**A PROJECT REPORT**

**ON**

**SMART LOCK USING ARDUINO AND RADIO FREQUENCY**

**IDENTIFICATION(RFID) MODULE**



**DEPARTMENT OF ELECTRONICS  
AND COMMUNICATION ENGINEERING**

**BY**

**GVS SAI SUMANTH:1602-16-735-089**

**K. SHRAVANI:1602-16-735-101**

**P. SRILAKSHMI:1602-16-735-106**

**ABSTRACT**

Tired of being locked out when you lose or forget your keys? Well, I have the perfect solution for you! Today we will build an Arduino RFID door lock. I wanted to find an easy and secure way to lock a door, without having to buy an expensive locking device. We will learn about radio frequency identification (RFID) and make use of wireless communication.

Here in this project we are using **Arduino to trigger the Electric Door Lock** and RFID to authenticate, so your RFID tag will act as a key. If you place wrong RFID card near RFID reader a buzzer will beep to alert about wrong card. If you are new to **RFID** first read  its working an interfacing with Arduino.­

The RFID reader consist of a radio frequency module, a control unit and an antenna coil which generates high frequency electromagnetic field. On the other hand, the tag is usually a passive component, which consist of just an antenna and an electronic microchip, so when it gets near the electromagnetic field of the transceiver, due to induction, a voltage is generated in its antenna coil and this voltage serves as power for the microchip.

RFID systems are mostly used in Access management, tracking of goods, Tracking of persons and animals, Toll collection and contactless payment, Machine readable travel documents, Airport baggage tracking logistics etc.

This project serves the purpose of electronics for smart locking system in a more efficient way. This project also involves the usage of modern technology, RFID based system whose future is growing and expanding as more industries and companies invest in the technology.

**INTRODUCTION**

The issue of security is very paramount in any organization. Hence, we intend to aid in security by bringing a smart lock system that involves an individual who walks up to an invisible lock that only he has the key for, and can open it without even touching. The technology behind this project is already used in many industries. RFID technology bridges two technologies in the area of Information and Communication Technologies (ICT), namely Product Code (PC) technology and Wireless technology. This broad-based rapidly expanding technology impacts business, environment and society. The device is powered by rectifying an incoming Radio Frequency signal from the reader. A RFID based access-control system allows only authorized people to enter a particular area of an establishment. Authorized people are provided with unique tags, using which they can access that area.

You have seen **RFID Door Lock** Mechanism in some Hotels and other places, where you don’t need a key to unlock the room. You are given a card and you just need to put it in front of a RFID Reader box, and the lock gets unlocked with a Beep and a Blink of LED. This RFID Door Lock can be made easily at your home and you can install it in any door. These door locks are just electrically operating door lock which gets open when you apply some voltage (typically 12v) to it.

The RFID reader consist of a radio frequency module, a control unit and an antenna coil which generates high frequency electromagnetic field. On the other hand, the tag is usually a passive component, which consist of just an antenna and an electronic microchip, so when it gets near the electromagnetic field of the transceiver, due to induction, a voltage is generated in its antenna coil and this voltage serves as power for the microchip.

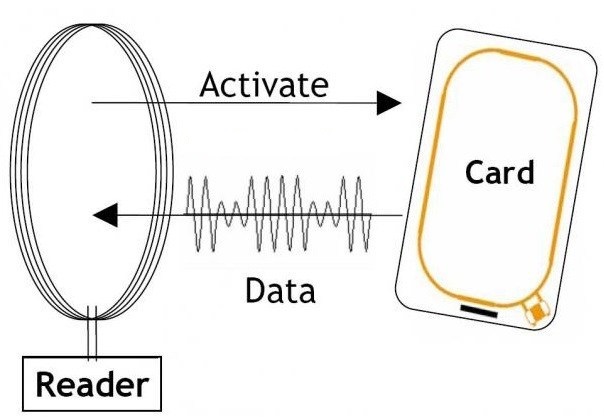
**SMART LOCK USING ARDUINO AND RADIO FREQUENCY IDENTIFICATION(RFID) MODULE**

This RFID based security system is based on micro controller AT89C52 and comprises a RFID module. The card used in our project is a passive RFID device for low-frequency applications (100 kHz-400kHz)



**RFID WORKING:**

An RFID system consists of two main components, a transponder or a tag which is located on the object that we want to be identified, and a transceiver or a reader.

