



## Willkommen!

Vielen Dank, dass Sie sich für unseren *AZ-Delivery RFID RC522 Reader mit RFID Chip und RFID Card* entschieden haben. Auf den nachfolgenden Seiten werden wir Ihnen erklären, wie Sie das Gerät einrichten und nutzen können.

Viel Spaß!



## Anwendungsbereiche

Bildung und Lehre: Einsatz in Schulen, Hochschulen und Ausbildungseinrichtungen zur Vermittlung von Grundlagen der Elektronik, Programmierung und eingebetteten Systemen. Forschung und Entwicklung: Verwendung in Forschungs- und Entwicklungsprojekten zur Erstellung von Prototypen und Experimenten in den Bereichen Elektronik und Informatik. Prototypenentwicklung: Einsatz in der Entwicklung und Erprobung neuer elektronischer Schaltungen und Geräte. Hobby und Maker-Projekte: Verwendung durch Elektronikenthusiasten und Hobbyisten zur Entwicklung und Umsetzung von DIY-Projekten.

## Erforderliche Kenntnisse und Fähigkeiten

Grundlegendes Verständnis der Elektronik und Elektrotechnik. Kenntnisse in der Programmierung, insbesondere in der Programmiersprache C/C++. Fähigkeit, Schaltpläne zu lesen und einfache Schaltungen zu entwerfen. Erfahrung im Umgang mit elektronischen Komponenten und Löten.

## Betriebsbedingungen

Das Produkt darf nur mit den im Datenblatt spezifizierten Spannungen betrieben werden, um Beschädigungen zu vermeiden. Eine stabilisierte Gleichstromquelle ist zum Betrieb erforderlich. Bei der Verbindung mit anderen elektronischen Komponenten und Schaltungen sind die maximalen Strom- und Spannungsgrenzen zu beachten, um Überlastungen und Schäden zu vermeiden.

## Umweltbedingungen

Das Produkt sollte in einer sauberen, trockenen Umgebung verwendet werden, um Schäden durch Feuchtigkeit oder Staub zu vermeiden. Schützen Sie das Produkt vor direkter Sonneneinstrahlung (UV)

## Bestimmungsgemäße Verwendung

Das Produkt ist das für den Einsatz in Bildungs-, Forschungs- und Entwicklungsumgebungen konzipiert wurde. Es dient zur Entwicklung, Programmierung und Prototypenentwicklung von elektronischen Projekten und Anwendungen. Das Sensor Produkt ist nicht als fertiges Verbraucherprodukt gedacht, sondern als Werkzeug für technisch versierte Nutzer, darunter Ingenieure, Entwickler, Forscher und Studenten.

## Nicht bestimmungsgemäße vorhersehbare Verwendung

Das Produkt eignet sich nicht für den Industriellen Einsatz oder sicherheitsrelevante Anwendungen. Eine Verwendung des Produkts in Medizingeräten oder für Zwecke der Luft- und Raumfahrt ist nicht zulässig

## Entsorgung

Nicht mit dem Hausmüll entsorgen! Ihr Produkt ist entsprechend der europäischen Richtlinie über Elektro- und Elektronik-Altgeräte umweltgerecht zu entsorgen. Die darin enthaltenen, wertvollen Rohstoffe können so der Wiederverwendung zugeführt werden. Die Anwendung dieser Richtlinie trägt zum Umwelt- und Gesundheitsschutz bei. Nutzen Sie die von Ihrer Kommune eingerichtete Sammelstelle zur Rückgabe und Verwertung elektrischer und elektronischer Altgeräte. WEEE-Reg.-Nr.: DE 62624346

## Elektrostatische Entladung

Achtung: Elektrostatische Entladungen können das Produkt beschädigen. Hinweis: Erden Sie sich, bevor Sie das Produkt berühren, indem Sie beispielsweise ein antistatisches Armband tragen oder eine geerdete Metalloberfläche berühren.

## Sicherheitshinweise

Obwohl unser Produkt den Anforderungen der RoHS-Richtlinie (2011/65/EU) entspricht und keine gefährlichen Stoffe in über den Grenzwerten zulässigen Mengen enthält, können dennoch Rückstände vorhanden sein. Beachten Sie die folgenden Sicherheitshinweise, um chemische Gefährdungen zu vermeiden: Achtung: Beim Löten können Dämpfe entstehen, die gesundheitsschädlich sein können. Hinweis: Verwenden Sie einen Lötdampfabsauger oder arbeiten Sie in einem gut belüfteten Bereich. Tragen Sie gegebenenfalls eine Atemschutzmaske. Achtung: Einige Personen könnten empfindlich auf bestimmte Materialien oder Chemikalien reagieren, die im Produkt enthalten sind. Hinweis: Sollten Hautreizungen oder allergische Reaktionen auftreten, unterbrechen Sie die Nutzung und suchen Sie gegebenenfalls einen Arzt auf. Achtung: Halten Sie das Produkt außerhalb der Reichweite von Kindern und Haustieren, um versehentlichen Kontakt und Verschlucken von Kleinteilen zu vermeiden. Hinweis: Bewahren Sie das Produkt in einem sicheren, geschlossenen Behälter auf, wenn es nicht verwendet wird. Achtung: Vermeiden Sie den Kontakt des Produkts mit Nahrungsmitteln und Getränken. Hinweis: Lagern und verwenden Sie das Produkt nicht in der Nähe von Lebensmitteln, um eine Kontamination zu verhindern. Obwohl unser Produkt den

