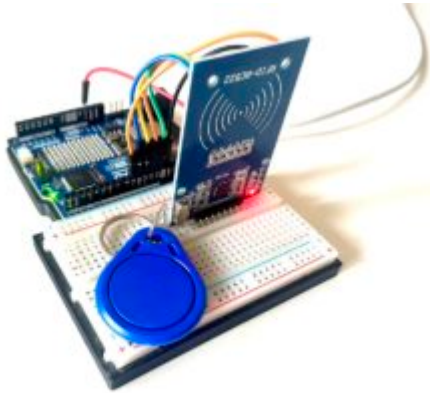


Arduino RFID: Der Einstieg mit dem RC522-Reader



In diesem Tutorial tauchen wir in eine faszinierende Technologie ein, die uns im Alltag ständig begegnet: RFID. Ob beim kontaktlosen Bezahlen, bei der Zeiterfassung oder in modernen Schließsystemen – die kleinen Chips sind überall. Aber hast du dich schon mal gefragt, wie das eigentlich funktioniert? Und noch besser: Wie du es selbst für deine eigenen Projekte nutzen kannst?

Du wirst einen RFID-Reader an einen Arduino anschließen und die einzigartige ID von RFID-Tags und -Karten auslesen. Also, schnapp dir deinen Arduino und lass uns loslegen!

Was ist RFID eigentlich?

RFID steht für „Radio-Frequency Identification“, also die Identifizierung mithilfe von Radiowellen. Ein RFID-System besteht immer aus zwei Teilen:

1. **Einem Transponder (der Tag oder die Karte):** Dieser enthält einen winzigen Chip mit einer Antenne. Auf dem Chip ist eine eindeutige Kennung (UID – Unique Identifier) gespeichert.
2. **Einem Lesegerät (der Reader):** Dieses erzeugt ein

elektromagnetisches Feld. Kommt ein Transponder in die Nähe dieses Feldes, wird er mit Energie versorgt und sendet seine Informationen zurück an das Lesegerät.

Das Schöne daran ist, dass dies alles kontaktlos und ohne Sichtverbindung funktioniert.

Für dieses Arduino RFID Tutorial brauchst du

- Ein Arduino-Board (im Folgenden ein Arduino Uno)
- Ein RFID-RC522 Modul (meistens kommen hier schon eine Karte und ein Schlüsselanhänger-Tag mit)
- Ein Breadboard & Kabel

Schritt 1: Die Verkabelung

Das Herzstück ist das RC522-Modul. Es kommuniziert mit dem Arduino über eine Schnittstelle namens SPI (Serial Peripheral Interface).

Wichtiger Hinweis: Das RC522-Modul arbeitet mit 3.3V! Schließe es auf keinen Fall an den 5V-Pin deines Arduinos an, da es sonst beschädigt werden könnte.

Verbinde die Pins des RC522-Moduls wie folgt mit deinem Arduino Uno:

RC522 Pin	Arduino Uno Pin	Beschreibung
3.3V	3.3V	Spannungsversorgung
RST	Pin 9	Reset-Pin

GND	GND	Masse
MISO	Pin 12	Master In, Slave Out (SPI)
MOSI	Pin 11	Master Out, Slave In (SPI)
SCK	Pin 13	Serial Clock (SPI)
SDA	Pin 10	Slave Select (SPI)
IRQ	–	(wird nicht verbunden)

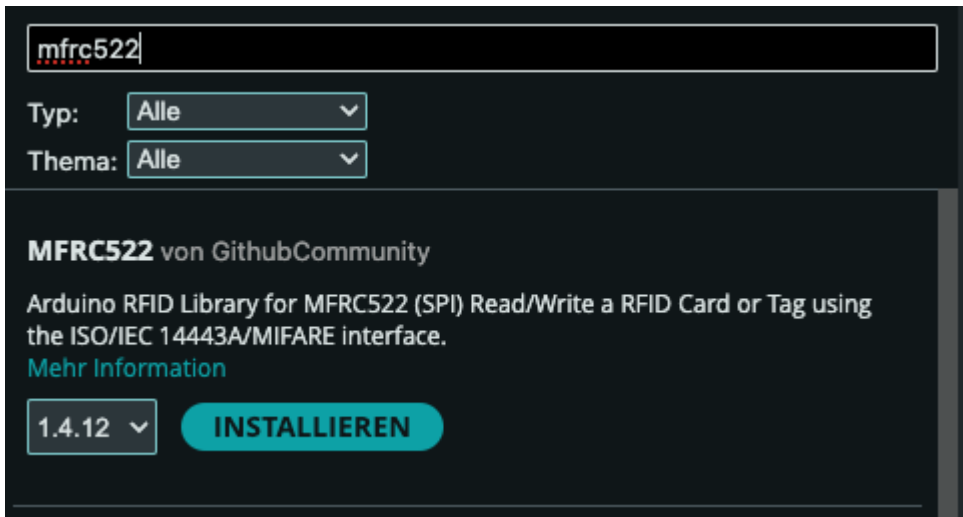
Nimm dir für die Verkabelung einen Moment Zeit und prüfe alles doppelt. Eine saubere Verkabelung ist die halbe Miete!

