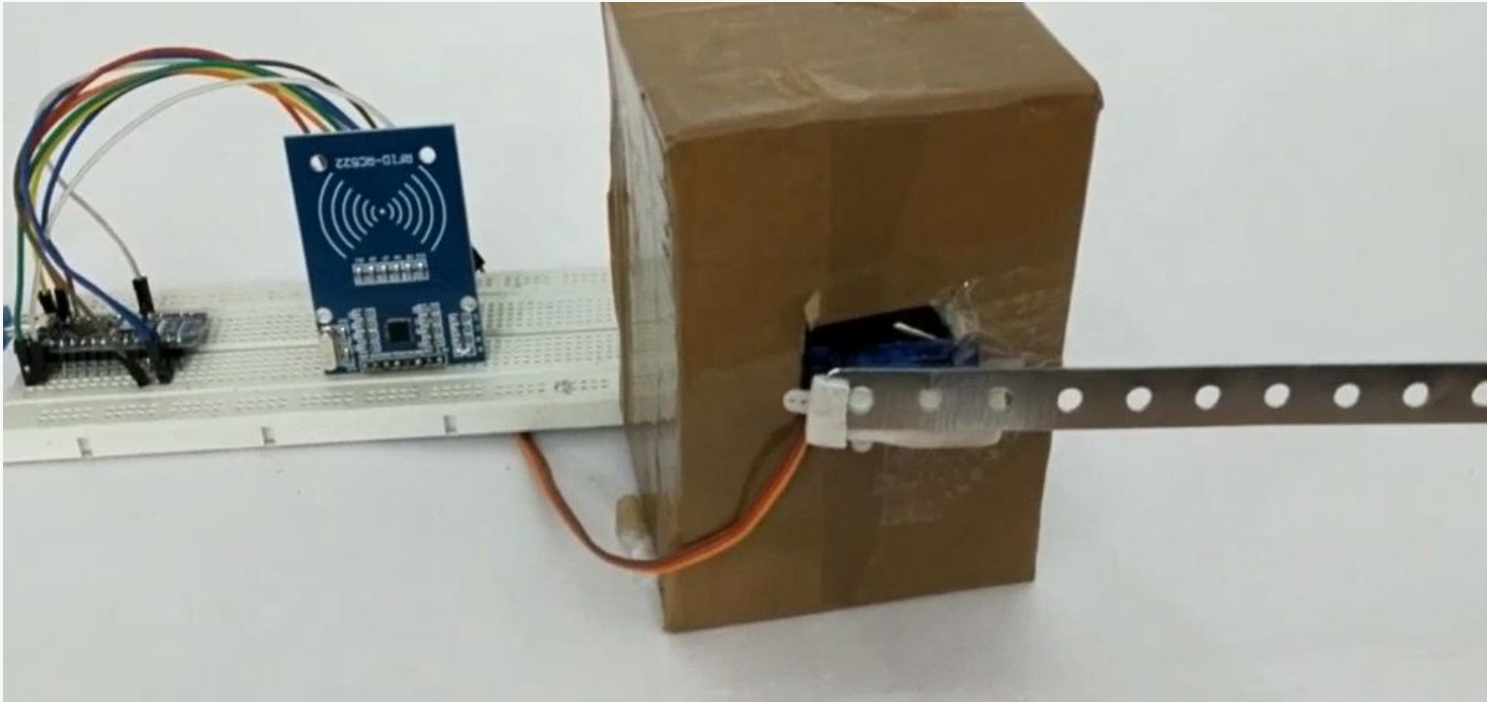


Automatic Door Lock using RFID Access module



RFID

- Radio-frequency identification (**RFID**) is the wireless use of electromagnetic fields to transfer data, for the purposes of automatically identifying and tracking tags attached to objects. The tags contain electronically stored information.
- A radio frequency identification reader (**RFID reader**) is a device used to gather information from an RFID tag, which is used to track individual objects. Radio waves are used to transfer data from the tag to a reader.



Servo Motor [SG90]

A **Servo motor** is an electrical device which can push or rotate an object with great precision. If you want to rotate an object at some specific angles or distance, then you use **Servo motor**. It is just made up of simple **motor** which run through **servo** mechanism.

It consists of three parts:

1. Controlled device
2. Output sensor
3. Feedback system



Working principle of Servo motor

1. A Servo consists of a Motor (DC or AC), a potentiometer, gear assembly and a controlling circuit.
2. First of all we use gear assembly to reduce RPM and to increase torque of motor.
3. Say at initial position of servo motor shaft, the position of the potentiometer knob is such that there is no electrical signal generated at the output port of the potentiometer.
4. Now an electrical signal is given to another input terminal of the error detector amplifier.
5. Now difference between these two signals, one comes from potentiometer and another comes from other source.