Anforderungen der RoHS-Richtlinie (2011/65/EU) entspricht und keine gefährlichen Stoffe in über den Grenzwerten zulässigen Mengen enthält, können dennoch Rückstände vorhanden sein. Beachten Sie die folgenden Sicherheitshinweise, um chemische Gefährdungen zu vermeiden: Achtung: Beim Löten können Dämpfe entstehen, die gesundheitsschädlich sein können. Hinweis: Verwenden Sie einen Lötdampfabsauger oder arbeiten Sie in einem gut belüfteten Bereich. Tragen Sie gegebenenfalls eine Atemschutzmaske. Achtung: Einige Personen könnten empfindlich auf bestimmte Materialien oder Chemikalien reagieren, die im Produkt enthalten sind. Hinweis: Sollten Hautreizungen oder allergische Reaktionen auftreten, unterbrechen Sie die Nutzung und suchen Sie gegebenenfalls einen Arzt auf. Achtung: Halten Sie das Produkt außerhalb der Reichweite von Kindern und Haustieren, um versehentlichen Kontakt und Verschlucken von Kleinteilen zu vermeiden. Hinweis: Bewahren Sie das Produkt in einem sicheren, geschlossenen Behälter auf, wenn es nicht verwendet wird. Achtung: Vermeiden Sie den Kontakt des Produkts mit Nahrungsmitteln und Getränken. Hinweis: Lagern und verwenden Sie das Produkt nicht in der Nähe von Lebensmitteln, um eine Kontamination zu verhindern. Das Produkt enthält empfindliche elektronische Komponenten und scharfe Kanten. Unsachgemäßes Umgang oder Montage kann zu Verletzungen oder Beschädigungen führen. Beachten Sie die folgenden Sicherheitshinweise, um mechanische Gefährdungen zu vermeiden: Achtung: Die Platine und die Anschlüsse des Produkts können scharfe Kanten aufweisen. Gehen Sie vorsichtig vor, um Schnittverletzungen zu vermeiden. Hinweis: Tragen Sie bei der Handhabung und Montage des Produkts geeignete Schutzhandschuhe. Achtung: Vermeiden Sie übermäßigen Druck oder mechanische Belastung der Platine und der Komponenten. Hinweis: Montieren Sie das Produkt nur auf stabilen und ebenen Oberflächen. Verwenden Sie geeignete Abstandshalter und Gehäuse, um mechanische Belastungen zu minimieren. Achtung: Stellen Sie sicher, dass das Produkt sicher befestigt ist, um unbeabsichtigtes Verrutschen oder Herunterfallen zu verhindern. Hinweis: Verwenden Sie passende Unterlage oder eine sicheren Befestigung in Gehäusen oder auf Montageplatten. Achtung: Achten Sie darauf, dass alle Kabelverbindungen sicher und korrekt angeschlossen sind, um Zugbelastungen und versehentliches Herausziehen zu vermeiden. Hinweis: Führen Sie Kabel so, dass sie nicht unter Spannung stehen und keine Stolpergefahr darstellen. Das Produkt arbeitet mit elektrischen Spannungen und Strömen, die bei unsachgemäßem Gebrauch zu elektrischen Schlägen, Kurzschlägen oder anderen Gefahren führen können. Beachten Sie die folgenden Sicherheitshinweise, um elektrische Gefährdungen zu vermeiden: Achtung: Verwenden Sie das Produkt nur mit den spezifizierten Spannungen. Hinweis: Die Leistungsgrenzen des Produkts finden Sie im dazugehörigen Datenblatt Achtung: Vermeiden Sie Kurzschlüsse zwischen den Anschlüssen und Komponenten des Produkts Hinweis: Achten Sie darauf, dass keine leitenden Objekte die Platine berühren oder überbrücken. Verwenden Sie isolierte Werkzeuge und beachten Sie die Anordnung der Verbindungen. Achtung: Führen Sie keine Arbeiten am Produkt durch, wenn es mit einer Stromquelle verbunden ist. Hinweis: Trennen Sie das Produkt von der Stromversorgung, bevor Sie Änderungen an der Schaltung vornehmen oder Komponenten anschließen bzw. entfernen. Achtung: Überschreiten Sie nicht die spezifizierten Stromstärken für die Ein- und Ausgänge des Produkts. Hinweis: Die Leistungsgrenzen des Produkts finden sich in den technischen Spezifikationen oder im Datenblatt Achtung: Stellen Sie sicher, dass die verwendeten Stromquellen stabil und korrekt dimensioniert sind. Hinweis: Verwenden Sie nur geprüfte und geeignete Netzteile, um Spannungsschwankungen und Überlastungen zu vermeiden. Achtung: Halten Sie ausreichenden Abstand zu spannungsführenden Teilen ein, um unabsichtlichen Kontakt zu vermeiden. Hinweis: Sorgen Sie entsprechend der verwendeten Spannung für eine sichere und übersichtliche Anordnung der Verkabelung. Achtung: Verwenden Sie isolierende Gehäuse oder Schutzbabdeckungen, um das Produkt vor direktem Kontakt zu schützen. Hinweis: Setzen Sie das Produkt in ein nicht leitendes Gehäuse ein, um versehentliche Berührungen und Kurzschlüsse zu vermeiden. Das Produkt und die darauf befindlichen Komponenten können sich während des Betriebs erwärmen. Unsachgemäßes Umgang oder eine Überlastung des Produkts kann zu Verbrennungen, Beschädigungen oder Bränden führen. Beachten Sie die folgenden Sicherheitshinweise, um thermische Gefährdungen zu vermeiden: Achtung: Stellen Sie sicher, dass das Produkt innerhalb der empfohlenen Betriebstemperaturen verwendet wird. Hinweis: Der empfohlene Betriebstemperaturbereich liegt typischerweise zwischen -40°C und +85°C. Überprüfen Sie die spezifischen Angaben im Datenblatt des Produkts. Achtung: Platzieren Sie das Produkt nicht in der Nähe von externen Wärmequellen wie Heizkörpern oder direkter Sonnenstrahlung. Hinweis: Sorgen Sie dafür, dass das Produkt in einem kühlen und gut belüfteten Bereich betrieben wird. Achtung: Stellen Sie sicher, dass das Produkt gut belüftet ist, um eine Überhitzung zu vermeiden. Hinweis: Verwenden Sie Lüfter oder Kühlkörper, wenn das Produkt in einem geschlossenen Gehäuse betrieben wird oder in einer Umgebung mit eingeschränkter Luftzirkulation. Achtung: Montieren Sie das Produkt auf hitzebeständigen Oberflächen und in hitzebeständigen Gehäusen. Hinweis: Verwenden Sie Materialien für Gehäuse, die hohe Temperaturen aushalten können, um eine Beschädigung oder Feuergefahr zu vermeiden. Achtung: Implementieren Sie eine Überwachung der Temperatur bei Verwendung eines Gehäuses und gegebenenfalls Schutzmechanismen, die das Produkt abschalten, wenn es überhitzt. Hinweis: Verwenden Sie Temperatutfühler und entsprechende Software, um die Temperatur des Produkts zu überwachen und das System bei Bedarf abzuschalten. Achtung: Vermeiden Sie Überlastungen, die zu übermäßiger Erwärmung der Komponenten führen können. Hinweis: Überschreiten Sie nicht die spezifizierten Grenzwerte für Strom und Spannung, um eine Überhitzung zu verhindern. Achtung: Kurzschlüsse können erhebliche Hitze entwickeln und Brände verursachen. Hinweis: Stellen Sie sicher, dass alle Verbindungen korrekt und sicher sind und dass keine leitenden Objekte unbeabsichtigt Kurzschlüsse verursachen können.



RFID bedeutet Radiofrequenz-Identifikation. RFID nutzt elektromagnetische Felder, um Daten über kurze Distanzen zu übertragen. RFID-Tags werden in vielen Branchen verwendet, zum Beispiel kann ein RFID-Tag, der während der Produktion an einem Auto angebracht wird, verwendet werden, um den Fortgang auf dem Montageband zu verfolgen, RFID-getagte Arzneimittel können in Lagerhallen verfolgt werden, die Implantation von RFID-Mikrochips in Vieh und Haustieren ermöglicht positive Identifizierung der Tiere usw.

Ein RFID-System besteht aus zwei Teilen: einem Tag (sprich "Tegg") und einem Lesegerät. RFID-Tags sind in einen Sender und einen Empfänger eingebettet. Die RFID-Komponente auf den Tags besteht aus zwei Teilen: einem Mikrochip, der Informationen speichert und verarbeitet, und einer Antenne zum Empfangen und Senden eines Signals. Das Tag enthält die spezifische Seriennummer für ein bestimmtes Objekt; es muss sich nicht in direkter Sicht zum Lesegeräts befinden.

