**A PROJECT REPORT**
**ON**

# BIOMETRIC CAR IGNITION MECHANISM

**SUBMITTED TO**
**SBSSU FEROZEPUR(PUNJAB)**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF**

**BACHELOR OF TECHNOLOGY**

**In**

**ELECTRONICS & COMMUNICATION ENGINEERING**

**SUBMITTED BY**
**DIPESH KUMAR MISHRA (1908433)**
**ANIKET KUMAR PANDEY (1809257)**
**JULY-DECEMBER**
**2021**

**UNDER GUIDANCE OF: -Dr. AMIT GROVER**


**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**
**SHAHEED BHAGAT SINGH UNIVERSITY**
**FEROZEPUR, PUNJAB (INDIA)**

**A PROJECT REPORT**
**ON**

# BIOMETRIC CAR IGNITION MECHANISM

**SUBMITTED TO**
**SBSSU FEROZEPUR(PUNJAB)**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF THE DEGREE OF**

**BACHELOR OF TECHNOLOGY**

**In**

**ELECTRONICS & COMMUNICATION ENGINEERING**

**SUBMITTED BY**
**DIPESH KUMAR MISHRA (1908433)**
**ANIKET KUMAR PANDEY (1809257)**
**JULY-DECEMBER**
**2021**

**UNDER GUIDANCE OF: -Dr. AMIT GROVER**


**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**
**SHAHEED BHAGAT SINGH UNIVERSITY**
**FEROZEPUR, PUNJAB (INDIA)**

# SHAHEED BHAGAT SINGH STATE UNIVERSITY

**FEROZEPUR PUNJAB**

***Department of Electronics and Communication Engg.***

## CERTIFICATE

*This is to certify that **"DIPESH KUMAR MISHRA" (1908433)** and **"ANIKET KUMAR PANDEY" (1809257)** of final year student of Electronics and Communication Engineering has submitted **MAJOR PROJECT** report entitled "**BIOMETRIC CAR IGINATION MECHANISM"** in partial fulfilment for the award of Bachelor of Technology Degree of Shaheed Bhagat Singh University, Ferozepur in session July -December 2021. It has been found to be satisfactory and hereby approved for the submission.*

**Dr. RAJNI**
(H.O.D. of ECE deptt.)

**Dr. AMIT GROVER**
(Project Guide)

# DECLARATION

I the undersigned solemnly declare that the report of the project work entitled **BIOMETRIC CAR IGINATION MECHANISM**, is based my own work carried out during the course of my study under the supervision of **Dr. AMIT GROVER**

I assert that the statements made and conclusions drawn are an outcome of the project work. I further declare that to the best of my knowledge and belief that the project report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University.

| | |
|---|---|
| _____ | _____ |
| DIPESH KUMAR MISHRA | ANIKET KUMAR PANDEY |
| 1908433 | 1809257 |

# CERTIFICATE BY GUIDE

This to certify that the report of the project submitted is the outcome of the project work entitled **BIOMETRIC CAR IGINATION MECHANISM** carried out by **DIPESH KUMAR MISHRA (1908433)** and **ANIKET KUMAR PANDEY (1809257)**

Carried by under my guidance for the award of Degree in Bachelor of Technology of Shaheed Bhagat Singh State University Punjab.

To the best of my knowledge the report

i) Embodies the work of the candidate him/herself,
ii) Has duly been completed,
iii) Fulfils the requirement of the ordinance relating to the B-Tech degree of the University and
iv) Is up to the desired standard for the purpose of which is submitted.

Dr. AMIT GROVER
Assistant Professor
Electronics and Communication
SBSSU Ferozepur Punjab

# CERTIFICATE BY SUPERVISOR

This to certify that the report of the project submitted is the outcome of the project work entitled **BIOMETRIC CAR IGINATION MECHANISM** carried out by **DIPESH KUMAR MISHRA (1908433)** and **ANIKET KUMAR PANDEY (1809257)**

Carried by under my supervision for the award of Degree in Bachelor of Technology of Shaheed Bhagat Singh State University Punjab.

To the best of my knowledge the report

i) Embodies the work of the candidate him/herself,
ii) Has duly been completed,
iii) Fulfils the requirement of the ordinance relating to the B-Tech degree of the University and
iv) Is up to the desired standard for the purpose of which is submitted.

Mr. CHAKSHU GOEL
Assistant Professor
Electronics and Communication
SBSSU Ferozepur Punjab

# ACKNOWLEDGEMENT

Place:
Date:

On this great occasion of accomplishment of our project on **BIOMETRIC CAR IGINATION MECHANISH**, we would like to sincerely express our gratitude to **Dr. AMIT GROVER**, who has been supported through the completion of this project.

We would also be thankful to our HOD. Dr. RAJNI of Electronic and Communication Engineering for providing all the required facilities in completion of this project.

Finally, as one of the team members, I would like to appreciate all my group members for their support and coordination, I hope we will achieve more in our future endeavors.

DIPESH KUMAR MISHRA (1908433)

ANIKET KUMAR PANDEY (1809257)

# TABLE OF CONTENTS

# BIOMETRIC CAR IGNITION MECHANISM

**Introduction:** Nowadays most of the car comes with keyless entry and push-button ignition system, in which you only need to carry the key in your pocket and just need to put your finger on the capacitive sensor on the door handle to open the car door. Here in this project, we are adding a few more security features to this system by using RFID and Fingerprint sensor. RFID sensor will validate the license of the user and the fingerprint sensor will only allow an authorized person in the vehicle.