Working principle of Servo motor

6. It will be processed in feedback mechanism and output will be provided in term of error signal.
7. This error signal acts as the input for motor and motor starts rotating.
8. Now motor shaft is connected with potentiometer and as motor rotates so the potentiometer and it will generate a signal.
9. So as the potentiometer's angular position changes, its output feedback signal changes. After sometime the position of potentiometer reaches at a position that the output of potentiometer is same as external signal provided.

Working of project

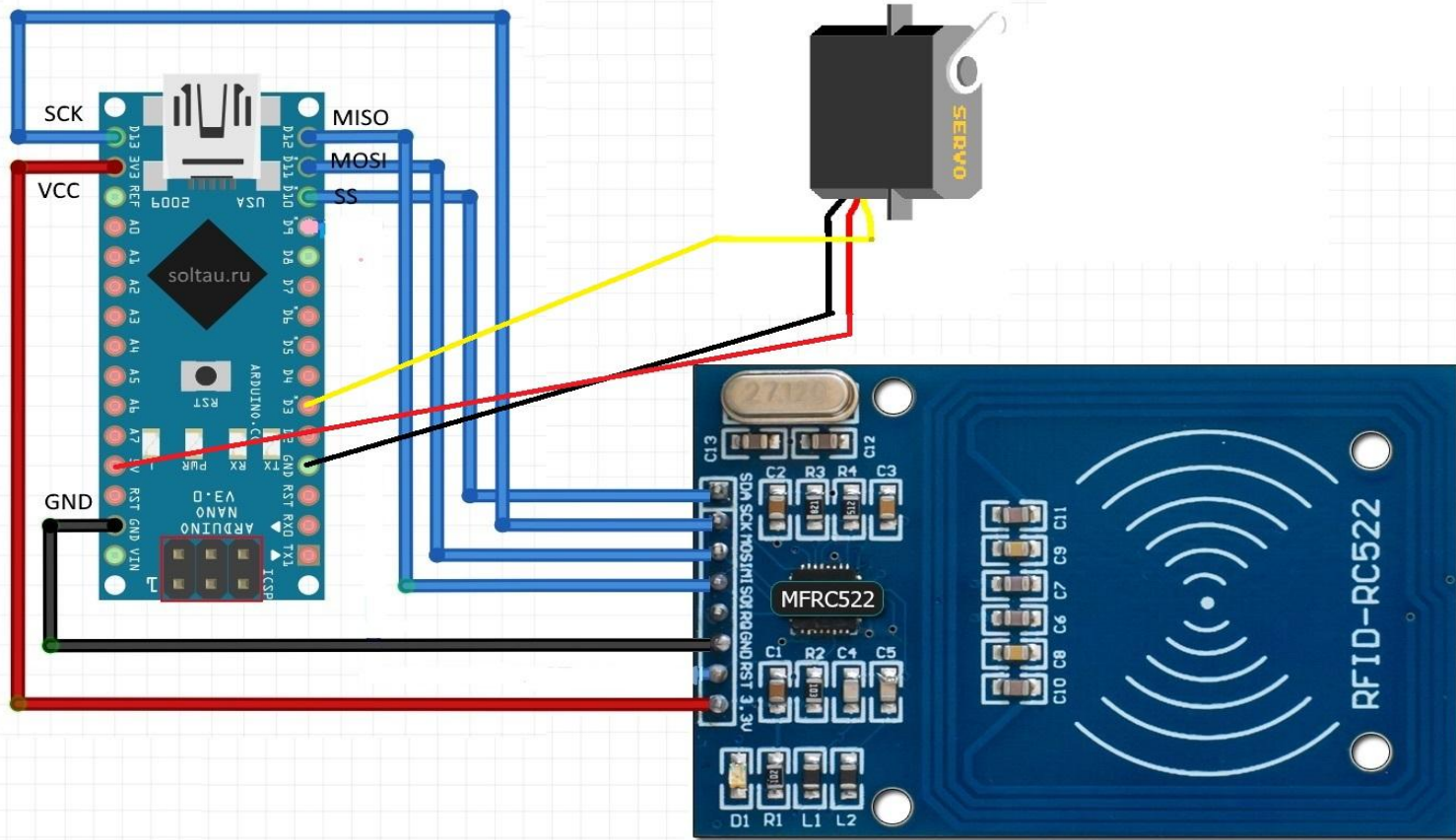
The project has the following workflow:

First we have to set a master tag and then the system goes into normal mode. If we scan an unknown tag the access will be denied, but if we scan the master tag we will enter a program mode from where we can add and authorize the unknown tag. So now if we scan the tag again the access will be granted so we can open the door.