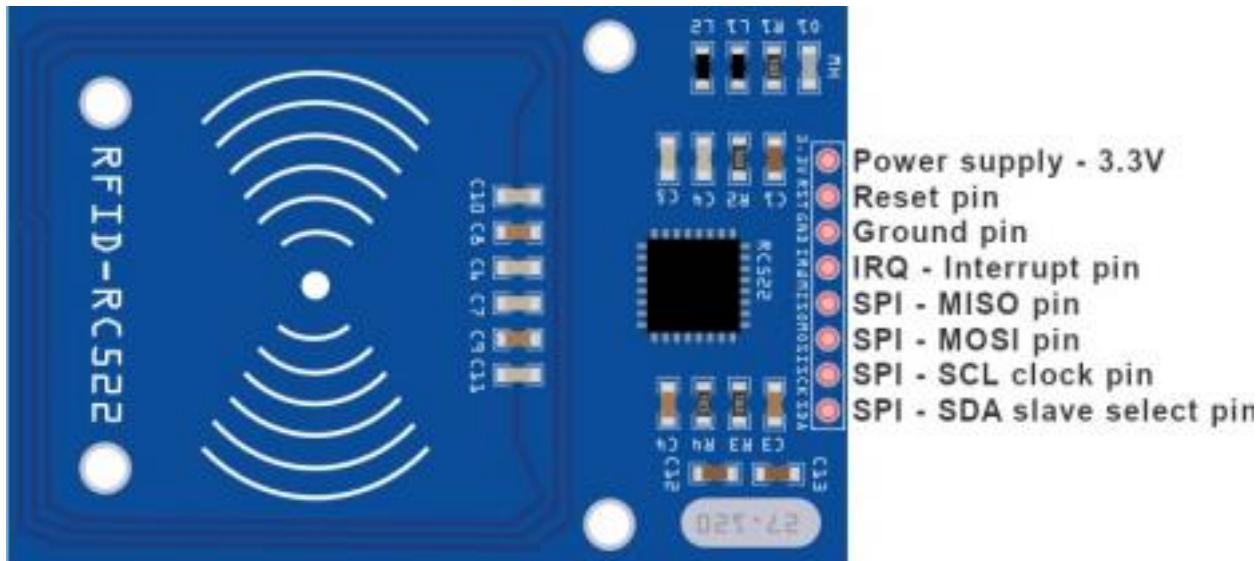
Um die auf einem Tag kodierten Informationen zu lesen, sendet ein Funksender-Empfänger, der sogenannte Reader, ein Signal über eine Antenne an das Tag. Das Tag antwortet mit den in seine Speicherbank geschriebenen Informationen. Das Lesegerät überträgt dann die Leseergebnisse an einen Mikrocontroller.

## Spezifikation

- » Chip: MFRC522
- » Operating frequency: 13.56MHz
- » Power supply voltage: 3.3V
- » Current: 13 - 26mA
- » Read Range: Approx 30mm
- » Communication: SPI interface
- » Max Data Transfer Rate: 10Mbit/s
- » Dimensions: 40 x 60mm [1.6 x 2.4in]

Das Lesegerät ist ein Nahfeldgerät, kein echtes Funkgerät. Das Lesegerät funktioniert nicht durch das Senden und Empfangen von Funksignalen, sondern durch das Erfassen kleiner Impedanzänderungen des abgestimmten Antennenkreises, die durch das RFID-Tag verursacht werden. Bewegen Sie das RFID-Tag aus dem Nahfeld der Antenne und es ist keine Wirkung zu messen. Das Lesegerät ist nicht wirklich ein Empfänger, sondern einedetektiert die Schwankungen der mittleren Ausgangsspannung im Schwingkreis.

## Pinbelegung des Lesegeräts (Reader)



**Spannungsversorgung nur mit 3,3V!**

**Nicht an 5V anschließen; das kann das Lesegerät zerstören!**

Das Lesegerät benutzt die SPI Schnittstelle (Serial Peripheral Interface) für die Kommunikation mit dem Microcontroller.

Der Interrupt IRQ pin ist normalerweise *HIGH* state, und beim Interrupt Ereignis wechselt der Status kurzzeitig auf *LOW*. Das Interrupt Ereignis geschieht, wenn das RFID tag im Nahfeld des Readers detektiert wird.

## Set-up der Arduino IDE

Wenn Sie die Arduino IDE noch nicht installiert haben, laden Sie das Programmpaket für Ihr Betriebssystem bei <https://www.arduino.cc/en/software> herunter.

## Downloads



The screenshot shows the download page for Arduino IDE 1.8.16. On the left, there's a large button for "Arduino IDE 1.8.16" with a teal icon. Below it, a description of the software and a link to the "Getting Started" page. Further down, there's a "SOURCE CODE" section with instructions about GitHub and PGP keys. On the right, a teal sidebar titled "DOWNLOAD OPTIONS" lists download links for Windows (Win 7 and newer, ZIP file, app), Linux (32 bits, 64 bits, ARM 32 bits, ARM 64 bits), and Mac OS X (10.10 or newer). A "Release Notes" link and a "Checksums (sha512)" link are also present.

Arduino IDE 1.8.16

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

**DOWNLOAD OPTIONS**

**Windows** Win 7 and newer  
**Windows** ZIP file  
**Windows app** Win 8.1 or 10 [Get](#)

**Linux** 32 bits  
**Linux** 64 bits  
**Linux** ARM 32 bits  
**Linux** ARM 64 bits

**Mac OS X** 10.10 or newer

[Release Notes](#) [Checksums \(sha512\)](#)

Für Windows laden Sie Windows ZIP file, entpacken die Datei, und starten im Verzeichnis die Datei "arduino.exe" durch Doppelklick. Beim ersten Mal werden dann ein Verzeichnis Arduino angelegt, in dem Sie u.a. Beispiel-Sketchen finden, und in dem später Ihre eigenen Sketche abgespeichert werden.

# Az-Delivery

Für Linux laden Sie die Datei mit der Erweiterung ".tar.xz" herunter, die Sie dann extrahieren müssen. Wenn Sie es extrahieren, gehen Sie in das extrahierte Verzeichnis und öffnen Sie das Terminal in diesem Verzeichnis. Sie müssen zwei „.sh“-Skripte ausführen, das erste heißt „arduino-linux setup.sh“ und das zweite heißt „install.sh“.