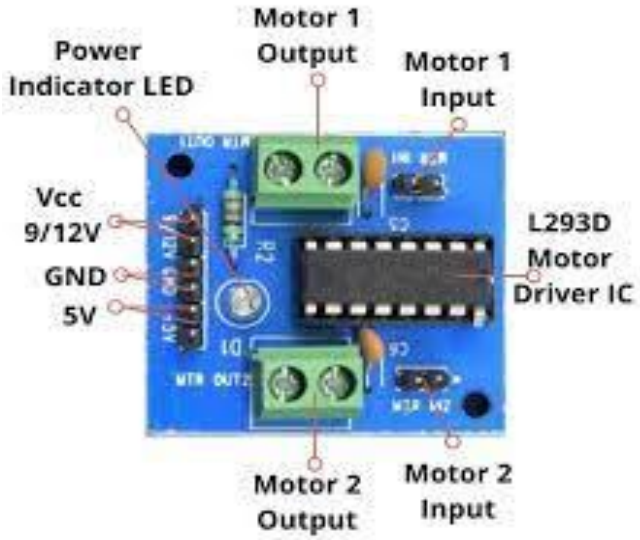
For this **BIOMETRIC CAR IGNITION MECHANISM**, we are using **Arduino with an R305 Fingerprint sensor and an EM18 RFID reader**.

**COMPONENT USED: -**

- R305 Fingerprint sensor
- EM18 RFID reader
- 16*2 Alphanumeric LCD
- DC motors
- L293D Motor driver IC
- Connecting wires

**THERE WOULD BE USAGE OF ARDUINO UNO AS THE EMBEDDED PLATFORM AND SEVERAL JUMP CABLES WOULD BE USED TO CONNECT THE SYSTEM OF THE WORKING MODLE.**

## 1.L293D MOTOR DRIVER IC:-


# Features

- Can be used to run Two DC motors with the same IC.
- Speed and Direction control is possible
- Motor voltage Vcc2 (Vs): 4.5V to 36V
- Maximum Peak motor current: 1.2A
- Maximum Continuous Motor Current: 600mA
- Supply Voltage to Vcc1(vss): 4.5V to 7V
- Transition time: 300ns (at 5Vand 24V)
- Automatic Thermal shutdown is available
- Available in 16-pin DIP, TSSOP, SOIC packages

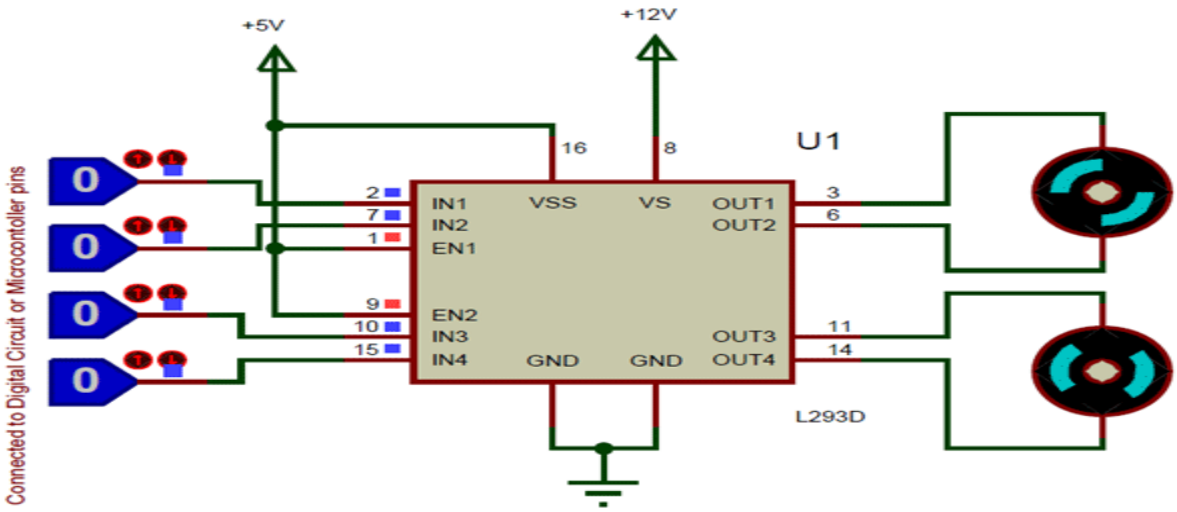
# Where to use L293D IC

The L293D is a popular 16-Pin **Motor Driver IC**. As the name suggests it is mainly used to drive motors. A single **L293D IC** is capable of running two DC motors at the same time; also the direction of these two motors can be controlled independently. So if you have motors which has operating voltage less than 36V and operating current less than 600mA, which are to be controlled by digital circuits like Op-

Amp, 555 timers, digital gates or even Microcontrollers like Arduino, PIC, ARM etc.. this IC will be the right choice for you.

## How to use a L293D Motor Driver IC

Using this **L293D motor driver IC** is very simple. The IC works on the principle of **Half H-Bridge**, let us not go too deep into what H-Bridge means, but for now just know that H bridge is a set up which is used to run motors both in clock wise and anti clockwise direction. As said earlier this IC is capable of running two motors at the any direction at the same time, the circuit to achieve the same is shown below.