****

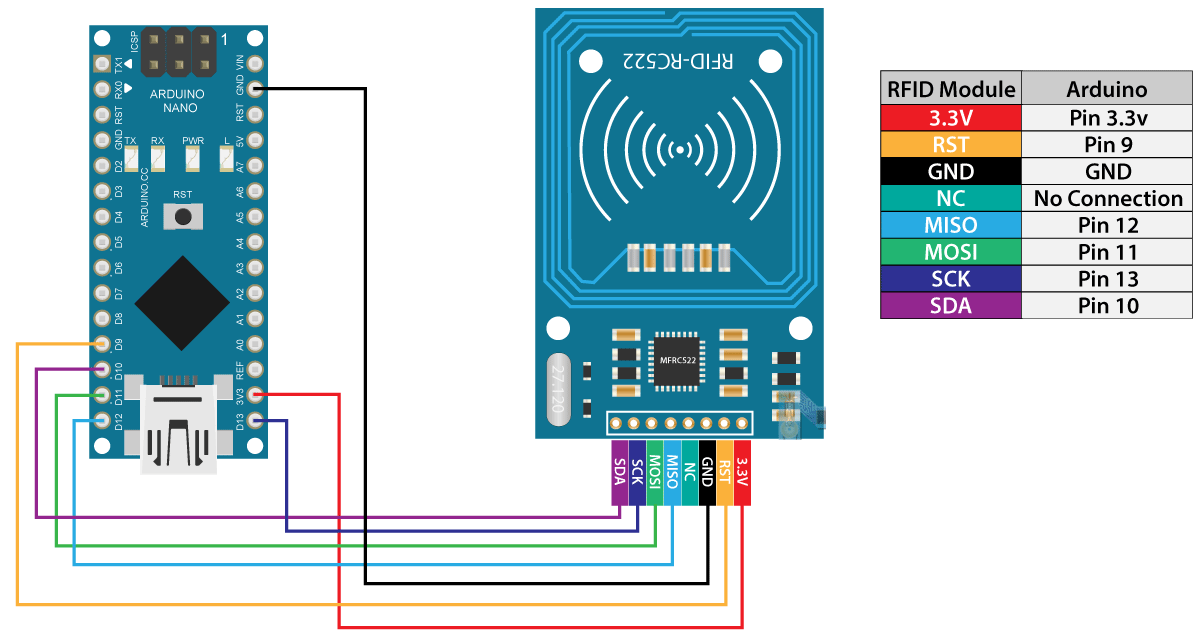
The RFID reader consist of a radio frequency module, a control unit and an antenna coil which generates high frequency electromagnetic field. On the other hand, the tag is usually a passive component, which consist of just an antenna and an electronic microchip, so when it gets near the electromagnetic field of the transceiver, due to induction, a voltage is generated in its antenna coil and this voltage serves as power for the microchip.

**RFID and Arduino**

So that’s the basic working principle and now let’s see how we can use RFID with Arduino and build our own RFID door lock. We will use tags that are based on the MIFARE protocol and the MFRC522 RFID reader

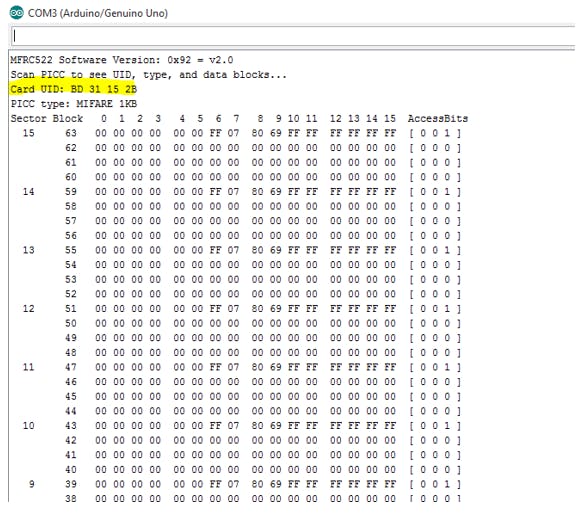
These tags have 1kb of memory and have a microchip that can perform arithmetic operations. Their operating frequency is 13.56 MHz and the operating distance is up to 10 cm depending on the geometry of antenna. If we bring one of these tags in front of a light source, we can notice the antenna and the microchip that we previously talked about.