Führen Sie den folgenden Befehl aus, um das erste Skript im Terminal auszuführen:

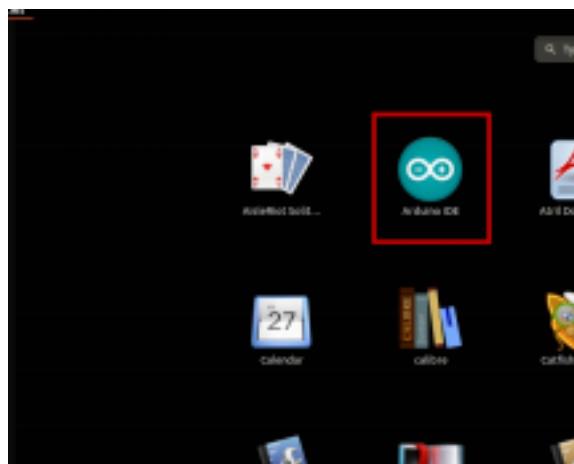
**sh arduino-linux-setup.sh user\_name**

**user\_name** - ist der Name des super user im Linux operating system. Danach werden Sie aufgefordert, das Passwort für den Superuser anzugeben. Warten Sie einige Minuten, bis das Skript alles abgeschlossen hat.

Führen Sie nach der Installation des ersten Skripts das zweite Skript namens "install.sh" aus. Führen Sie im Terminal den folgenden Befehl aus:

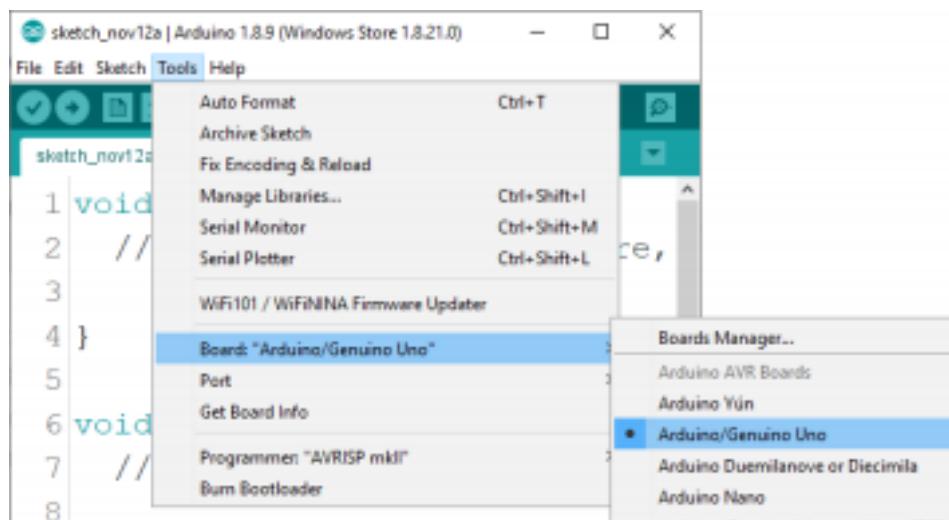
**sh install.sh**

Gehen Sie nach der Installation dieser Skripte zu Alle Apps, um die installierte Arduino IDE zu finden.



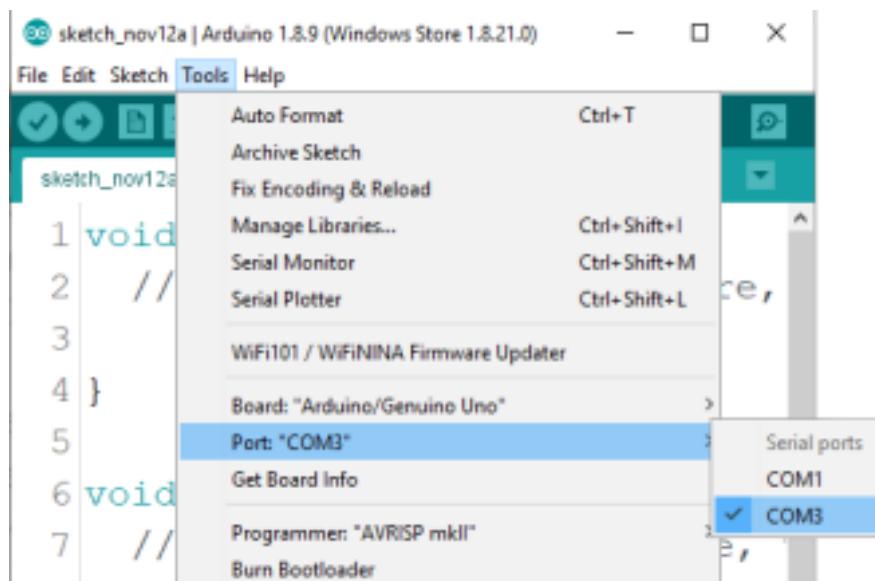
# Az-Delivery

Als Nächstes wählen wir auf der Registerkarte Werkzeuge > Board den passenden Micro Controller aus. Für unseren Micro Controller mit ATmega328p wählen Sie *Arduino/Genuino Uno*, wie auf dem folgenden Bild gezeigt:



# Az-Delivery

Danach wird der (virtuelle) COM-Port des Micro Controllers auf der Registerkarte Werkzeuge > Port ausgewählt..

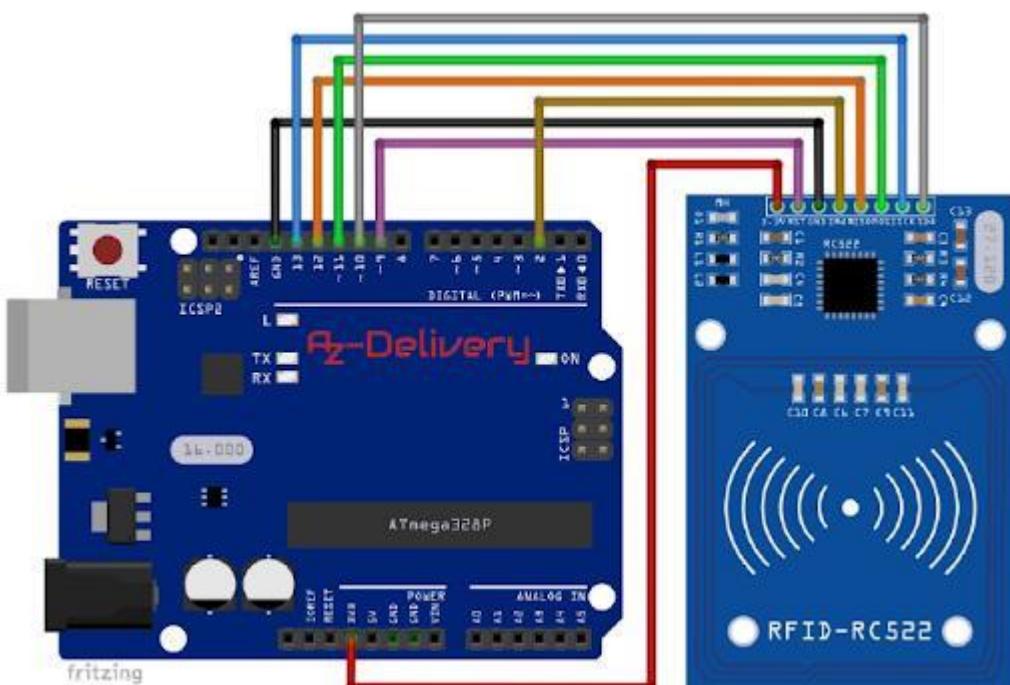


Bei Linux ist der Port Name z.B. "/dev/ttyUSBx" wobei "x" eine Integer Zahl zwischen 1 und 9 ist.