Components Required

- Arduino Nano
- RFID reader and tags
- Servo Motor (SG90)
- Jumper Wires
- Breadboard

Connection Diagram



Connections

1. Connect SDA pin of RFID reader with D10 pin of Arduino Nano.
2. Connect SCK pin of RFID reader with D13 pin of Arduino Nano.
3. Connect MOSI pin of RFID reader with D11 pin of Arduino Nano.
4. Connect MISO pin of RFID reader with D12 pin of Arduino Nano.
5. Connect GND pin of RFID reader with GND pin of Arduino Nano.
6. Connect 3.3V pin of RFID reader with 3.3V pin of Arduino Nano.
7. Connect Red wire of servo with VCC(+5V) of Arduino.
8. Connect Black wire of servo with GND of Arduino.
9. Connect orange wire of servo with D3 pin Arduino.



Code

Automatic_door_lock_using_RFID_Access_module | Arduino 1.8.19

File Edit Sketch Tools Help



Automatic_door_lock_using_RFID_Access_module

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

#define SS_PIN 10
#define RST_PIN 9
#define LED_G 5 //define green LED pin
#define LED_R 4 //define red LED
#define BUZZER 2 //buzzer pin
MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance.
Servo myServo; //define servo name

void setup()
{
  Serial.begin(9600); // Initiate a serial communication
  SPI.begin(); // Initiate SPI bus
  mfrc522.PCD_Init(); // Initiate MFRC522
  myServo.attach(3); //servo pin
  myServo.write(0); //servo start position
  pinMode(LED_G, OUTPUT);
  pinMode(LED_R, OUTPUT);
  pinMode(BUZZER, OUTPUT);
  noTone(BUZZER);
```



Automatic_door_lock_using_RFID_Access_module

```
pinMode(LED_G, OUTPUT);
pinMode(LED_R, OUTPUT);
pinMode(BUZZER, OUTPUT);
noTone(BUZZER);
Serial.println("Put your card to the reader...");
Serial.println();

}

void loop()
{
  // Look for new cards
  if ( ! mfr522.PICC_IsNewCardPresent() )
  {
    return;
  }
  // Select one of the cards
  if ( ! mfr522.PICC_ReadCardSerial() )
  {
    return;
  }
  //Show UID on serial monitor
  Serial.print("UID tag :");
  String content= "";
  byte letter;
```



Automatic_door_lock_using_RFID_Access_module

```
}  
//Show UID on serial monitor  
Serial.print("UID tag :");  
String content= "";  
byte letter;  
for (byte i = 0; i < mfr522.uid.size; i++)  
{  
    Serial.print(mfr522.uid.uidByte[i] < 0x10 ? " 0" : " ");  
    Serial.print(mfr522.uid.uidByte[i], HEX);  
    content.concat(String(mfr522.uid.uidByte[i] < 0x10 ? " 0" : " "));  
    content.concat(String(mfr522.uid.uidByte[i], HEX));  
}  
Serial.println();  
Serial.print("Message : ");  
content.toUpperCase();  
if (content.substring(1) == "07 23 AF 11") //change here the UID of the card/cards that you want to give access  
{  
    Serial.println("Authorized access");  
    Serial.println();  
    delay(500);  
    digitalWrite(LED_G, HIGH);  
    tone(BUZZER, 500);  
    delay(300);  
    tone(BUZZER, 0);  
}
```



Automatic_door_lock_using_RFID_Access_module

```
}  
//Show UID on serial monitor  
Serial.print("UID tag :");  
String content= "";  
byte letter;  
for (byte i = 0; i < mfr522.uid.size; i++)  
{  
    Serial.print(mfr522.uid.uidByte[i] < 0x10 ? " 0" : " ");  
    Serial.print(mfr522.uid.uidByte[i], HEX);  
    content.concat(String(mfr522.uid.uidByte[i] < 0x10 ? " 0" : " "));  
    content.concat(String(mfr522.uid.uidByte[i], HEX));  
}  
Serial.println();  
Serial.print("Message : ");  
content.toUpperCase();  
if (content.substring(1) == "07 23 AF 11") //change here the UID of the card/cards that you want to give access  
{  
    Serial.println("Authorized access");  
    Serial.println();  
    delay(500);  
    digitalWrite(LED_G, HIGH);  
    tone(BUZZER, 500);  
    delay(300);  
}
```

Project Link : <https://youtu.be/vK8rfozAaXw>