CHITKARA UNIVERSITY INSTITUTE OF ENGINEERING & TECHNOLOGY

COMPUTER SCIENCE ENGINEERING

Internet of Things

Course Code – CS201

Submitted By:

Group-7 :
Lovish(0076) , Neerav(0103),
Paras(0106) ,Aditya(0111) ,Anmol(0090).

Class : CSE-1st Sem.
Session : 2019-20

Submitted To :

Dr. Dhawan Singh

CHITKARA UNIVERSITY
NH-7, CHANDIGARH PATIALA NATIONAL HIGHWAY
RAJPURA, PUNJAB 140401.

Aim of the Experiment: "SMART DOOR LOCK SYSTEM" using Arduino and IoT Cloud.

Components Used: Arduino ESP8266 NodeMCU, Servo Motor SG90, Breadboard ,Jumper wires , MFRC522 RFID Reader, RFID Tags , USB cable.

INTRODUCTION :The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. An IoT system consists of sensors/devices which "talk" to the cloud through some kind of connectivity. Once the data gets to the cloud, software processes it and then might decide to perform an action, such as sending an alert or automatically adjusting the sensors/devices without the need for the user. The Internet of things (IoT) is the inter-networking of physical devices, vehicles , smart-devices, buildings, and other items embedded with electronics, software, sensors, actuators, and network connectivity which enable these objects to collect and exchange data. Internet of things is the communication of anything with any other things. The door plays an important role in home security. So, providing a secured door system for houses has become a vital research.Home security implicitly means a secured mechanism for the door. So the idea of Smart door lock system has been proposed. This is one of the most popular home security systems. With this system, only the authorized individuals can gain the permission to access the doors with the help of a card .The door will only open when one scans that card on RFID scanner .The lock is connected to the cloud and all the data is sent and stored there ,thus one can monitor his/her house from anywhere.

WORKING: In this project, we have defined a default Card which has a code in in it. The code of the card can easily be changed via coding. The lock is connected to cloud via coding and the default value is set as '0'. When we scan any card ,the code of the card will then be matched with the one stored in the Arduino. If the card is correct, then it will show 'Access Granted' on serial monitor and then rotate servo motor from 0^0 to 180^0 and will update the output as '1' on the cloud .Servo motor is connected to the lock and then it will give 3 seconds time interval for entering via door. Once 3 second is completed, the door will get locked automatically and the output will again get updated as '0'. If the card is incorrect, then it will show 'Access Denied' on the serial monitor and the value on the cloud will get updated to '10' and it will show a trigger as 'WRONG CARD!!!ENTRY NOT ALLOWED'.

Circuit Diagram:

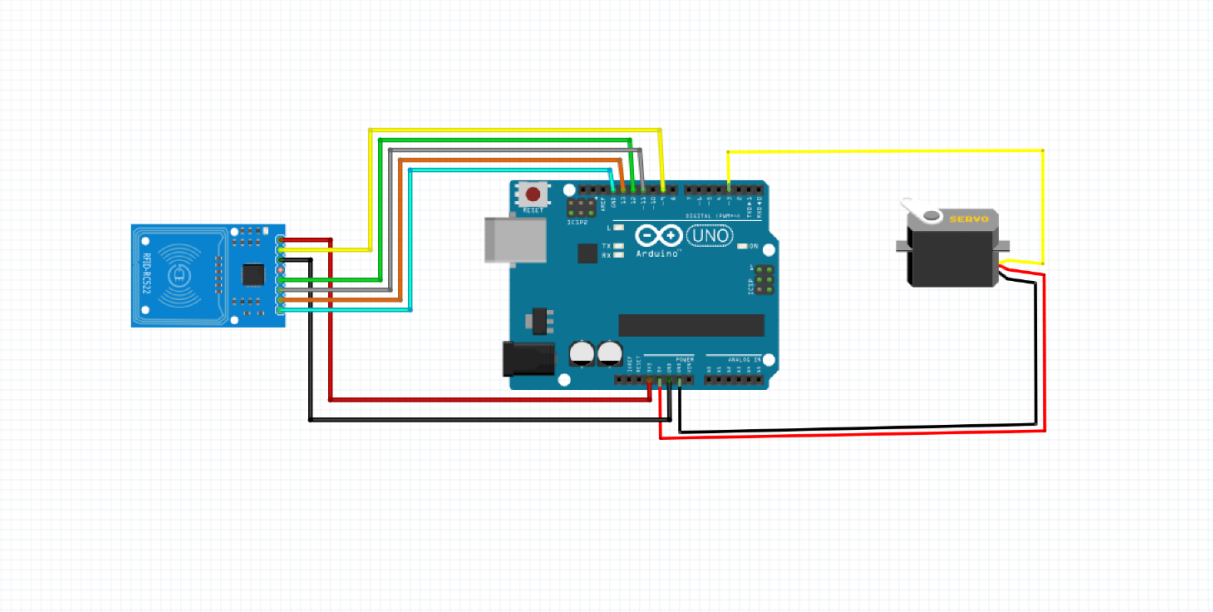

Fig 11.1 Circuit Diagram

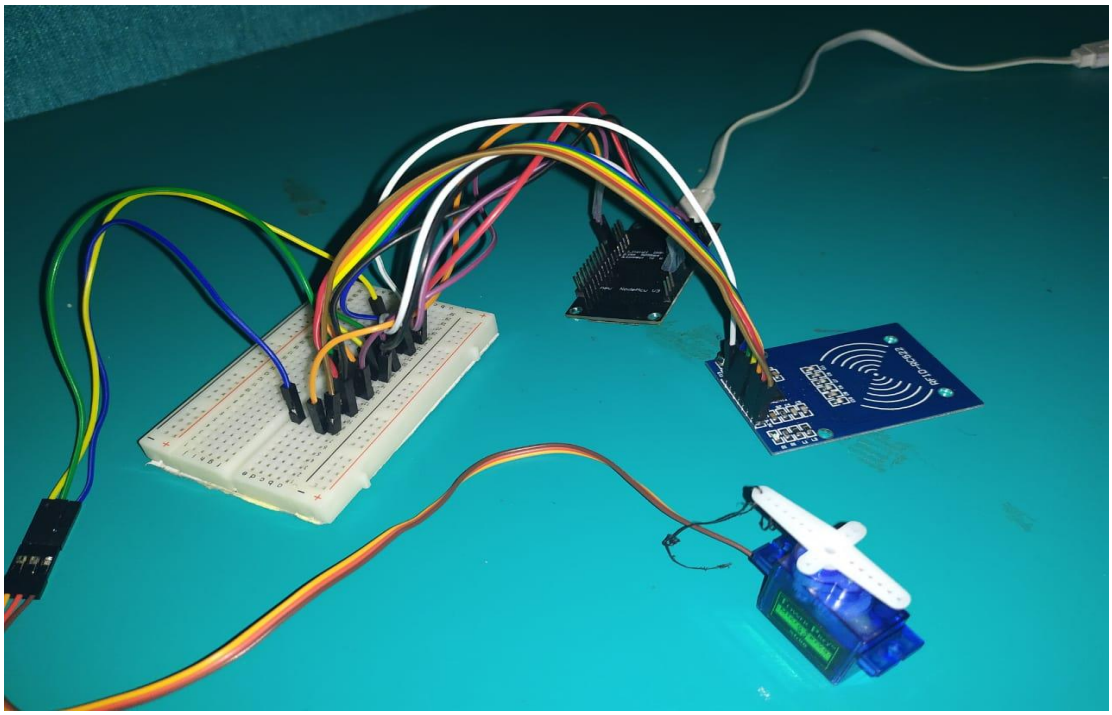

Fig 11.2 Real time image of the circuit

Software Program:

```
#include <ESP8266WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"
#include <SPI.h>
#include <MFRC522.h>
#include <Servo.h>
#define SS_PIN D4
#define RST_PIN D3
#define SERVO_PIN D4

Servo myservo;
#define ACCESS_DELAY 2000
#define DENIED_DELAY 1000

MFRC522 mfrc522(SS_PIN, RST_PIN);
#define WLAN_SSID       "MBH"
#define WLAN_PASS       "CUPunjab"
#define AIO_SERVER      "io.adafruit.com"
#define AIO_SERVERPORT  1883                   // use 8883 for SSL
#define AIO_USERNAME    "lovish2804"
#define AIO_KEY         "388c8fa1314749fea4dcf702af8c6daa"

WiFiClient client;
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT,
AIO_USERNAME, AIO_KEY);
Adafruit_MQTT_Publish photocell = Adafruit_MQTT_Publish(&mqtt,
AIO_USERNAME "/feeds/Door Lock");

void MQTT_connect();

void setup() {
  Serial.begin(115200);
  delay(10);


  Serial.println(F("Adafruit MQTT demo"));

  // Connect to WiFi access point.
```

```
  Serial.println(); Serial.println();
  Serial.print("Connecting to ");
  Serial.println(WLAN_SSID);

  WiFi.begin(WLAN_SSID, WLAN_PASS);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();

  Serial.println("WiFi connected");
  Serial.println("IP address: "); Serial.println(WiFi.localIP());
   SPI.begin();
  mfrc522.PCD_Init();
   myservo.attach(SERVO_PIN);
  myservo.write( 60 );
  delay(3000);
  myservo.write( 0 );
  Serial.println("Put your card to the reader...");
  Serial.println();
}
uint32_t x=0;

void loop() {
MQTT_connect();
  int k=0;
if ( ! mfrc522.PICC_IsNewCardPresent())
  {
    return;
  }
  // Select one of the cards
  if ( ! mfrc522.PICC_ReadCardSerial())
  {
    return;
  }
  Serial.print("UID tag :");
  String content= "";
  byte letter;