As for the RFID reader module, it uses the SPI protocol for communication with the Arduino board and here’s how we need to connect them. Please note that we must connect the VCC of the module to 3.3V and as for the other pins we don’t have to worry as they are 5V tolerant.



Once we connect the module, we need to download the MFRC522 library from GitHub. The library comes with several good examples from which we can learn how to use the module.

First, we can upload the “Dump Info” example and test whether our system works properly. Now if we run the Serial Monitor and bring the tag near the module, the reader will start reading the tag and all information from the tag will be displayed on the serial monitor.



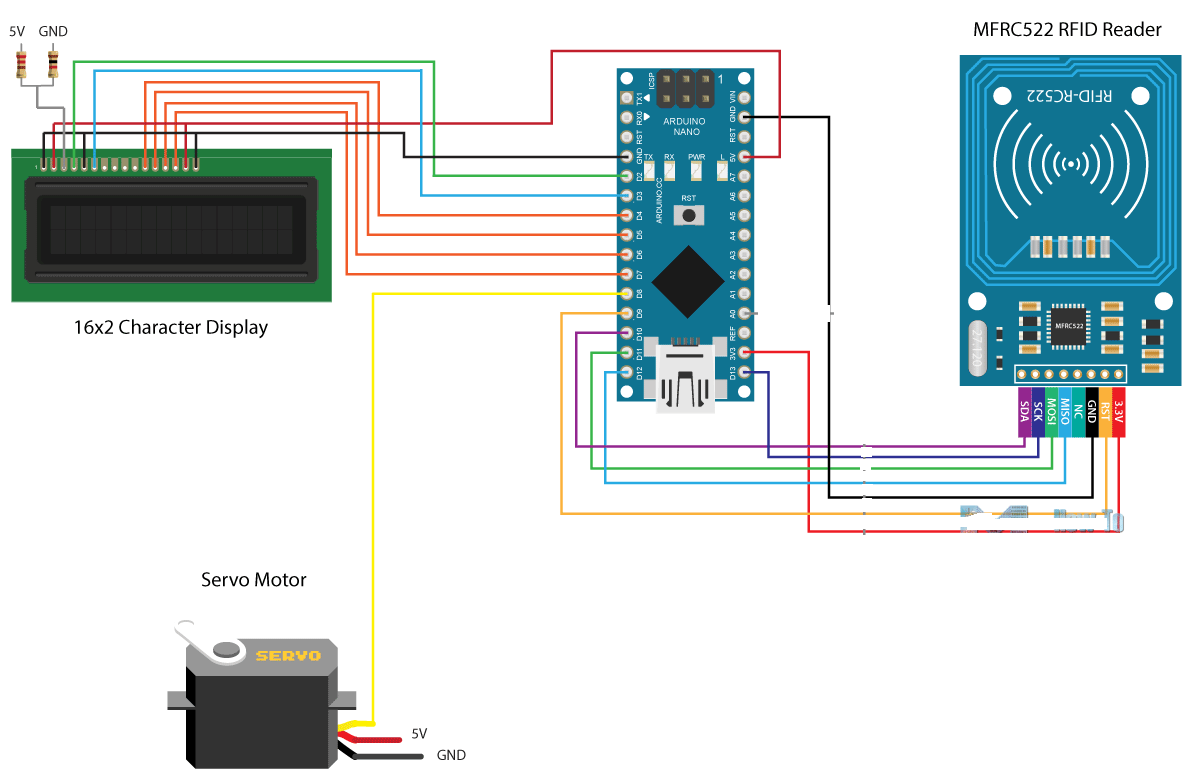
Here we can notice the UID number of the tag as well as the 1 KB of memory which is actually divided into 16 sectors, each sector into 4 blocks and each block can store 2 bytes of data. For this tutorial we won’t use any of the tag’s memory, we will just use the UID number of the tag.