# Az-Delivery

## Anschluss des Readers an den ATmega328p

Schließen Sie den Reader an den Microcontroller mit ATmega328p an, wie auf dem nachstehenden Anschlussdiagramm (Schaltbild) gezeigt:

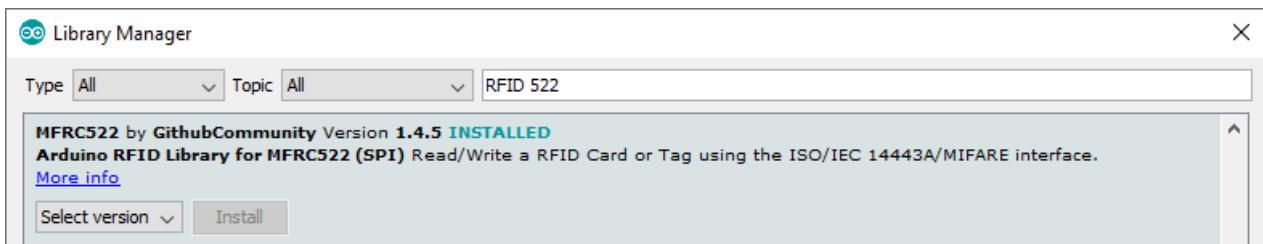


Module pin	> ATmega328p pin	
3.3V	> 3.3V !	Red wire
RST	> D9	Purple wire
GND	> GND	Black wire
IRQ	> D2	Ochre wire
MISO	> D12	Orange wire
MOSI	> D11	Green wire
SCK	> D13	Blue wire
SDA (SS)	> D10	Gray wire

# Az-Delivery

## Die Bibliothek (library) für die Arduino IDE

Um den Reader mit dem MicroController mit ATmega328p zu benutzen, wird die Benutzung der folgenden Programm-Bibliothek empfohlen. Dazu öffnen Sie in der Arduino IDE unter der Registerkarte Werkzeuge > *Bibliotheken verwalten*. Das nachstehend abgebildete kleine Fenster öffnet sich; dann geben Sie im Suchfenster “RFID 522” ein. Installieren Sie dann die Bibliothek “MFRC522” der “GithubCommunity”.



Bei der Installation der eigentlichen Bibliothek werden auch Beispiel-Sketche gespeichert, die Sie unter der Registerkarte Datei > Beispiele >MFRC522 finden. Wir benutzen hier das Beispiel “*MinimallInterrupt*”. Also:

*Datei > Beispiele > MFRC522 > MinimallInterrupt*



## Sketch Beispiel

```
#include <SPI.h>
#include <MFRC522.h>
#define RST_PIN 9 // Configurable, see typical pin layout above
#define SS_PIN 10 // Configurable, see typical pin layout above
#define IRQ_PIN 2 // Configurable, depends on hardware
MFRC522 mfrc522(SS_PIN, RST_PIN); // Create MFRC522 instance
MFRC522::MIFARE_Key key;
volatile bool bNewInt = false;
byte regVal = 0x7F;

void setup() {
    Serial.begin(9600); // Initialize serial communications with the PC
    SPI.begin(); // Init SPI bus
    mfrc522.PCD_Init(); // Init MFRC522 card

    // Read and printout the MFRC522 version (valid values 0x91 & 0x92) Serial.print(F("Ver:
    0x"));
    byte readReg = mfrc522.PCD_ReadRegister(mfrc522.VersionReg);
    Serial.println(readReg, HEX);

    pinMode(IRQ_PIN, INPUT_PULLUP); // Setup the IRQ pin // Allow the
    irq to be propagated to the IRQ pin
    regVal = 0xA0; // Rx IRQ
    mfrc522.PCD_WriteRegister(mfrc522.ComIEnReg, regVal);
    bNewInt = false; // Interrupt flag
    // Activate the interrupt
    attachInterrupt(digitalPinToInterrupt(IRQ_PIN), readCard, FALLING);

    do { // Clear a spurious interrupt at start
    ;
    } while (!bNewInt);
    bNewInt = false;
    Serial.println(F("End setup"));
}
```

# Az-Delivery

```
void loop() {
    if (bNewInt) { // New read interrupt
        Serial.print(F("Interrupt."));
        mfrc522.PICC_ReadCardSerial(); // Read the tag data
        // Show some details of the PICC
        Serial.print(F("Card UID:"));
        dump_byte_array(mfrc522.uid.uidByte, mfrc522.uid.size);

        Serial.println();
        clearInt(mfrc522);
        mfrc522.PICC_HaltA();
        bNewInt = false;
    }
    activateRec(mfrc522);
    delay(100);
}

// MFRC522 interrupt serving routine
void readCard() {
    bNewInt = true;
}

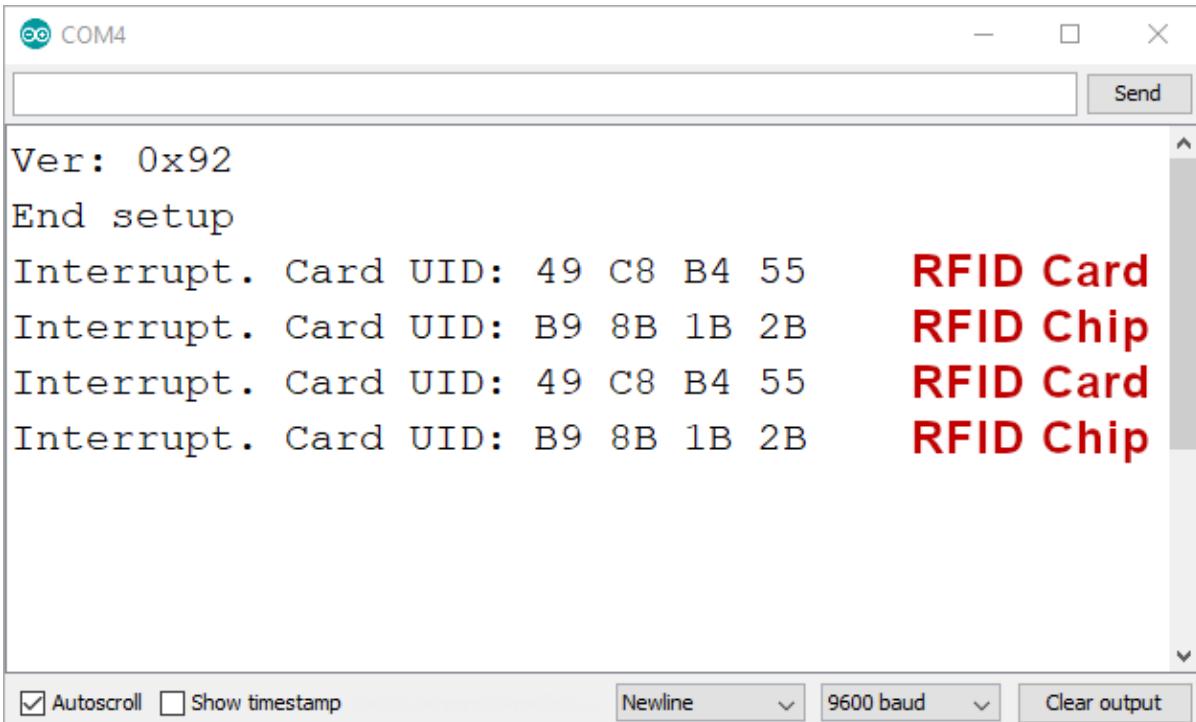
// Helper routine to dump a byte array as hex values to Serial
void dump_byte_array(byte *buffer, byte bufferSize) {
    for (byte i = 0; i < bufferSize; i++) {
        Serial.print(buffer[i] < 0x10 ? " 0" : " ");
        Serial.print(buffer[i], HEX);
    }
}

// The function sending to the MFRC522 the needed commands to activate the reception
void activateRec(MFRC522 mfrc522) {
    mfrc522.PCD_WriteRegister(mfrc522.FIFODataReg, mfrc522.PICC_CMD_REQA);
    mfrc522.PCD_WriteRegister(mfrc522.CommandReg, mfrc522.PCD_Transceive);
    mfrc522.PCD_WriteRegister(mfrc522.BitFramingReg, 0x87); }

// The function to clear the pending interrupt bits after ISR
void clearInt(MFRC522 mfrc522) {
    mfrc522.PCD_WriteRegister(mfrc522.ComIrqReg, 0x7F);
}
```

