

# RFID-Reader

## Benötigtes Material

- Arduino Uno
- USB-Kabel für den Arduino
- RFID-Reader (MFRC522)
- RFID-TAG
- LED
- Breadboard
- Batterie
- 9x MM-Kabel

## Benötigte Software

- Bibliothek MFRC522 von GithubCommunity

## Beschreibung

Der RFID („radio-frequency identification“) Reader wird verwendet, um von RFID Sendern (auch „RFID Tags“ genannt) per Funk einen bestimmten Code auszulesen. Jeder Sender hat dabei eine einmalige ID (UID). Somit können wir den Arduino verwenden, um Schließanlagen oder ähnliche Projekte zu realisieren, bei denen man sich identifizieren soll.

RFID-TAGs können verschiedene Formen haben, wie z.B. Schlüsselanhänger oder Karten im Kreditkartenformat.

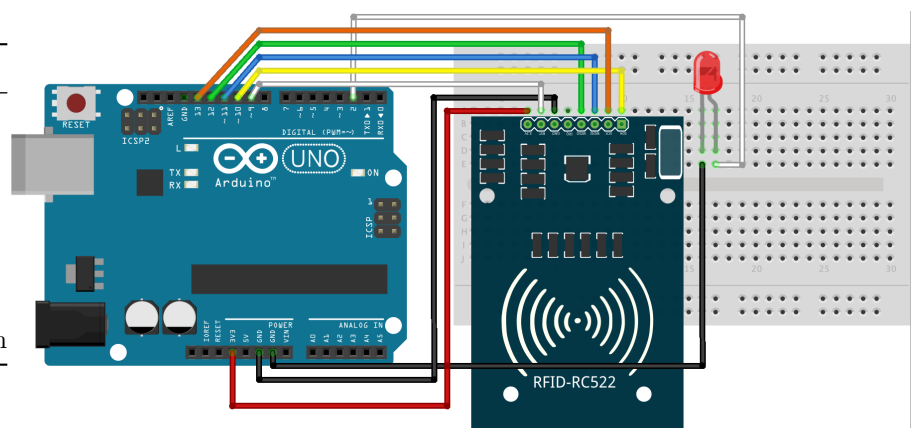
Auf dem folgenden Bild sieht man links und rechts zwei RFID-TAGs und in der Mitte den RFID-Empfänger RFID-RC522 mit noch nicht angelöteter Stiftleiste. Es gibt auch Versionen, bei denen die Stiftleiste bereits an den RFID-Empfänger angelötet wurde.



RFID-TAGs mit RFID-Empfänger

## Verkabelung

Arduino	Sensor	LED
10	SDA	
13	SCK	
11	MOSI	
12	MISO	
GND	GND	
9	RST	
3.3 V	3.3 V	
2		Langes Bein



fritzing

Schaltplan

## Code

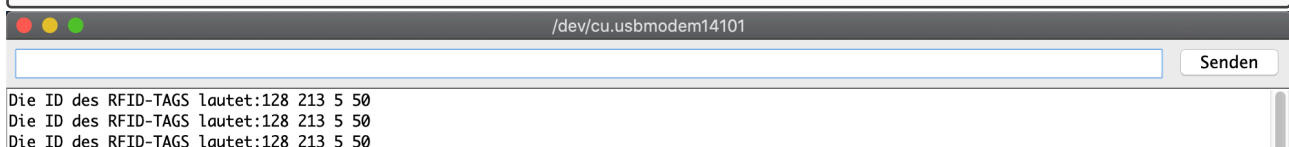
Es muss eine Library im Arduino Programm hinzugefügt werden. Wir verwenden die Library „MFRC522“ von GithubCommunity. **Siehe Bibliotheksverwaltung**