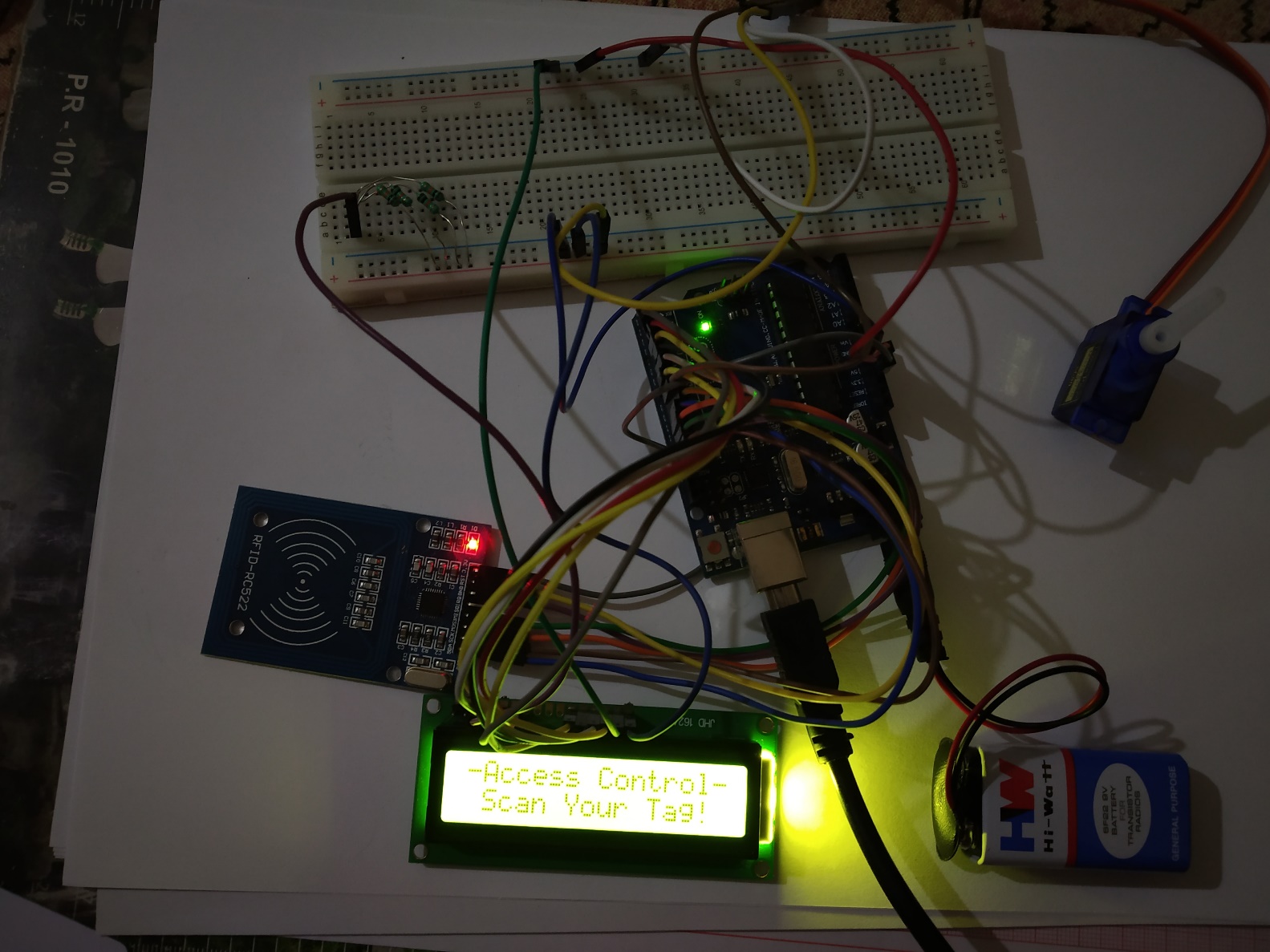
## Arduino RFID Door Lock Access Control Project

Before we go through the code of our RFID door lock project, let’s take a look at the components and the circuit schematics of this project.