  for (byte i = 0; i < mfrc522.uid.size; i++)
```

```
 {
    Serial.print(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " ");
    Serial.print(mfrc522.uid.uidByte[i], HEX);
    content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
    content.concat(String(mfrc522.uid.uidByte[i], HEX));
 }
 Serial.println();
 Serial.print("Message : ");
 content.toUpperCase();

 if (content.substring(1) == "81 E5 D5 2F")
 {
   Serial.println("Authorized access");
  k=1;
   Serial.println();
  myservo.write( 60 );

  if (! photocell.publish(k)) {
   Serial.println(F("Failed"));
   k=0;
 } else {
   Serial.println(F("OK!"));
   k=0;
 }
  delay(3000);
 myservo.write( 0 );
}

else   {
 k=10;
  if (! photocell.publish(k)) {
   Serial.println(F("Failed"));
 } else {
   Serial.println(F("OK!"));
 }
   Serial.println(" Access denied");
k=0;
}
 delay(DENIED_DELAY);
 Serial.print(F("\nSending photocell val "));
```

```
  Serial.print(x);
  Serial.print("...");

  if (! photocell.publish(k)) {
    Serial.println(F("Failed"));
  } else {
    Serial.println(F("OK!"));
  }
  delay (1000);
}
void MQTT_connect() {
  int8_t ret;

  // Stop if already connected.
  if (mqtt.connected()) {
    return;
  }

  Serial.print("Connecting to MQTT... ");

  uint8_t retries = 3;
  while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected
       Serial.println(mqtt.connectErrorString(ret));
       Serial.println("Retrying MQTT connection in 5 seconds...");
       mqtt.disconnect();
       delay(5000);  // wait 5 seconds
       retries--;
       if (retries == 0) {
         while (1);
       }
  }
  Serial.println("MQTT Connected!");
}
```

Bill of Material: The pricing of each component used in the project is listed below:

- NodeMCU ₹360
- Breadboard ₹50
- Battery (2) ₹40
- Snapper (2) ₹20
- Servo Motor ₹90
- Jumper Wires ₹60
- RFID scanner and tags ₹190

TOTAL = ₹ 810