All the Ground pins should be grounded. There are two power pins for this IC, one is the Vss(Vcc1) which provides the voltage for the IC to work, this must be connected to +5V. The other is Vs(Vcc2) which provides voltage for the motors to run, based on the specification of your motor you can connect this pin to anywhere between 4.5V to 36V, here I have connected to +12V.

The Enable pins (Enable 1,2 and Enable 3,4) are used to Enable Input pins for Motor 1 and Motor 2 respectively. Since in most cases we will be using both the motors both the pins are held high by default by connecting to +5V supply. The input pins Input 1,2 are used to control the motor 1 and Input pins 3,4 are used to control the Motor 2. The input pins are connected to the any Digital circuit or microcontroller to

control the speed and direction of the motor. You can toggle the input pins based on the following table to control your motor.

## Applications

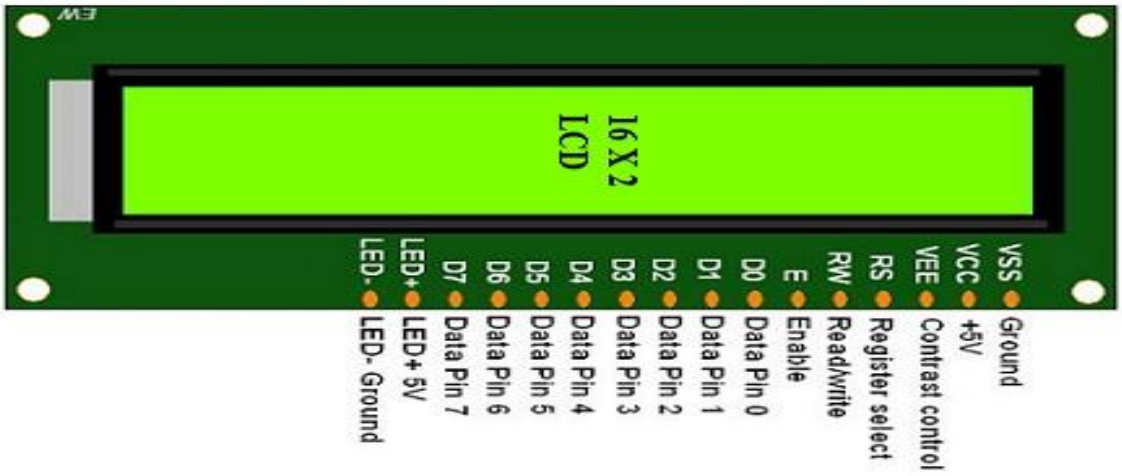
- Used to drive high current Motors using Digital Circuits
- Can be used to drive Stepper motors
- High current LED's can be driven
- Relay Driver module (Latching Relay is possible)

## 2.LCD 16×2:-

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.


**The 16×2 LCD pinout is shown below**.

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.
- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.




## Features of LCD16x2

The features of this LCD mainly include the following.

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight

- Every character can be built with a 5×8 pixel box
- The alphanumeric LCDs alphabets & numbers
- Is display can work on two modes like 4-bit & 8-bit

## Registers of LCD

A 16×2 LCD has two registers like data register and command register. The RS (register select) is mainly used to change from one register to another. When the register set is '0', then it is known as command register. Similarly, when the register set is '1', then it is known as data register.

## Command Register

The main function of the command register is to store the instructions of command which are given to the display. So that predefined tasks can be performed such as clearing the display, initializing, set the cursor place, and display control. Here commands processing can occur within the register.

## Data Register

The main function of the data register is to store the information which is to be exhibited on the LCD screen. Here, the ASCII value of the character is the information which is to be exhibited on the screen of LCD. Whenever we send the information to LCD, it transmits to the data register, and then the process will be starting there. When register set =1, then the data register will be selected.

## LCD 16×2 Arduino

Please refer to this link to know more about-How to Interface Liquid Crystal Display using An Arduino.

Thus, this is all about LCD 16×2 datasheet, which includes what is a 16X2 LCD, pin configuration, working principle, and its applications. The main advantages of this LCD device include power consumption is less and low cost. The main disadvantages of this LCD device include it occupies a large area, slow devices and also lifespan of these devices will be reduced due to direct current. So these LCDs use AC supply with less than 500Hz frequency. Here is a question for you

## 3.ARDUINO UNO: -


**Arduino Uno** is a microcontroller board based on 8-bit ATmega328P microcontroller. Along with ATmega328P, it consists other components such as crystal oscillator, serial communication, voltage regulator, etc. to support the microcontroller. Arduino Uno has 14 digital input/output pins (out of which 6 can be used as PWM outputs), 6 analog input pins, a USB connection, A Power barrel jack, an ICSP header and a reset button.

## How to use Arduino Board

The 14 digital input/output pins can be used as input or output pins by using pinMode(), digitalRead() and digitalWrite() functions in arduino programming. Each pin operate at 5V and can provide or receive a maximum of 40mA current, and has an internal pull-up resistor of 20-50 KOhms which are disconnected by default. Out of these 14 pins, some pins have specific functions as listed below:

- **Serial Pins 0 (Rx) and 1 (Tx):** Rx and Tx pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB to TTL serial chip.
- **External Interrupt Pins 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- **PWM Pins 3, 5, 6, 9 and 11:** These pins provide an 8-bit PWM output by using analogWrite() function.