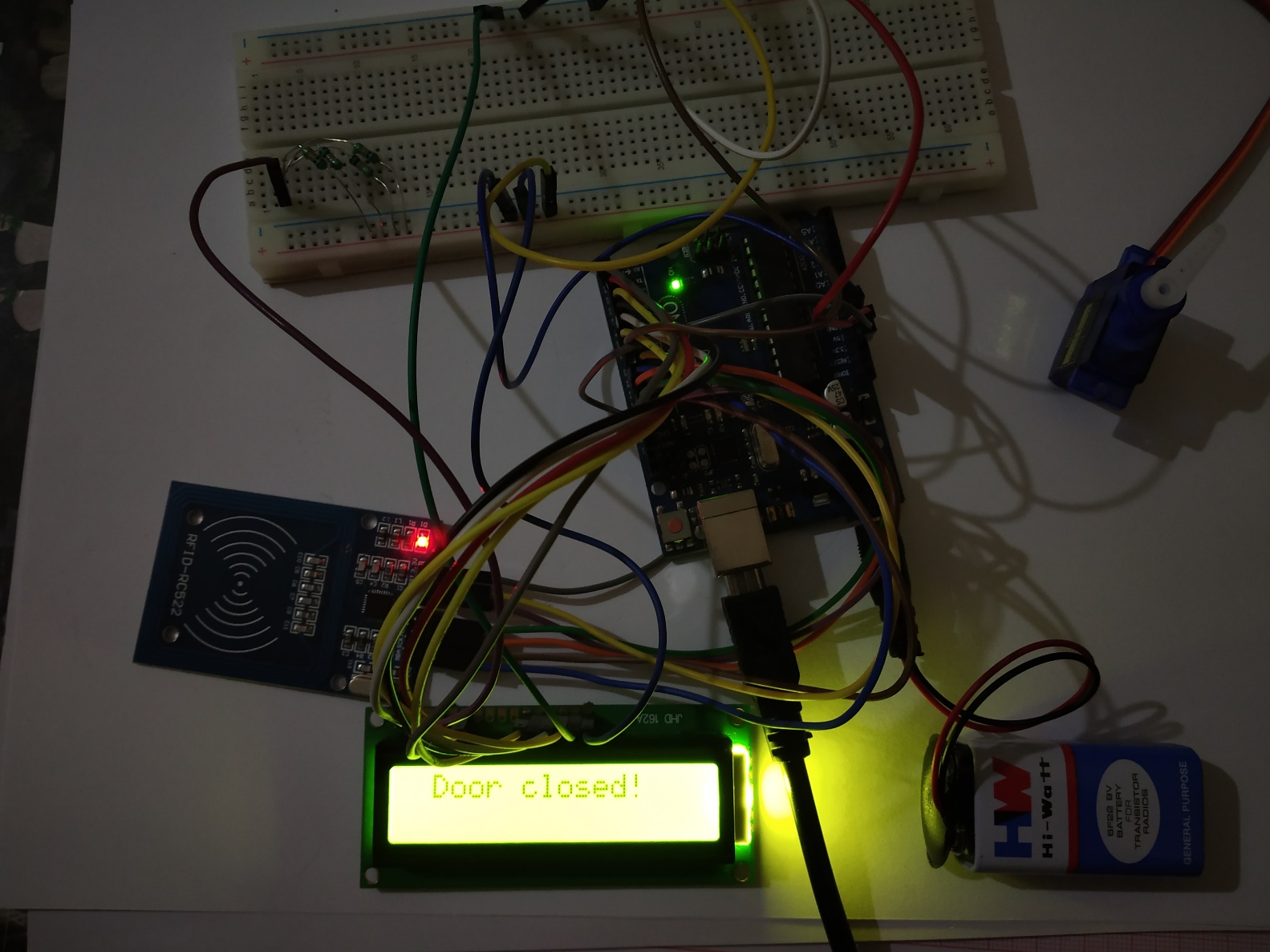
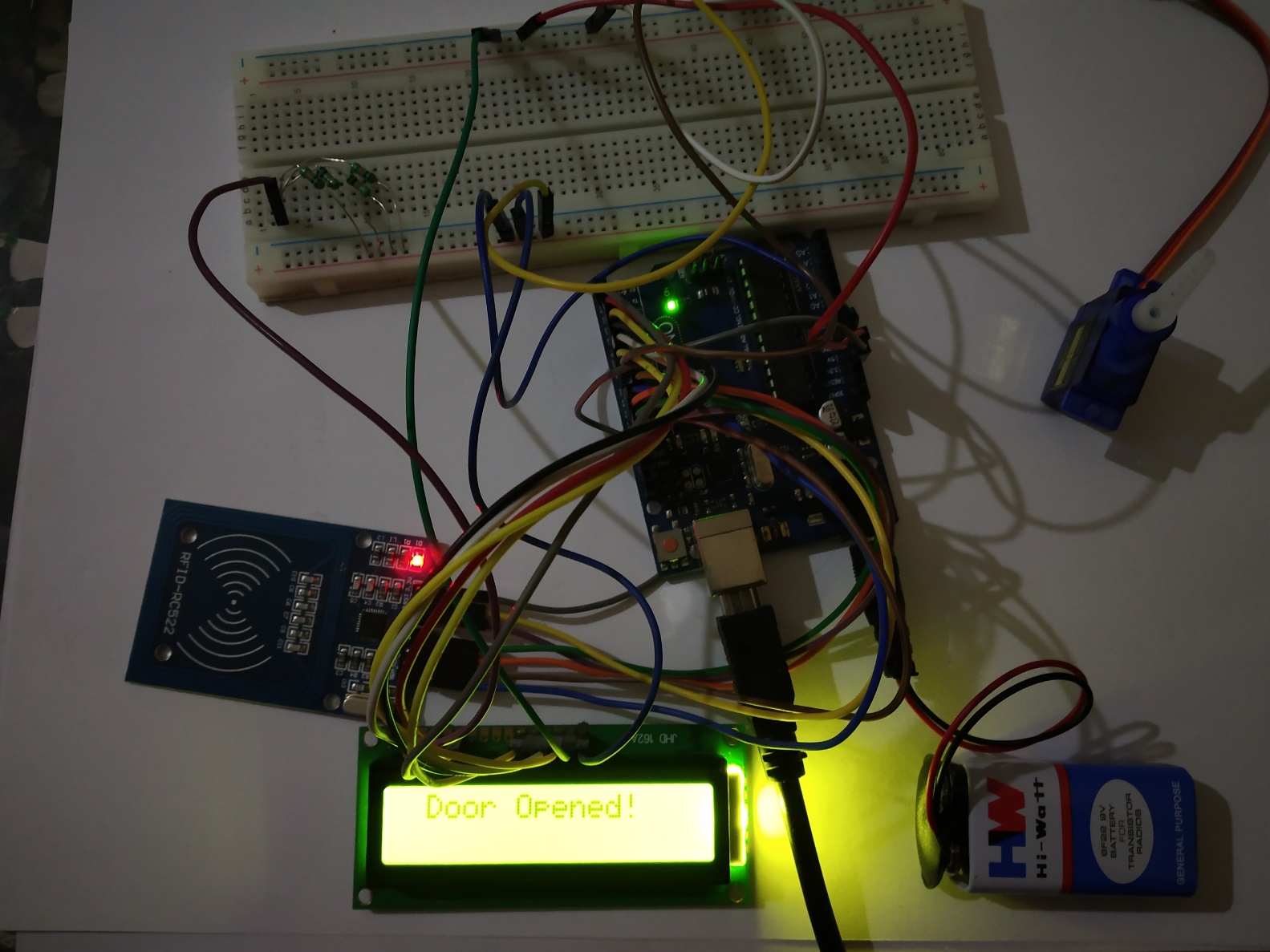
In addition to the RFID module we will use a proximity sensor for checking whether the door is closed or opened, a servo motor for the lock mechanism and a character display.