# Az-Delivery

Nach dem Hochladen des Sketches auf den MC mit ATmega328p wird der Serielle Monitor geöffnet, (*Werkzeuge > Serieller Monitor oder Mausklick auf das rechte Symbol*). Die Ausgabe sieht ähnlich dem folgenden Bild aus::



The screenshot shows the Arduino Serial Monitor window titled "COM4". The window has a "Send" button at the top right. The text output is as follows:

```
Ver: 0x92
End setup
Interrupt. Card UID: 49 C8 B4 55
Interrupt. Card UID: B9 8B 1B 2B
Interrupt. Card UID: 49 C8 B4 55
Interrupt. Card UID: B9 8B 1B 2B
```

Below the text, there are four red labels on the right side: "RFID Card", "RFID Chip", "RFID Card", and "RFID Chip". At the bottom of the monitor window, there are checkboxes for "Autoscroll" and "Show timestamp", and buttons for "Newline", "9600 baud", and "Clear output".

Wenn Sie eine RFID-Karte oder einen Chip in die Nähe des Lesegeräts bringen, erfolgt ein Interrupt und die UID-Nummer der Karte/des Chips wird an den Seriellen Monitor gesendet.

Am Anfang des Sketches werden zwei Bibliotheken importiert, SPI für die SPI-Schnittstelle und MFRC522 für den RFID-Reader. Dann werden drei Makros erstellt, eines für die Reset-Pin-Nummer, das zweite für die Slave-Select-Pin-Nummer und das letzte für die Interrupt-Pin-Nummer.

Danach werden das Objekt „mfrc522“, das den Reader in der Software repräsentiert, und die Schlüsselvariable instanziert. Die Schlüsselvariable wird verwendet, um die UID-Nummer der spezifischen Karte zu speichern.



Dann wird die Variable bNewInt erstellt. Wenn ein Interrupt auftritt, ändert sich der Status der Variable bNewInt auf true, was anzeigt, dass ein Interrupt aufgetreten ist. Dann prüfen wir im Code den Zustand der bNewInt-Variablen und geben die entsprechende Ausgabe an dem Seriellen Monitor aus (mehr dazu später im Text).

Dann erstellen wir die Variable regVal, die verwendet wird, um Register des Chips im RFID-Lesegerät einzurichten.

In der setup()-Funktion richten wir zuerst die serielle Schnittstelle mit einer Baudrate von 9600 ein. Danach richten wir die SPI-Schnittstelle ein und initialisieren das mfrc522-Objekt.

Später geben wir die MFRC522-Reader-Version auf dem Serial Monitor aus. Gültige Zahlen sind 0x91 und 0x92.

Dann richten wir den Pin-Modus des Interrupt-Pins auf INPUT mit internem PULL-UP-Widerstand ein. Danach hängen wir die Interrupt-Routine mit folgender Codezeile an:

`attachInterrupt(digitalPinToInterrupt(IRQ_PIN), readCard, FALLING)`  
Das erste Argument dieser Funktion ist die built-in function:

`digitalPinToInterrupt()`

die den `IRQ_PIN` (digital I/O pin 2) setzt, um das Interrupt Ereignis zu erkennen. Das zweite Argument ist die Funktion, die ausgeführt wird, wenn ein Interrupt auftritt. Und das dritte Argument ist die Flanke des digitalen Signals, die die Art des Interrupt-Ereignis festlegt.



Für dieses Beispiel verwenden wir die FALLENDE Flanke des digitalen Signals, das am IRQ\_PIN ankommt, als Interrupt-Ereignis. Der Wert dieses Arguments kann auch RISING oder BOTH sein, aber das verwenden wir in unserem Beispiel nicht.

Am Ende der `setup()`-Funktion löschen wir falsche Interrupt-Abfragen mit folgenden Codezeilen:

```
do { ; } while (!bNewInt);  
bNewInt = false;
```

In der Funktion `loop()` gibt es eine if-Anweisung, in der wir den Zustand der Variablen `bNewInt` überprüfen. Wenn der Zustand `bNewInt` wahr ist, ist ein Interrupt aufgetreten und wir geben die Ausgabe auf dem Seriellen Monitor aus. Die Ausgabe enthält die Meldung „Interrupt. Card UID: number“, wobei `number` die UID-Nummer der RFID-Karte ist, die den Interrupt verursacht hat. Am Ende der if-Anweisung setzen wir die Variable `bNewInt` auf false.