- **SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK):** These pins are used for SPI communication.

- **In-built LED Pin 13:** This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.

Along with 14 Digital pins, there are 6 analog input pins, each of which provide 10 bits of resolution, i.e. 1024 different values. They measure from 0 to 5 volts but this limit can be increased by using AREF pin with analog Reference() function.

- Analog pin 4 (SDA) and pin 5 (SCA) also used for TWI communication using Wire library.

Arduino Uno has a couple of other pins as explained below:

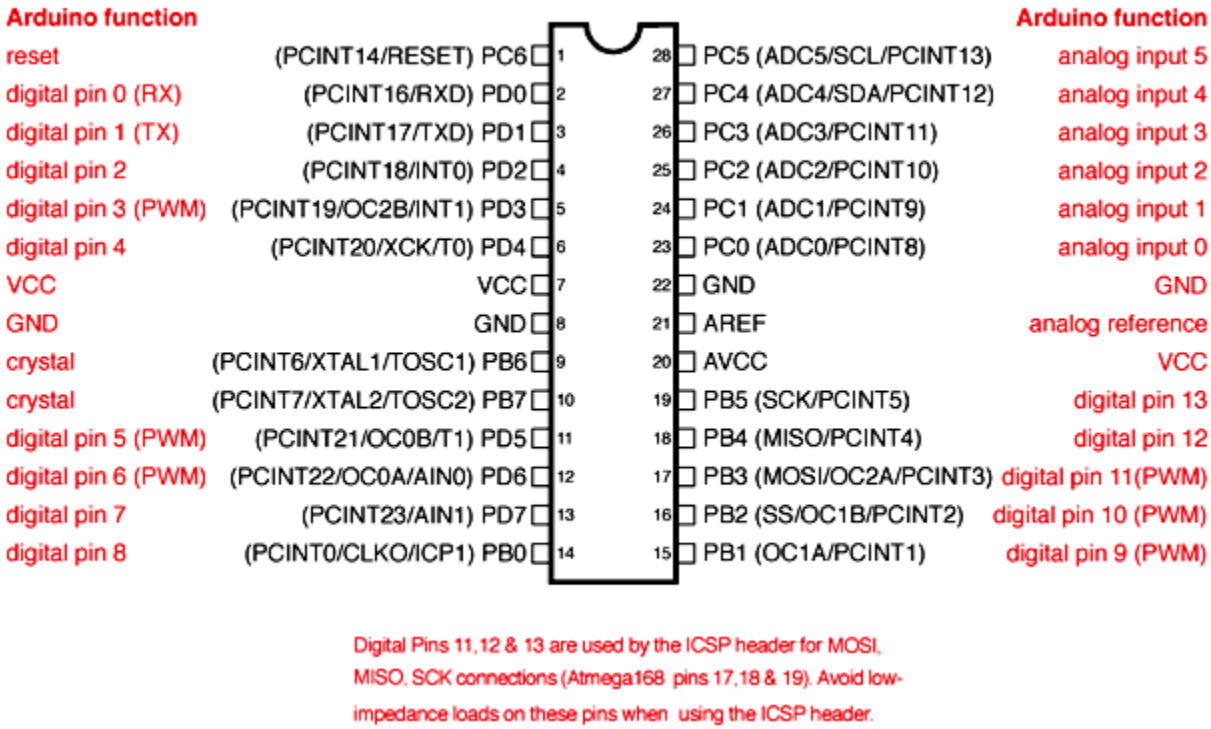
- **AREF:** Used to provide reference voltage for analog inputs with analog Reference () function.
- **Reset Pin:** Making this pin LOW, resets the microcontroller.

## Communication

Arduino can be used to communicate with a computer, another Arduino board or other microcontrollers. The ATmega328P microcontroller provides UART TTL (5V) serial communication which can be done using digital pin 0 (Rx) and digital pin 1 (Tx). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The ATmega16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. There are two RX and TX LEDs on the arduino board which will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (not for serial communication on pins 0 and 1). A SoftwareSerial library allows for serial communication on any of the Uno's digital pins. The ATmega328P also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus.

## Arduino Uno to ATmega328 Pin Mapping

When ATmega328 chip is used in place of Arduino Uno, or vice versa, the image below shows the pin mapping between the two.




## Software

Arduino IDE (Integrated Development Environment) is required to program the Arduino Uno board. [Download it from here.]

## Programming Arduino

Once arduino IDE is installed on the computer, connect the board with computer using USB cable. Now open the arduino IDE and choose the correct board by selecting Tools>Boards>Arduino/Genuino Uno, and choose the correct Port by selecting Tools>Port. Arduino Uno is programmed using Arduino programming language based on Wiring. To get it started with Arduino Uno board and blink the built-in LED, load the example code by selecting Files>Examples>Basics>Blink. Once the example code (also shown below) is loaded into your IDE, click on the

'upload' button given on the top bar. Once the upload is finished, you should see the Arduino's built-in LED blinking. Below is the example code for blinking:

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}


// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

## Applications

- Prototyping of Electronics Products and Systems
- Multiple DIY Projects.
- Easy to use for beginner level DIYers and makers.
- Projects requiring Multiple I/O interfaces and communications.