The components used are:

* MFRC522 RFID Module
* Servo Motor
* LCD Display
* Arduino Board
* Breadboard and Jump Wires
* Proximity Sensor CNY70
* 



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, 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.



The door will automatically lock after we will close the door. If we want to remove a tag from the system, we just have to go again into program mode, scan the know tag and it will be removed.

**Code Parts:**

## So first we need to include the libraries for the RFID module, the display and the servo motor, define some variables needed for the program below as well as create the instances of the libraries.

#include <SPI.h>

#include <MFRC522.h>

#include <LiquidCrystal.h>

#include <Servo.h>

#define RST\_PIN 9

#define SS\_PIN 10

byte readCard[4];

char\* myTags[100] = {};

int tagsCount = 0;

String tagID = "";

boolean successRead = false;

boolean correctTag = false;

int proximitySensor;

boolean doorOpened = false;

// Create instances

MFRC522 mfrc522(SS\_PIN, RST\_PIN);

LiquidCrystal lcd(2, 3, 4, 5, 6, 7); //Parameters: (rs, enable, d4, d5, d6, d7)

Servo myServo; // Servo motor

In the setup section, first we initialize the modules, and set the initial value of the servo motor into a lock position. Then we print the initial message to the display and with the following “while” loop we wait until a master tag is scanned. The getID() custom function gets the tag UID and we put it into the first location of the myTags[0] array.

void setup() {

// Initiating

SPI.begin(); // SPI bus

mfrc522.PCD\_Init(); // MFRC522

lcd.begin(16, 2); // LCD screen

myServo.attach(8); // Servo motor

myServo.write(10); // Initial lock position of the servo motor

// Prints the initial message

lcd.print("-No Master Tag!-");

lcd.setCursor(0, 1);

lcd.print(" SCAN NOW");

// Waits until a master card is scanned

while (!successRead) {

successRead = getID();

if ( successRead == true) {

myTags[tagsCount] = strdup(tagID.c\_str()); // Sets the master tag into position 0 in the array

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Master Tag Set!");

tagsCount++;

}

}

successRead = false;

printNormalModeMessage();

}

Let’s take a look at the getID() custom function. First it checks whether there is a new tag placed near the reader and if so we will continue to the “for” loop which will get the UID of the tag. The tags that we are using have 4 byte UID number so that’s why we need to do 4 iterations with this loop, and using the concat() function we add the 4 bytes into a single String variable. We also set all characters of the string to upper cases and the end we stop the reading.

uint8\_t getID() {

// Getting ready for Reading PICCs

if ( ! mfrc522.PICC\_IsNewCardPresent()) { //If a new PICC placed to RFID reader continue

return 0;

}

if ( ! mfrc522.PICC\_ReadCardSerial()) { //Since a PICC placed get Serial and continue

return 0;

}

tagID = "";

for ( uint8\_t i = 0; i < 4; i++) { // The MIFARE PICCs that we use have 4 byte UID

readCard[i] = mfrc522.uid.uidByte[i];

tagID.concat(String(mfrc522.uid.uidByte[i], HEX)); // Adds the 4 bytes in a single String variable

}

tagID.toUpperCase();

mfrc522.PICC\_HaltA(); // Stop reading

return 1;

}

Before we enter the main loop, at the end of the setup section, we also call the printNormalModeMessage() custom function which prints the “Access Control” message on the display.

void printNormalModeMessage() {

delay(1500);

lcd.clear();

lcd.print("-Access Control-");

lcd.setCursor(0, 1);

lcd.print(" Scan Your Tag!");

}

**SOURCE CODE**

#include <SPI.h>

#include <MFRC522.h>

#include <LiquidCrystal.h>

#include <Servo.h>

#define RST\_PIN 9

#define SS\_PIN 10

byte readCard[4];

char\* myTags[100] = {"E3 1C 81 24"};

int tagsCount = 0;

String tagID = "E3 1C 81 24";

boolean successRead = false;

boolean correctTag = false;

int proximitySensor;

boolean doorOpened = false;

// Create instances

MFRC522 mfrc522(SS\_PIN, RST\_PIN);

LiquidCrystal lcd(2, 3, 4, 5, 6, 7); //Parameters: (rs, enable, d4, d5, d6, d7)

Servo myServo; // Servo motor

void setup() {

// Initiating

SPI.begin(); // SPI bus

mfrc522.PCD\_Init(); // MFRC522

lcd.begin(16, 2); // LCD screen

myServo.attach(8); // Servo motor

myServo.write(10); // Initial lock position of the servo motor

// Prints the initial message

lcd.print("-No Master Tag!-");

lcd.setCursor(0, 1);

lcd.print(" SCAN NOW");