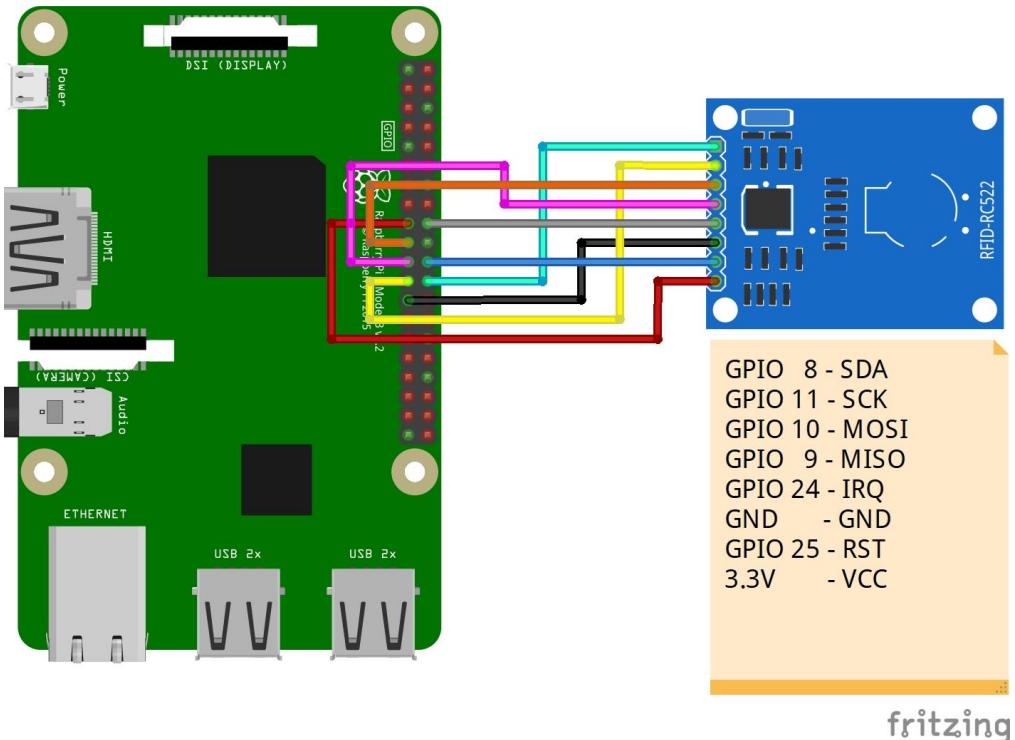
Am Ende der Funktion `loop()` richten wir die Register des Lesechips erneut ein und warten 100 Millisekunden.

Nach der Funktion `loop()` erstellen wir die Funktion `readCard()`, die ausgeführt wird, wenn der Interrupt auftritt. In der Funktion `readCard()` setzen wir nur den Zustand der Variablen `bNewInt` auf true. Danach erstellen wir drei Funktionen, die verwendet werden, um Register des Leserchips einzurichten und mit dem Reader zu kommunizieren.

# Az-Delivery

## Anschluss des Readers an den Raspberry Pi

Schließen Sie den Reader an den Raspberry Pi wie folgt an:



Module pin	> Raspberry Pi pin	
SDA (SS)	> GPIO8 [pin 24]	Cyan wire
SCK	> GPIO11 [pin 23]	Yellow wire
MOSI	> GPIO10 [pin 19]	Orange wire
MISO	> GPIO9 [pin 21]	Purple wire
IRQ	> GPIO24 [pin 18]	Gray wire
GND	> GND [pin 25]	Black wire
RST	> GPIO25 [pin 22]	Blue wire
3.3V	> 3V3 [pin 1 or 17]	Red wire



## Die Bibliothek für Python (Python-Modul)

Um den Reader mit dem Raspberry Pi zu verwenden, wird die Benutzung eines Python-Moduls (Bibliothek) empfohlen. Hier benutzen wir das Modul „*pi-rc522*“. Für die Installation öffnen wir das Terminal und aktualisieren zunächst das Betriebssystem, dann wird das Modul mit pip3 installiert:

```
sudo apt-get update
sudo apt-get upgrade
sudo pip3 install pi-rc522
```

Auf der folgenden Seite zeigen wir ein Beispiel-Programm.

**Anmerkung:** Für dieses Python-Modul muss der IRQ Pin angeschlossen sein.

Eine alternative Bibliothek, die ohne IRQ-Pin funktioniert, ist SimpleMFRC522.

Installation im Terminal:

```
sudo pip3 install mfrc522
```

Import in das Python-Programm:

```
from mfrc522 import SimpleMFRC522
```

Das Beispiel-Programm nutzt pi-rc522

# Az-Delivery

## Python script

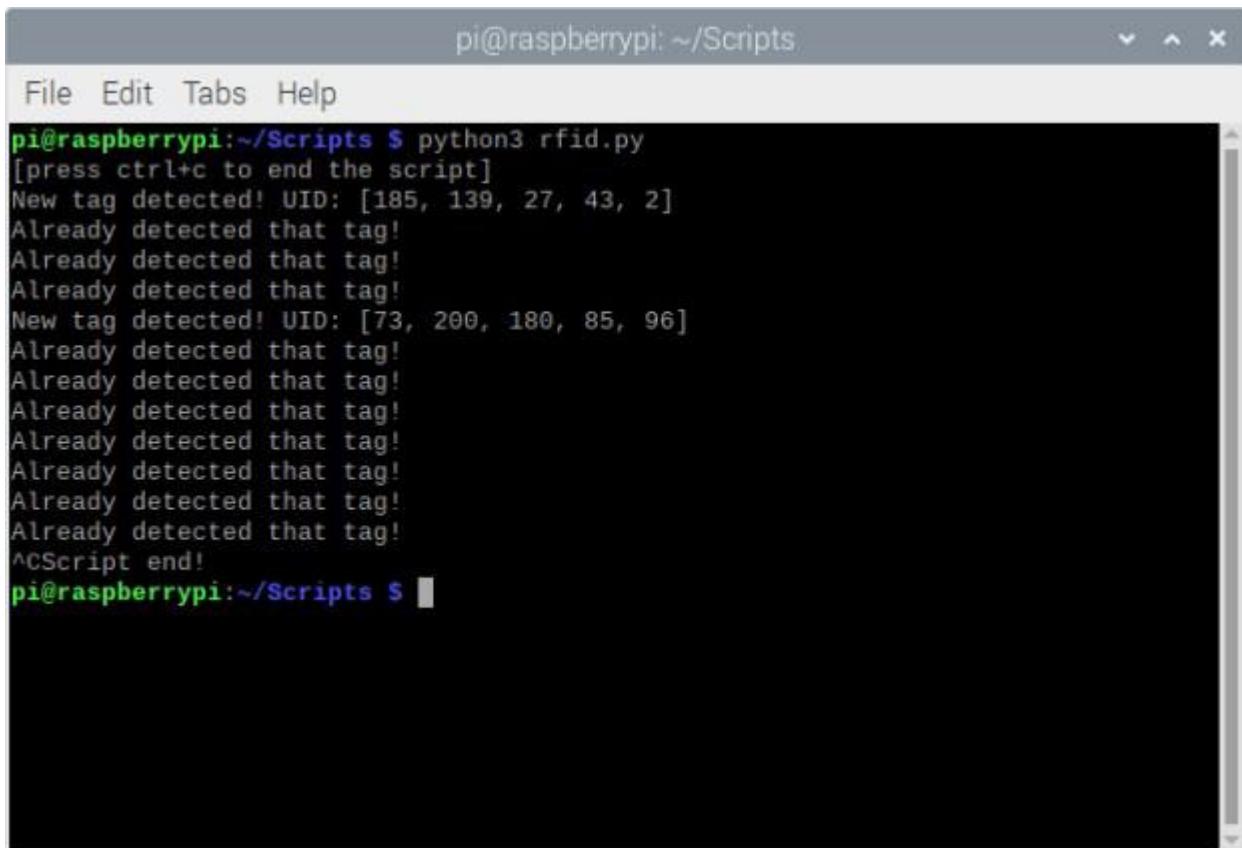
```
from pirc522 import RFID
from time import sleep
reader = RFID()
keys = []
key_read = True
print('[press ctrl+c to end the script]')
try:
    while True:
        reader.wait_for_tag()
        error, tag_type = reader.request()
        if not error:
            error, uid = reader.anticoll()
            if not error:
                if len(keys) > 0:
                    for key in keys:
                        if key == uid:
                            key_read = False
                            break
                else:
                    key_read = True
            else:
                key_read = True
        if key_read:
            keys.append(uid)
            print('New tag detected! UID: {}'.format(uid))
        reader.stop_crypto() # always call this when done working
        key_read = False
    else:
        print('Already detected that tag!')
    sleep(0.1)
# Scavenging work after the end of the program
except KeyboardInterrupt:
    print('Script end!')
finally:
    reader.cleanup() # Calls GPIO cleanup
```