Zunächst wollen wir die eindeutige ID (UID) unseres RFID TAGS ermitteln. Wir schauen uns einmal mithilfe des folgenden Codes an, wie sie lautet:

```

1  #include <MFRC522.h>           // RFID-Bibliothek hinzufügen.
2  #include <SPI.h>              // SPI-Bibliothek hinzufügen.
3  #define SS_PIN 10            // SDA an Pin 10.
4  #define RST_PIN 9           // RST an Pin 9.
5  MFRC522 mfrc522(SS_PIN, RST_PIN); // RFID-Empfänger benennen.
6
7  void setup() {
8      Serial.begin(9600); // Seriellen Monitor starten.
9      SPI.begin();       // SPI-Verbindung aufbauen.
10     mfrc522.PCD_Init(); // Initialisierung des RFID-Empfängers.
11 }
12
13 void loop() {
14     if (!mfrc522.PICC_IsNewCardPresent()) { // Wenn kein TAG in Reichweite ist...
15         return; // ...springt das Programm zurück an den Anfang ( zu void loop() )
16     }
17
18     if (!mfrc522.PICC_ReadCardSerial()) { // Wenn kein TAG ausgewählt wurde...
19         return; // ...springt das Programm zurück an den Anfang ( zu void loop() )
20     }
21
22     Serial.print("Die ID des RFID-TAGS lautet:"); // Der Text in
23                                                    // Anführungsstrichen wird in
24                                                    // den Serial Monitor
25                                                    // geschrieben.
26
27     for (byte i = 0; i < mfrc522.uid.size; i++) {
28         // Dann wird die UID ausgelesen, die aus vier einzelnen Blöcken besteht und
29         // der Reihe nach an den Serial Monitor gesendet. Die Endung DEC bedeutet,
30         // dass die vier Blöcke der UID als Dezimalzahlen ausgegeben werden, da sie
31         // ursprünglich als Hexadezimal Zahl vorliegt. (z.B. F3 F8 A6 0)
32         Serial.print(mfrc522.uid.uidByte[i], DEC);
33         Serial.print(" "); // Fügt zwischen die einzelnen Blöcke ein Leerzeichen
34                             // ein.
35     }
36 }

```



Ausgabe des seriellen Monitors.

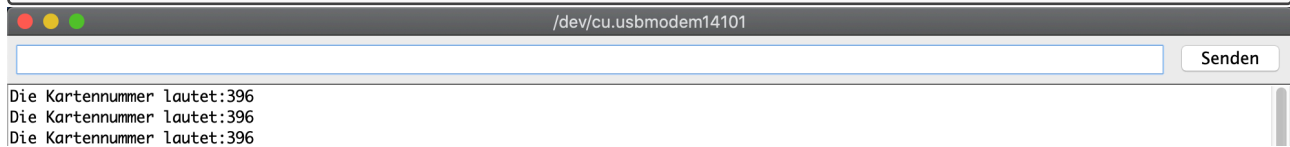
Nun können wir diese ID noch nicht verwenden, da es vier einzelne Zahlenblöcke sind. Daher passen wir den Code etwas an, indem wir einfach alle 4 Blöcke miteinander addieren. Dazu ersetzen wir die Zeilen 27 - 35 mit diesem Code:

```

27     int code = 0; // Diese Variable speichert die UID als zusammenhängende Zahl
28                 // ab.
29     for (byte i = 0; i < mfrc522.uid.size; i++) {
30         code = (code + mfrc522.uid.uidByte[i]); // Addition der vier Zahlenblöcke.
31     }

```

```
32 Serial.print("Die Kartennummer lautet:"); // Ausgabe des Zahlencodes.  
33 Serial.println(code);
```



Ausgabe des seriellen Monitors.

Die gewonnene Kennzahl nutzen wir jetzt, um eine LED anzuschalten, wenn der korrekte RFID TAG erkannt wird. Wir ergänzen den Code nach Zeile 33 um die folgenden Zeilen:

```
if (code == 396) {  
    digitalWrite(2, HIGH); // ...dann soll die LED an Pin 2 leuchten...  
    delay(500);           // für eine halbe Sekunde  
    digitalWrite(2, LOW); // ...und danach wieder aus gehen.  
}
```

## Aufgaben

### ① Beschreibe, was der Code tut!

Die Pins und ein Objekt für den RFID-Empfänger werden gesetzt. Im Setup wird der serielle Monitor gestartet, eine Verbindung aufgebaut und der RFID-Empfänger initialisiert. In der Loop wird getestet ob ein TAG in Reichweite ist und ausgewählt wurde. Die ID des TAGS wird ausgegeben und in vier Blöcken, der Reihe nach und durch ein Leerzeichen getrennt an den seriellen Monitor gesendet. Zudem wird die Ausgabe des seriellen Monitors addiert, in einen Zahlencode umgewandelt und ausgegeben. Dieser Code wird schließlich zur Steuerung einer LED genutzt.

### ② Was ist ein RFID-Reader und welche Aufgabe hat er?

RFID steht für Radio Frequency Identification. Es wird ein kontaktloser Datenaustausch zwischen einem RFID-Transponder (bei uns der RIFD-RC522) und einem RFID-Schreib-/Lesegerät (hier der Schlüsselanhänger) ermöglicht. Der Reader hat die Aufgabe verschiedene Objektinformationen z.B. einen Code auszulesen.

Das Material und dessen Inhalte sind - sofern nicht anders angegeben - lizenziert unter der Creative Commons Lizenz CC BY-NC-SA 4.0 (für den vollständigen Lizenztext siehe <https://creativecommons.org/licenses/by-sa/4.0/legalcode>)

