

# RFID

## Welchen RFID Leser /Schreiber kann für Raspberry PI oder Aduino nutzen.

Für den Arduino ist man sehr eingeschränkt, da man eine serielle Schnittstelle braucht (Aktueller Stand). Aus diesem Grund kann man nur folgende Leser

nutzen: <https://www.karley.de/index.php?route=product/search&search=raspberry>

Diese Leser können auch nur bestimmte Standards. NTAG203 NFC ist so z.B. nicht möglich. Es gibt natürlich noch weitere Leser.

## Raspberry PI

Mit dem Syris RD300-H1 ist man recht flexibel, da dieser eine USB Schnittstelle hat. Möchte man nur die Seriennummer (UID) Lesen, so kann man den HID Modus benutzen, der Standardgemäß eingestellt ist. Will man Daten von Sektoren Lesen und Schreiben, kann man per Firmware diesen auf den Virtuellen COM Port Modus umschalten und den USB CDC driver.

(0E6A0316\_USBEasyCOM\_v1.06.inf) nutzen. Linux fügt dann eine serielle Schnittstelle hinzu über die man dann per Python oder anderen Programmiersprachen mit dem Syris RD300-H1 kommunizieren kann. Eine vollständige Dokumentation über die seriellen Befehle ist dann vorhanden und kann beim Verkauf mitgeliefert werden. Hier muss aber der Programmierer vollständig mit der seriellen Schnittstelle umgehen können. Er sendet also Befehle über die Serielle Schnittstelle und muss entsprechend die Antwort abwarten und auswerten. Hier ein Beispiel für so einen Befehl:

### 3.4 Ultralight/NTAG Write Data (0x14)

Request				
	LEN	CMD	{DATA}	Description
<STX>	[14]	[14]	[00] + [00] + [Block Number] + [Data]*16	[Block Number]: e.g. [00]  [Data]: e.g. [FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF]

Figure 3.4.1 –Ultralight/NTAG Write Data - Request

Response					
	LEN	CMD	STATUS	{DATA}	Description
<STX>	[02]	[14]	[10]		Command error
<STX>	[02]	[14]	[01]		No card or invalid Key
<STX>	[12]	[14]	[00]	[Data]*16	Command complete

Figure 3.4.2 –Ultralight/NTAG Write Data - Response

Ein einfaches SDK für Python oder Linux gibt es nicht. Man muss schon den seriellen Port beherrschen - sprich Abfragen, auf Daten warten, Daten senden etc. das wäre die eigene Arbeit. Zudem muss man sich mit der Speicherbelegung der entsprechenden Chips (z.B. NTAG ) auskennen. Sie müssen also wissen welche Bereiche Sie überhaupt schreiben und lesen können und ggf. für die

# RFID

Passworte Kryptographie Libraries integrieren. Diese Informationen findet man in den offiziellen Datenblättern der Hersteller.

Sollten Ihnen jetzt schon ein paar Begriffe fehlen, würde ich das nicht empfehlen, dann wäre es günstiger einen Intel NUC mit Windows zu nehmen und dann mit C das SDK anzusprechen. Karley kann Ihnen die SDK Entwicklung jederzeit auch für Raspberry anbieten - solche Projekte sind aber recht Zeitintensiv und Kostspielig - je nach Wunsch der Integration.

Außer in Projekten - bei zusätzlicher Abrechnung - können wir auch keinen erweiterten Support bieten wie: Ich sende ein H13 001 22233, bekomme aber keine Antwort - warum. So etwas können wir zwar prinzipiell nachstellen, ist aber nicht im Artikelpreis enthalten. Beim Raspberry muss leider einfach klar sein, das man viel selbst basteln muss - es ist ja auch als BastelPC konzipiert.

Eindeutige ID: #1349

Verfasser: n/a

Letzte Änderung: 2019-02-19 17:11