# Az-Delivery

Das Programm wird z.B. mit dem Namen "rfid.py" gespeichert. Dann öffnet man das Terminal, wechselt in das Verzeichnis, in dem das Programm abgespeichert wurde und startet es mit:

**python3 rfid.py**

The output should look like the output on the image below:



pi@raspberrypi:~/Scripts

```
pi@raspberrypi:~/Scripts $ python3 rfid.py
[press ctrl+c to end the script]
New tag detected! UID: [185, 139, 27, 43, 2]
Already detected that tag!
Already detected that tag!
Already detected that tag!
New tag detected! UID: [73, 200, 180, 85, 96]
Already detected that tag!
^CScript end!
```

pi@raspberrypi:~/Scripts \$

Das Programm wird mit "CTRL + C" beendet.



Am Anfang des Skripts importieren wir zwei Bibliotheken, eine namens pirc522, die wir gerade installiert haben, und die andere ist für die Zeitfunktion sleep.

Dann erstellen wir ein Reader-Objekt mit der folgenden Codezeile:  
`reader = RFID()`

Sie können den SDA-Pin des Lesegeräts mit CE1 (GPIO7 [Pin 26]) anstelle von CE0 (GPIO8 [Pin 24]) verbinden. Wenn Sie dies tun, müssen Sie ein Reader-Objekt mit der folgenden Codezeile erstellen:

`reader = RFID(bus=0, device=1)`

Außerdem können Sie den RST-Pin mit jedem anderen freien GPIO-Pin verbinden. In diesem Fall müssen Sie ein Reader-Objekt mit der folgenden Codezeile erstellen: `reader = RFID(pin_rst=number)`  
Dabei ist Zahl die Pin-Nummer der Raspberry Pi-Platine (nicht GPIO-Nummer, sondern Pin-Nummer).

Außerdem können Sie den IRQ-Pin mit jedem anderen freien GPIO-Pin verbinden. In diesem Fall müssen Sie ein Reader-Objekt mit der folgenden Codezeile erstellen:

`reader = RFID(pin_irq=number)`

Dabei ist Zahl die Pin-Nummer der Raspberry Pi-Platine (nicht GPIO-Nummer, sondern Pin-Nummer).

Danach erstellen wir eine Liste namens keys, in der wir neue UID-keys speichern. Außerdem erstellen wir eine boolesche Variable namens key\_read, die verwendet wird, um zu überprüfen, ob key bereits in der Schlüsselliste keys enthalten ist.



Dann erstellen wir try-except-finally-Block. Wir tun dies, um Tastaturunterbrechungen zu erkennen. Die Tastaturunterbrechung tritt auf, wenn wir STRG + C auf der Tastatur drücken. Schließlich wird ein Teil des try-except-finally-Blocks verwendet, um die GPIO-Pins von allen verwendeten Schnittstellen oder definierten Pin-Modi nach einem Tastatur-Interrupt zu bereinigen.

Im try-Teil des try-except-finally-Blocks erstellen wir eine Endlosschleife (while True:). In dieser Schleife warten wir zunächst darauf, dass das Tag erkannt wird. Wenn ein Tag erkannt wird, erfolgt ein Interrupt auf IRQ\_PIN. Wenn ein Interrupt auftritt, werden die Tag-Daten gelesen.

Die Funktion reader.request() gibt ein Tupel von zwei Elementen zurück. Das erste Element ist ein boolescher Wert, ob ein Fehler vorliegt (True, es liegt ein Fehler vor, False, es liegt kein Fehler vor), und das zweite Element stellt den Typ des Tags dar (den wir im Skript nicht verwenden).

Dann prüfen wir, ob ein Fehler vorliegt. Wenn keine Fehler vorliegen, fahren wir mit dem Lesen der UID des Tags fort.

Die Funktion reader.anticoll() gibt ein Tupel von zwei Elementen zurück. Das erste Element ist ein boolescher Wert, ob ein Fehler vorliegt (Wahr, es liegt ein Fehler vor, Falsch, es liegt kein Fehler vor) und das zweite Element ist die UID-Nummer des Tags.



Dann prüfen wir erneut, ob beim Lesen der UID ein Fehler aufgetreten ist. Wenn keine Fehler vorliegen, überprüfen wir, ob die UID-Nummer bereits in der Schlüsselliste enthalten ist. Wenn die UID-Nummer nicht in der Schlüsselliste enthalten ist, fügen wir eine neue Nummer hinzu und drucken die Meldung „*New tag detected! UID: number*“, wobei Zahl die UID-Nummer des Tags ist. Wenn sich die UID-Nummer des Tags in der Schlüsselliste befindet, drucken wir die Meldung „*Already detected that tag!*“.

Wenn wir mit dem Lesen der vom Reader gesendeten Daten fertig sind, müssen wir die folgende Funktion ausführen:

```
reader.stop_crypto()
```

Am Ende des Endlosschleifenblocks rufen wir mit sleep(0.1) ein Wartezeitintervall von 100 Millisekunden zwischen zwei Schleifen des Endlosschleifenblocks auf.

**Geschafft!**

**Nun können Sie den Reader in Ihren Projekten  
benutzen.**



Jetzt ist es an der Zeit, die Projekte selbst zu lernen und zu erstellen. Dies können Sie mit Hilfe vieler Beispielskripte und anderer Tutorials tun, die Sie in unserem Blog oder im Internet finden.

**Wenn Sie hochwertige Mikroelektronik und Zubehör suchen, sind Sie bei der AZ-Delivery Vertriebs GmbH an der richtigen Adresse. Sie erhalten zahlreiche Anwendungsbeispiele, vollständige Installationsanleitungen, eBooks, Bibliotheken und Unterstützung durch unsere technischen Experten.**

<https://az-delivery.de>

**Viel Vergnügen!**

**Impressum**

<https://az-delivery.de/pages/about-us>