**Hướng dẫn sử dụng Module RFID Reader with Cards Kit 13.56MHz**

**Ứng dụng :** dùng để đọc và ghi dữ liệu cho thẻ NFC ở tần số 13.56Mhz,với thiết kế nhỏ gọn giá thành rẻ module thích hợp cho nhiều ứng dụng như: kiểm soát xe cộ,hành khách bằng vé thẻ hay thay thế công nghệ mã vạch giúp lưu nhiều dữ liệu hơn hay trong quản lý nhân viên và chấm công, .v.v.

**Thông số:**

- Dòng làm việc : 13 – 26mA/3.3V-DC

- Dòng tĩnh : 10 – 13mA/3.3V-DC

- Dòng ở chế độ nghỉ : <80uA

- Dòng làm việc max: 30mA

- Tần số hoạt động : 13.56Mhz

- Các loại thẻ hỗ trợ : mifare\_1 S50, mifare\_1 S70, mifare UltraLight, mifare Pro, mifare Desfire.

- Giao tiếp : SPI

- Khoảng cách đọc : <60mm/1.95”(mifare\_1)

- Kích thước : 40mm\*60mm

- Tốc độ truyền dữ liệu : max 10Mbit/s

**Tính năng,đặc điểm:**

- MF RC522 ứng dụng cho việc tích hợp cao việc đọc và viết dữ liệu.

- Giao tiếp với thẻ tại tần số 13.56Mhz.

- Là sự lựa chọn tốt cho sự phát triển của các thiết bị thông minh và thiết bị di động cầm tay.

- MF RC552 sử dụng cho việc nâng cao điều chế và giải mã điều chế thông tin giao tiếp thụ động bằng các phương pháp hoàn toàn thích hợp trong tần số 13.56Mhz .

- Tương thích với bộ phát tín hiệu 14443A.

- ISO 14443A xử lý kỹ thuật để phát hiện lỗi và các khung hình.

- CRYPTO1 nhanh chóng hỗ trợ mã hóa thuật toán để xác nhận sản phẩm là mafire.

- MF RC552 hỗ trợ mafire giao tiếp với các chuỗi bằng tốc độ cao,tốc độ truyền dữ liệu 2 chiều lên tới 424kbit/s.

- MF RC552 cũng tương tự như MF RC500,MF RC530 nhưng cũng có những đặc điểm và sự khác biệt,giao tiếp giữa nó và máy chủ ở chế độ SPI giúp giảm thiểu các kết nối hạn hẹp của PCB,giảm chi phí đáng kể.

- Các MF 552 là các module được thiết kế để dể dàng sử dụng với các đầu đọc thẻ mạch.

- Giá thành rẻ và được áp dụng cho sự phát triển các thiết bị cho người sử dụng

- Nâng cao sự phát triển của các ứng dụng ,đáp ứng nhu cầu về sử dụng các thiết bị đầu/cuối sử dụng thẻ nhớ RF.

- Module này có thể được nạp trược tiếp vào các khuôn reader khác nhau,rất thuận tiện.

**Chân kết nối:**

- SDA(SS) chân lựa chọn chip khi giao tiếp SPI (kích hoạt mức thấp).

- SCK :chân xung trong chế độ SPI.

- MOSI(SDI): Master Data Out – Slave In trong chế độ giao tiếp SPI.

- MISO(SDO): Master Data In – Slave Out trong chế độ giao tiếp SPI.

- IRQ : chân ngắt.

- GND : chân nối mass.

- RST : chân reset lại module.

- VCC : nguồn 3.3V.

**Ứng dụng ví dụ: Đọc mã ID của thẻ qua hệ thống đọc thẻ RC552 kết nối Arduino.**

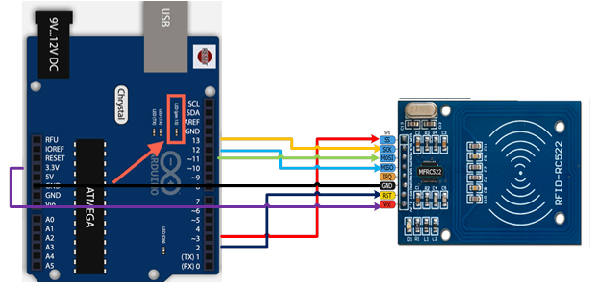
Chuẩn bị: + module Arduino Uno

                + module đọc thẻ RC552

                + led, dây nối, bảng mạch.

