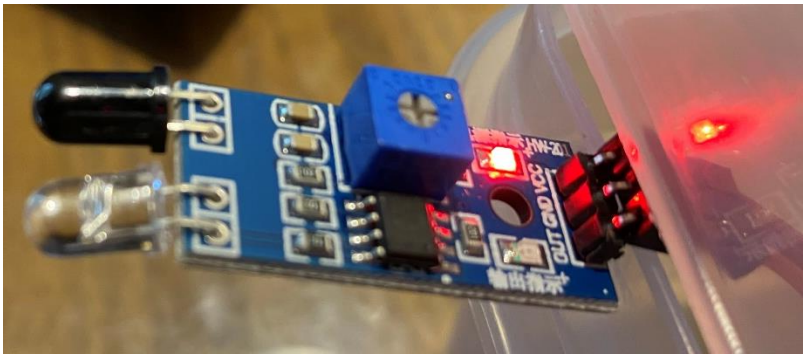
```
          lcd.print("Your current");
          lcd.setCursor(0, 1);
          lcd.print("balance: ");
          lcd.setCursor(9, 1);
          lcd.print (card2Balance);
          lcd.print(" Rs.");
          delay(3000);
          clearData();
          lcd.clear();
          recharge = 1;
        }
      }
      else
      {
        Data[data_count] = Key;
        lcd.setCursor(data_count, 1);
        lcd.print(Data[data_count]);
        data_count++;
      }
    }
}
```

```
}
void lcdPrint()
{
  digitalWrite(buzzerPin, HIGH);
  delay(200);
  digitalWrite(buzzerPin, LOW);
  delay(100);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("  Successfully");
  lcd.setCursor(0, 1);
  lcd.print(" paid your bill");
  delay(1500);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Your Remaining");
  lcd.setCursor(0, 1);
  lcd.print("balance: ");
}

void LcdPrint()
```

```
{
  digitalWrite(buzzerPin, HIGH);
  delay(200);
  digitalWrite(buzzerPin, LOW);
  delay(100);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("  Your balance");
  lcd.setCursor(0, 1);
  lcd.print(" is insufficent");
  delay(1500);
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Your Remaining");
  lcd.setCursor(0, 1);
  lcd.print("balance: ");
}
```

Output Screenshots

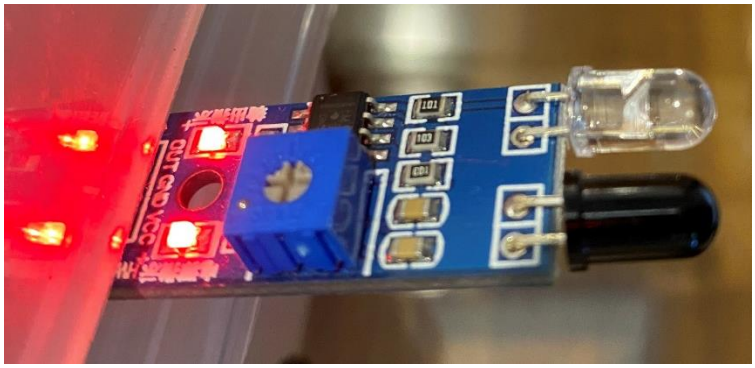


Infrared (IR) Sensor used to sense a vehicle and open and close the gate

Sensor has two values one is HIGH or 1 and other is LOW or 0. Right now this IR sensor has value as HIGH or 1

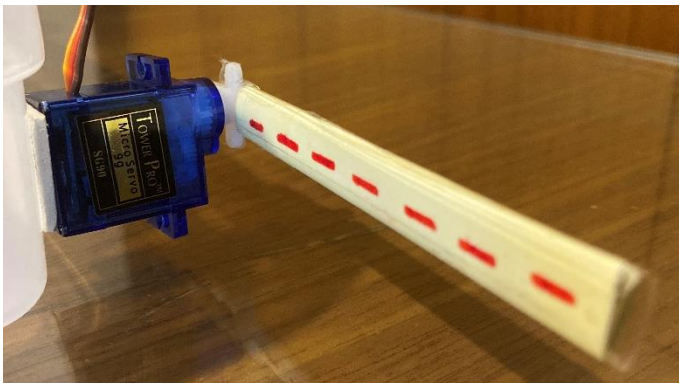
Infrared (IR) Sensor used to sense a vehicle and open and close the gate

Sensor has two values one is HIGH or 1 and other is LOW or 0. Right now this IR sensor has value as HIGH or 1



This message is shown when there is a vehicle in front of IR sensor

Welcome message is showed when you enter the toll booth



Servo Motor used to open and close the gate.