## 4.EM18 RFID Reader Module

RFID stands for Radio frequency identification. It refers to a technology, where digital data is encoded in RFID tags and they can be decoded by an RFID reader using radio waves. RFID is similar to barcoding in which data from a tag is decoded by a device. RFID technology is used in various applications like Security system, Employee attendance system, RFID Door Lock, RFID Based Voting Machine, Toll Collection System, etc.

**EM18 Reader** is a module that can read the ID information stored in the RFID tags. The RFID tags stores a 12-digit unique number which can be decoded by an EM18 reader module, when the tag comes in range with the Reader. This module operates at a frequency of 125 kHz, which is having an inbuilt antenna, and it is operated using

a 5-volt DC power supply. It gives a serial data output, and it has a range of 8-12 cm. The serial communication parameters are 8 data bits, 1 stop bit, and 9600 baud rates.


## Pin description:

VCC: 4.5- 5V DC voltage input

GND: Ground pin

Buzzer: Buzzer or LED pin

TX: Serial data Transmitter pin of EM18 for RS232 (Output)

SEL: This must be HIGH for using RS232 (LOW if using WEIGAND)

Data 0: WEIGAND data 0

Data 1: WEIGAND data 1

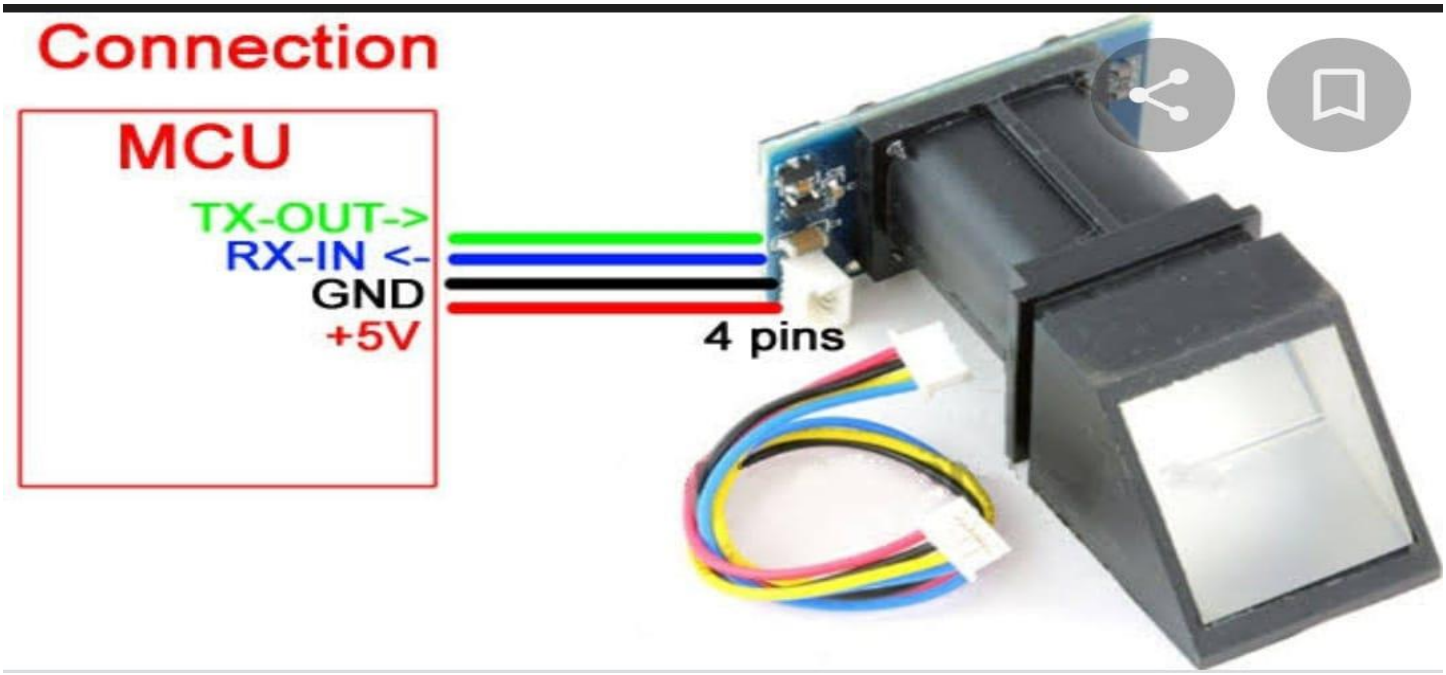
**Find out RFID Tag Unique 12-digit Code using Arduino:-**Before programming the Arduino for **Arduino car ignition system**, first, we need to find out the 12-digit RFID tag unique code. As we discussed before, RFID tags contain a 12-digit unique code and it can be decoded by using an RFID reader. When we swipe the RFID tag near the Reader, the Reader will give the unique codes via the output serial port. First, connect the Arduino to the RFID reader as per circuit diagram and then upload below given code to Arduino.

```
int count = 0;
char card_no[12];
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  if(Serial.available())
  {
    count = 0;
    while(Serial.available() && count < 12)
    {
      card_no[count] = Serial.read();
      count++;
      delay(5);
    }
    Serial.print(card_no);
  }
}
```

After successfully uploading the code, open the serial monitor, and set the baud rate to 9600. Then swipe the card near the Reader. Then the 12-digit code will start to display on the serial monitor. Do this process for all the used RFID tags and note it down for future references.