   Kết nối pin như sơ đồ sau:

|  |  |
| --- | --- |
| RFID | ARDUINO UNO |
| SDA | D3 |
| SCK | D13 |
| MOSI | D11 |
| MISO | D12 |
| IRQ |  |
| GND | GND |
| RST( RESET) | D2 |
| VCC | 3V3 |



**Code:**

#include <SPI.h>

#include <MFRC522.h>

#include<LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

#define SS\_PIN 3

#define RST\_PIN 2

MFRC522 rfid(SS\_PIN, RST\_PIN); // Instance of the class

MFRC522::MIFARE\_Key key;

// khai bao mang chua gia tri ID

byte nuidPICC[4];

void setup() {

  lcd.begin(16,2);

  Serial.begin(9600);

  SPI.begin(); // Init SPI bus

  rfid.PCD\_Init(); // Init MFRC522

}

void loop() {

    lcd.setCursor(0,0);

    lcd.print("Khong co the    ");

    lcd.setCursor(0,1);

    lcd.print("                ");

  // tim kiem the moi

  if ( ! rfid.PICC\_IsNewCardPresent())

    return;

  // Xac nhan  neu NUID da duoc doc

  if ( ! rfid.PICC\_ReadCardSerial())

    return;

    Serial.println(F("A new card has been detected."));

    // Doc gia tri ID va ghi ra

    for (byte i = 0; i < 4; i++) {

      nuidPICC[i] = rfid.uid.uidByte[i];

    }

    // so sanh ID the tung nguoi

if(nuidPICC[0] == 99 && nuidPICC[1]== 7 && nuidPICC[2]== 147 && nuidPICC[3]==  244){

    lcd.setCursor(0,0);

    lcd.print("LE TUAN ANH   ");

    lcd.setCursor(0,1);

    lcd.print("ID: ");

    for(int i = 0; i <4 ; i++){

      disPlay(nuidPICC[i], i);

      }

    }

    if(nuidPICC[0] == 101 && nuidPICC[1]== 82 && nuidPICC[2]== 17 && nuidPICC[3]==  2){

    lcd.setCursor(0,0);

    lcd.print("BUI NGOC NHA   ");

    lcd.setCursor(0,1);

    lcd.print("ID: ");

    for(int i = 0; i <4 ; i++){

      disPlay(nuidPICC[i], i);

      }

    }

    if(nuidPICC[0] == 229 && nuidPICC[1]== 15 && nuidPICC[2]== 14 && nuidPICC[3]==  2){

    lcd.setCursor(0,0);

    lcd.print("PHAM VAN LAP   ");

    lcd.setCursor(0,1);

    lcd.print("ID: ");

    for(int i = 0; i <4 ; i++){

      disPlay(nuidPICC[i], i);

      }

    }

      delay(5000);

    lcd.setCursor(0,0);

    lcd.print("               ");

    lcd.setCursor(0,1);

    lcd.print("               ");

}

// hien thi ID

void disPlay(byte number,int i){

      byte bitHigh = number/16;

      byte bitLow = number%16;

      lcd.setCursor((4 +3\*i),1);

      convert(bitHigh);

      lcd.setCursor((5 +3\*i),1);

      convert(bitLow); }

//  chuyen sang he 16

void convert(byte number){

  if(number < 10) lcd.print(number);

    if(number == 10) lcd.print("A");

      if(number == 11) lcd.print("B");

        if(number == 12) lcd.print("C");

          if(number == 13) lcd.print("D");

            if(number == 14) lcd.print("E");

              if(number == 15) lcd.print("F");

}