In this figure the gate is closed i.e. the transaction is not yet completed.

This message will be shown after the vehicle is detected and proceed with the transaction

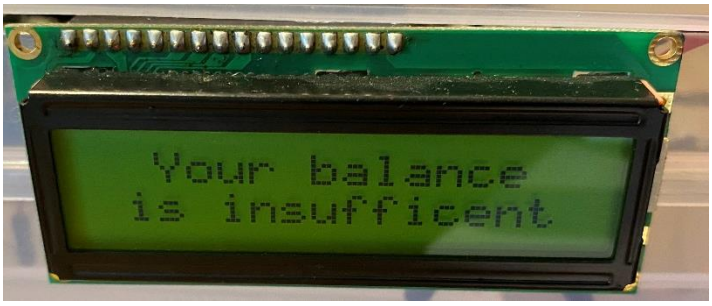
These are the RFID tags which will be read by the RFID card reader

These tags have different UID number

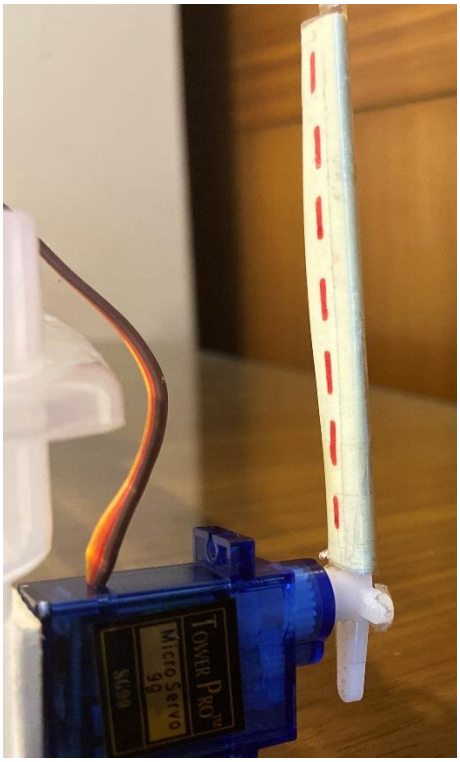
When the tag is scanned, and the transaction has been successfully completed

When the tag is scanned, and the toll is deduced, and then available balance is showed

When the tag is scanned, and there is insufficient balance in the card



The Gate is opened when the transaction of toll is done, and the vehicle is about to exit



Application of Project

This project deals with the simplification of procedure followed by passengers to pay toll at toll collection booths, like

- making it automated
- vehicle theft detection
- rule breaking etc.

All these activities are carried out using single smart card (RFID tag), thus saving the efforts of carrying money and records manually.

With the help of one single smart card we can use this card at various place like toll, parking lot, offices, industrial areas, societies, etc. where all the data will be stored in one single place.

Using this system with an another parking lot system, where the person will scan the RFID tag and he will be allotted with the parking number or slot number where he can park the car, reducing the wastage of time and fuel consumption.

In offices, industrial areas and societies we can use this system to just note the entry and exit of the registered vehicle and if there is someone who is not a resident of the society or employee he need to request the person he is going to visit and add his details in the database temporarily for some time by giving an entry and exit time so that he can enter the office or society.

We can even use this system in the elevators of societies. Now in this corona situation it is very risky to touch anywhere, instead of touching we can scan our smart card in elevator and it will automatically take us to the floor which is saved in database for the respective smart card.

We can even stop the corruption by using the same smart card to pay fines or buying things.

YouTube Link

https://youtu.be/ddawQMvD7SU