## Finger Print Sensor Module

**Finger Print Sensor Module or Finger Print Scanner** is a module that captures finger's print image and then converts it into the equivalent template and saves them into its memory on selected ID (location) by Arduino. Here all the process is commanded by Arduino like taking an image of a fingerprint, convert it into templates, and storing location, etc.
We can use the same R305 sensor to build a voting machine, attendance system, security system, etc. You can check all the finger print based projects here.




## Enrolling Fingerprints to Sensor:

Before proceeding with the program, we need to install the required libraries for the fingerprint sensor. Here we have used "***Adafruit_Fingerprint.h***" for using the R305 fingerprint sensor. So first of all, download the library using the link given below:

- *Adafruit fingerprint sensor library*

After successful download, In the Arduino IDE, go to ***File > Tools > Include library > Add .zip library*** and then select the zip file location to install the library.

After successful library installation, follow the steps given below to enroll a new fingerprint in the sensor memory.

**1.** In the Arduino IDE, go to **File** > **Examples** > **Adafruit Fingerprint Sensor Library** > **Enroll.**

**2.** Upload the code to the Arduino, and open the Serial monitor at a baud rate of 9600.

***Important: Change the Software serial pin in the program to Software Serial mySerial(12, 11).***

**3.** You should enter an ID for the fingerprint in which you want to store your fingerprint. As this is my first fingerprint, I typed 1 in the top left corner, and then, click the **Send** button.

**4.** Then the light on the fingerprint sensor will blink which indicates that you should place your finger on the sensor and after that follow the steps showing on serial monitor till it acknowledges you for successful enrolment.

## Programming for RFID Keyless Ignition

Complete code for this **Biometric Ignition System** is given at the end of the tutorial. Here we are explaining a few important parts of the code.

```
#include <Adafruit_Fingerprint.h>
#include <LiquidCrystal.h>
SoftwareSerial mySerial(12,11);
Adafruit_Fingerprint finger =
Adafruit_Fingerprint(&mySerial);
```

The first thing is to include all the required libraries. Here in my case, I have included "***Adafruit_Fingerprint.h***" for using R305 fingerprint sensor. Then configure the serial port in which the fingerprint sensor will be connected. In my case, I have declared 12 as RX Pin and 11 as a TX pin.

In the next step, declare all the variables, which will be used throughout the code. Then define the LCD connection pins with Arduino followed by the declaration of an object of *LiquidCrystal* class.

```
char input[12];
int count = 0;
int a = 0;
const int rs = 6, en = 7, d4 = 2, d5 = 3, d6 = 4, d7 = 5;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

Next, inside the ***loop (),*** code is written to get the unique 12-digit codes of the RFID tags and they are stored in an array. Here the elements of the array will be matched with the stored unique codes in the memory, to get the authenticated person details.

```
count = 0;
  while (Serial.available() && count < 12)
  {
   input[count] = Serial.read();
   count++;
   delay(5);
  }
```

Then, the received array is compared with the stored tag codes. If the code is matched, then the license is considered as valid, which allows the user to put valid fingerprint. Otherwise, it will show an invalid license.

```
if ((strncmp(input, "3F009590566C", 12) == 0)
&& (a == 0))
    {
     lcd.setCursor(0, 0);
     lcd.print("License Valid      ");
     lcd.setCursor(0, 1);
     lcd.print("Welcome          ");
     delay(1000);
     a = 1;
     fingerprint();
    }
```

In the next step, a function ***getFingerprintID*** is written which will return a valid fingerprint ID for an already enrolled fingerprint.

```
int getFingerprintID()
{
  uint8_t p = finger.getImage();
  if (p != FINGERPRINT_OK)  return -1;
  p = finger.image2Tz();
  if (p != FINGERPRINT_OK)  return -1;
  p = finger.fingerFastSearch();
  if (p != FINGERPRINT_OK)  return -1;
  return finger.fingerID;
}
```

Inside function ***fingerprint ()***, which is called after successful RFID match, ***getFingerprintID*** function is called to get a valid fingerprint ID. Then it is compared using the *if-else loop* to get the information regarding authenticated person data and if data is matched, then the vehicle is ignited, otherwise, it will prompt for the wrong Fingerprint.

```
int fingerprintID = getFingerprintID();
      delay(50);
      if (fingerprintID == 1)
      {
        lcd.setCursor(0, 0);
        lcd.print("Access Granted     ");
        lcd.setCursor(0, 1);
        lcd.print("Vehicle Started    ");
        digitalWrite(9,HIGH);
        digitalWrite(10,LOW);
        while(1);
      }
```

So, this is how this **RFID Car Ignition System** works which adds two layers of security to your car.

## How does Our Biometric car ignition mechanism work?

1.First, it is switched on by giving some authentication.
2.The whole circuit is based on the biometric authentication thus to get the engine started we need to imply the biometric signature.
3.There would be proper authentication that is using RFID that is the initial security block.
4.The RFID would sense the 12-digit code RF code that would be there in the transponder that we would be having with us to ignite the engine of the vehicle.
5. After authenticating the RF signal/code from the transceiver/transponder the biometric sensor would be switched on.

6. Then the owner will be asked to put his Finger impression to the sensor, as when the sensor would read the signature it would command the micro controlling unit to switch on the engine of the system.
7. This would create the ease of the maintenance of the vehicle and enhance the security of the vehicle from all kind of theft.

**This mechanism ensures safety and when used in EV i.e the Electric vehicle would be very environment friendly and would not bound you to carry unnecessary keys that sometimes harm your stuffs like phone wallets kept in your pocket. Along with that it would give you a feel of goodness and rejoice that your vehicle is advance and secure from different kind of scavenges.**

## Complete Code are given below.