// Waits until a master card is scanned

while (!successRead) {

successRead = getID();

if ( successRead == true) {

myTags[tagsCount] = strdup(tagID.c\_str()); // Sets the master tag into position 0 in the array

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Master Tag Set!");

tagsCount++;

}

}

successRead = false;

printNormalModeMessage();

}

void loop() {

int proximitySensor = analogRead(A0);

// If door is closed...

if (proximitySensor > 200) {

if ( ! mfrc522.PICC\_IsNewCardPresent()) { //If a new PICC placed to RFID reader continue

return;

}

if ( ! mfrc522.PICC\_ReadCardSerial()) { //Since a PICC placed get Serial and continue

return;

}

tagID = "E3 1C 81 24";

// The MIFARE PICCs that we use have 4 byte UID

for ( uint8\_t i = 0; i < 4; i++) { //

readCard[i] = mfrc522.uid.uidByte[i];

tagID.concat(String(mfrc522.uid.uidByte[i], HEX)); // Adds the 4 bytes in a single String variable

}

tagID.toUpperCase();

mfrc522.PICC\_HaltA(); // Stop reading

correctTag = false;

// Checks whether the scanned tag is the master tag

if (tagID == myTags[0]) {

lcd.clear();

lcd.print("Program mode:");

lcd.setCursor(0, 1);

lcd.print("Add/Remove Tag");

while (!successRead) {

successRead = getID();

if ( successRead == true) {

for (int i = 0; i < 100; i++) {

if (tagID == myTags[i]) {

myTags[i] = "E3 1C 81 24";

lcd.clear();

lcd.setCursor(0, 0);

lcd.print(" Tag Removed!");

printNormalModeMessage();

return;

}

}

myTags[tagsCount] = strdup(tagID.c\_str());

lcd.clear();

lcd.setCursor(0, 0);

lcd.print(" Tag Added!");

printNormalModeMessage();

tagsCount++;

return;

}

}

}

successRead = false;

// Checks whether the scanned tag is authorized

for (int i = 0; i < 100; i++) {

if (tagID == myTags[i]) {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print(" Access Granted!");

myServo.write(170); // Unlocks the door

lcd.clear();

lcd.setCursor(0, 0);

lcd.print(" Door Opened!");

delay(10000);

myServo.write(0);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print(" Door closed!");

printNormalModeMessage();

correctTag = true;

}

}

if (correctTag == false) {

lcd.clear();

lcd.setCursor(0, 0);

lcd.print(" Access Denied!");

printNormalModeMessage();

}

}

uint8\_t getID() {

// Getting ready for Reading PICCs

if ( ! mfrc522.PICC\_IsNewCardPresent()) { //If a new PICC placed to RFID reader continue

return 0;

}

if ( ! mfrc522.PICC\_ReadCardSerial()) { //Since a PICC placed get Serial and continue

return 0;

}

tagID = "E3 1C 81 24";

for ( uint8\_t i = 0; i < 4; i++) { // The MIFARE PICCs that we use have 4 byte UID

readCard[i] = mfrc522.uid.uidByte[i];

tagID.concat(String(mfrc522.uid.uidByte[i], HEX)); // Adds the 4 bytes in a single String variable

}

tagID.toUpperCase();

mfrc522.PICC\_HaltA(); // Stop reading

return 1;

}

void printNormalModeMessage() {

delay(1500);

lcd.clear();

lcd.print("-Access Control-");

lcd.setCursor(0, 1);

lcd.print(" Scan Your Tag!");}

**CONCLUSION:**

This Project shows how we can make our own security device for more security and we know the real time application of RFID tag the revolution of coming generation where every home can be seen with one such tags, not only house doors but even automobile industries can use this as it is reliable and more secure.

**REFERENCES**

Mechatronics:

<https://howtomechatronics.com/tutorials/arduino/rfid-works-make-arduino-based-rfid-door-lock/>

Circuit digest:

<https://circuitdigest.com/microcontroller-projects/arduino-rfid-door-lock-code>