Schritt 2: Die Bibliothek installieren

Damit dein Arduino versteht, wie er mit dem RFID-Modul sprechen soll, benötigst du eine passende Bibliothek. Für das RC522-Modul gibt es eine sehr beliebte und gut gepflegte Bibliothek von „GitHubCommunity“.

So installierst du sie ganz einfach über die Arduino IDE:

1. Öffne deine Arduino IDE.
2. Gehe zum Menüpunkt **Werkzeuge > Bibliotheken verwalten...**
3. Gib in das Suchfeld MFRC522 ein.
4. Suche nach der Bibliothek „MFRC522 by GithubCommunity“ und klicke auf **Installieren**.



Das war's schon! Jetzt kennt dein Arduino die Befehle, um mit dem RFID-Modul zu kommunizieren.

Schritt 3: Einen Tag auslesen

Jetzt wird es spannend! Hier kommt der Sketch, um die einzigartige ID eines Tags auszulesen. Diese UID ist quasi der Personalausweis des Chips, den du in deinen Projekten verwenden kannst.

Öffne die Arduino IDE und füge den folgenden Code ein:

```
// Arduino RFID
//polluxlabs.net

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

#define RST_PIN 9           // Reset-Pin, den wir an Pin 9
                           // angeschlossen haben
#define SS_PIN 10          // Slave-Select-Pin, den wir an Pin
                           // 10 angeschlossen haben

// Erstellen einer Instanz des MFRC522-Objekts
MFRC522 mfrc522(SS_PIN, RST_PIN);

void setup() {
```

```

    Serial.begin(9600);    // Starten der seriellen Kommunikation
für die Ausgabe am PC
    while (!Serial);      // Warten, bis der serielle Monitor
bereit ist
    SPI.begin();          // Initialisieren der SPI-
Kommunikation
    mfrc522.PCD_Init();   // Initialisieren des RFID-Moduls
    Serial.println("Halte einen RFID-Tag an das Lesegerät...");
}

void loop() {
    // Suche nach neuen Karten
    if ( ! mfrc522.PICC_IsNewCardPresent() ) {
        return;
    }

    // Wähle eine der Karten aus
    if ( ! mfrc522.PICC_ReadCardSerial() ) {
        return;
    }

    // Zeige die UID im Seriellen Monitor an
    Serial.print("UID des Tags: ");
    String content= "";
    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.println("-----");
    // Halte die Kommunikation an, damit derselbe Tag nicht
ständig neu gelesen wird
    mfrc522.PICC_HaltA();
}

```

Was macht dieser Code?

- **#include:** Wir binden die benötigten Bibliotheken für SPI und das RFID-Modul ein.
- **#define:** Wir legen fest, welche Arduino-Pins wir für RST und SS verwenden.
- **setup():** Hier wird die serielle Kommunikation gestartet und das RFID-Modul initialisiert.
- **loop():** Die Hauptschleife wartet darauf, dass ein neuer Tag (PICC_IsNewCardPresent()) in Reichweite kommt. Wenn einer erkannt wird, wird seine UID ausgelesen (PICC_ReadCardSerial()) und anschließend in einer lesbaren Form (Hexadezimal) auf dem Seriellen Monitor ausgegeben.

Lade den Sketch auf deinen Arduino hoch und halte nun eine RFID-Karte oder den Schlüsselanhänger an dein Modul. Voilà! Du solltest die UID des Tags auf dem Monitor sehen.

Wie geht es weiter?

Glückwunsch! Du hast soeben deinen eigenen RFID-Reader in Betrieb genommen und gelernt, wie man die ID eines Tags ausliest. Allein damit sind die Möglichkeiten schon riesig:

- **Baue ein smartes Türschloss:** Speichere die UID deines Tags im Code. Wird diese UID erkannt, steuert der Arduino ein Servo an, um ein Schloss zu öffnen.
- **Erstelle eine Musikbox:** Weise verschiedenen UIDs im Code unterschiedliche Lieder zu. Je nachdem, welchen Tag du an den Reader hältst, wird eine andere Melodie abgespielt.
- **Ein personalisiertes Begrüßungssystem:** Der Arduino erkennt deine UID und begrüßt dich mit Namen auf einem LCD-Display.

Experimentiere mit dem Code und entwickle deine eigenen Ideen.