```
#include <Adafruit_Fingerprint.h>
#include <LiquidCrystal.h>
char input[12];
int count = 0;
int a = 0;
const int rs = 6, en = 7, d4 = 2, d5 = 3, d6 = 4, d7 = 5;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
SoftwareSerial mySerial(12,11);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
void setup()
{
pinMode(9,OUTPUT);
pinMode(10,OUTPUT);
digitalWrite(9,LOW);
digitalWrite(10,LOW);
Serial.begin(9600);
lcd.begin(16, 2);
lcd.setCursor(0, 0);
lcd.print("   WELCOME TO       ");
lcd.setCursor(0, 1);
lcd.print("  CIRCUIT DIGEST       ");
delay(2000);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Please swipe          ");
lcd.setCursor(0, 1);
lcd.print("Your License          ");
```

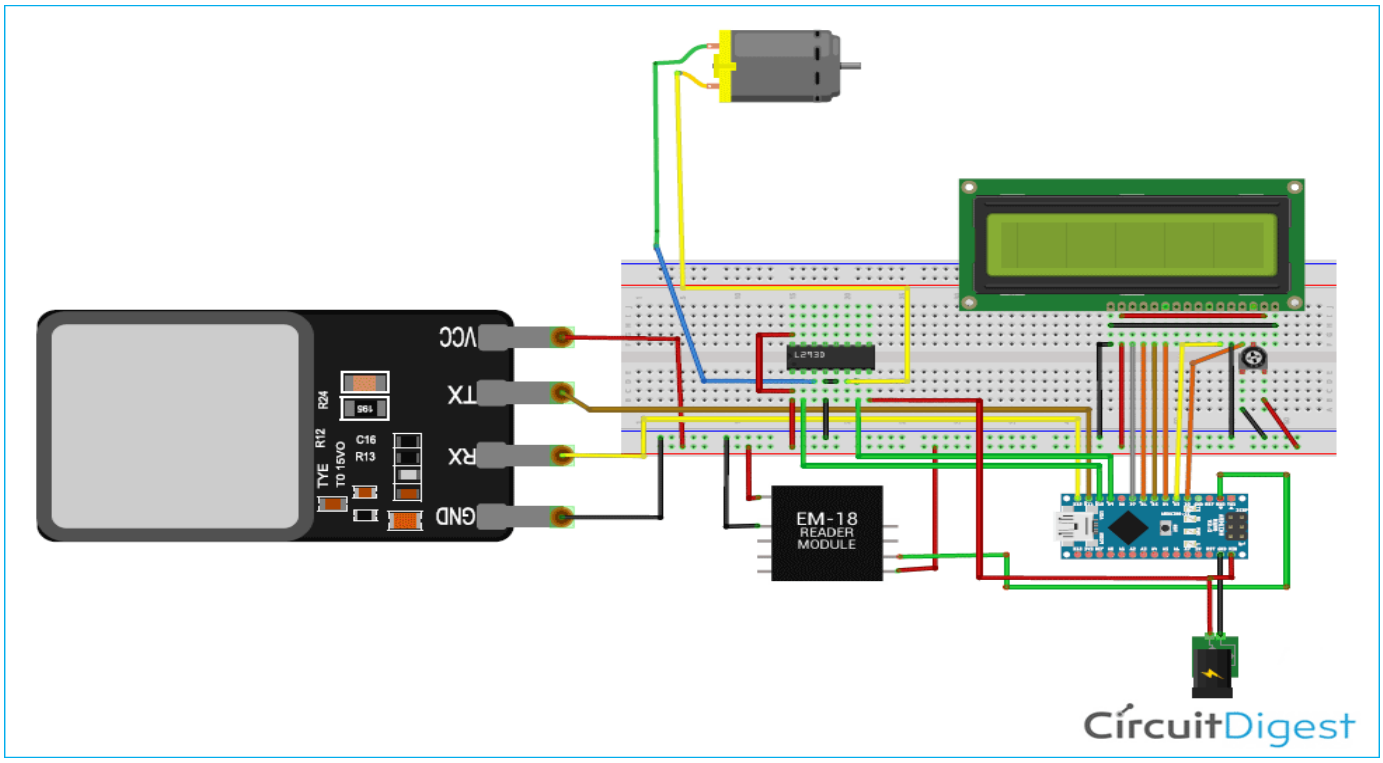
```
}
void loop()
{
if (Serial.available())
{
count = 0;
while (Serial.available() && count < 12)
{
input[count] = Serial.read();
count++;
delay(5);
}
if (count == 12)
{
if ((strncmp(input, "3F009590566C", 12) == 0) && (a == 0))
{
lcd.setCursor(0, 0);
lcd.print("License Valid        ");
lcd.setCursor(0, 1);
lcd.print("Welcome              ");
delay(1000);
a = 1;
fingerprint();
}
else if ((strncmp(input, "0B0028883E95", 12) == 0) && (a == 0))
{
lcd.setCursor(0, 0);
lcd.print("License Valid        ");
lcd.setCursor(0, 1);
lcd.print("Welcome              ");
delay(1000);
a = 1;
fingerprint();
}
else
{
if (a != 1)
{
lcd.setCursor(0, 0);
lcd.print("License Invalid      ");
lcd.setCursor(0, 1);
lcd.print("Try Again!!!         ");
delay(2000);
lcd.clear();
```

```
72.lcd.setCursor(0, 0);
73.lcd.print("Please swipe          ");
74.lcd.setCursor(0, 1);
75.lcd.print("Your License         ");
76.}
77.}
78.}
79.}
80.}
81.int getFingerprintID()
82.{
83.uint8_t p = finger.getImage();
84.if (p != FINGERPRINT_OK)  return -1;
85.p = finger.image2Tz();
86.if (p != FINGERPRINT_OK)  return -1;
87.p = finger.fingerFastSearch();
88.if (p != FINGERPRINT_OK)  return -1;
89.return finger.fingerID;
90.}
91.void fingerprint()
92.{
93.finger.begin(57600);
94.while(a==1)
95.{
96.int fingerprintID = getFingerprintID();
97.delay(50);
98.if (fingerprintID == 1)
99.{
100.       lcd.setCursor(0, 0);
101.       lcd.print("Access Granted       ");
102.       lcd.setCursor(0, 1);
103.       lcd.print("Vehicle Started      ");
104.       digitalWrite(9,HIGH);
105.       digitalWrite(10,LOW);
106.       while(1);
107.       }
108.       else if (fingerprintID == 2)
109.       {
110.       lcd.setCursor(0, 0);
111.       lcd.print("Access Granted       ");
112.       lcd.setCursor(0, 1);
113.       lcd.print("Vehicle Started      ");
114.       digitalWrite(9,HIGH);
115.       digitalWrite(10,LOW);
```

```
116.        while(1);
117.        }
118.        else
119.        {
120.        lcd.setCursor(0, 0);
121.        lcd.print("Pls Place a        ");
122.        lcd.setCursor(0, 1);
123.        lcd.print("Valid Finger       ");
124.        }
125.        }
126.        }
```

## Connection

Now connect all the components and set all those things in a small enclosure with a battery, as you can see in the pictures below.




**THERE WOULD BE USAGE OF ARDUINO UNO AS THE EMBEDDED PLATFORM AND SEVERAL JUMP CABLES OOULD BE USED TO CONNECT THE SYSTEM OF THE WORKING MODLE.**

# ADVANTAGE:

1. **In the present scenario safety is one of the most important aspect that can be ensured by this model.**
2. **Piercy can be avoided using it.**
3. **Convenience will be increased in the automation.**
4. **The system would even more improvise and would become even more easy to use.**
5. **Chances of theft of the automobile can be reduced as due to complex authentication system and computation.**
6. **This system can even be adopted for defense use for compliance to advancement.**
7. **This could make revolution in the present mobilization system of the machines.**

# Result:

This system was designed to focus on the stating of the engine by the means of fingerprint. The reason for developing this system is to increase the security level and the robustness of the vehicles from day-to-day threatens. The user touches the fingerprint sensor and it authenticates the user, if the user is authorized then it automatically starts the engine. The sensor is directly connected to the engine, the wires are attached in such a way that it starts-up. The main reason for using this, it is low in cost and the fingerprint biometric which is used it cannot be matched of any two people. So, it results in the accurate result for verifying the owner of the vehicle who can use only access their own vehicle.

# Conclusion:

The main motive of implementing the fingerprint sensor for the two or four wheelers is to provide the security for vehicles. It enhances the level of security for vehicles. As the finger impression is a promising biometric design for recognizing it is used in case of both security and usability. This technology easily reduces the theft cases

## Future Scope

Extra authentication mechanism can be studied after installation of this system as it can provide end to end as well as complex security to the user and its allies onto which the system is used. In addition, to overcome unrecognized fingerprint problem, this system can be extended by developing a keypad password system as an alternative to ignite the vehicle.

**THERE CAN BE FURHTER ENUMERATION USING DIFFRECTION PATTERN ASSESMENT TO MAKE THE SYSTEM EVEN MORE STRONG AND RESELIENT**

## References:


[1] Omidiora E. O. "A Prototype of a Fingerprint based Ignition Systems in 216X Vol.62 No.2, pp. 164-171 © Euro Journals Publishing, Inc. 2011 http://www.eurojournals.com/ejsr.html

[2] Visa M. Ibrahim "Microcontroller Based Anti-Theft Security System using Journal of Engineering Research and Development e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com Volume 2, Issue 10, PP. 18-22, August 2012.

[3] S. Pavithra "GSM and GPS Survey of Bike Security system using Biometric" ECE Department, Saveetha School of Engineering, Chennai, India.

[4] Anil K. Jain, Arun Ross "An Introduction to Biometric Recognition" IEEE Transactions on Circuits and Systems for Video Technology, Vol. 14, No. 1, January 2004.

[5] Kiran Rana Gill, Joel Sachin "Vehicle Ignition using Fingerprint Sensor" IJIRST – International Journal for Innovative Research in Science & Technology| Volume 2 | Issue 12 | May 2016.

[6] https://www.instructables.com/id/Biometric-Arduino-Engine-Security-System

**CREATORS NAMES**: -DIPESH KUMAR MISHRA (1908433)

ANIKET KUMAR PANDEY (1809257)

**UNDER GUIDANCE OF**: -Dr. AMIT GROVER

**SIGNATURE: -**