Terminals / Smartcard Reader

Terminals wie z.B. RFID Lesegeräte, welche über MQTT eingebunden werden können, z. B. für die Freischaltung von Maschinen. Damit benötigt der Benutzer nicht zwangsweise eine App, um Ressourcen zu verwenden.

Es gibt aktuell den FabReader v1 und den v2. Beide sind derzeit obsolet. Eine Version v3 ist in Entwicklung.

Details zu den passenden DESFire Karten: FabFire und NFC Tags

- FabAccess Case - FabReader v1 Gehäuse vom Hackwerk e.V.
- FabReader v1 (abgekündigt)
- FabReader v2 (abgekündigt)
- FabReader v3
- ACS ACR122U - USB NFC Reader
- OpenUnitState

FabAccess Case - FabReader v1 Gehäuse vom Hackwerk e.V.

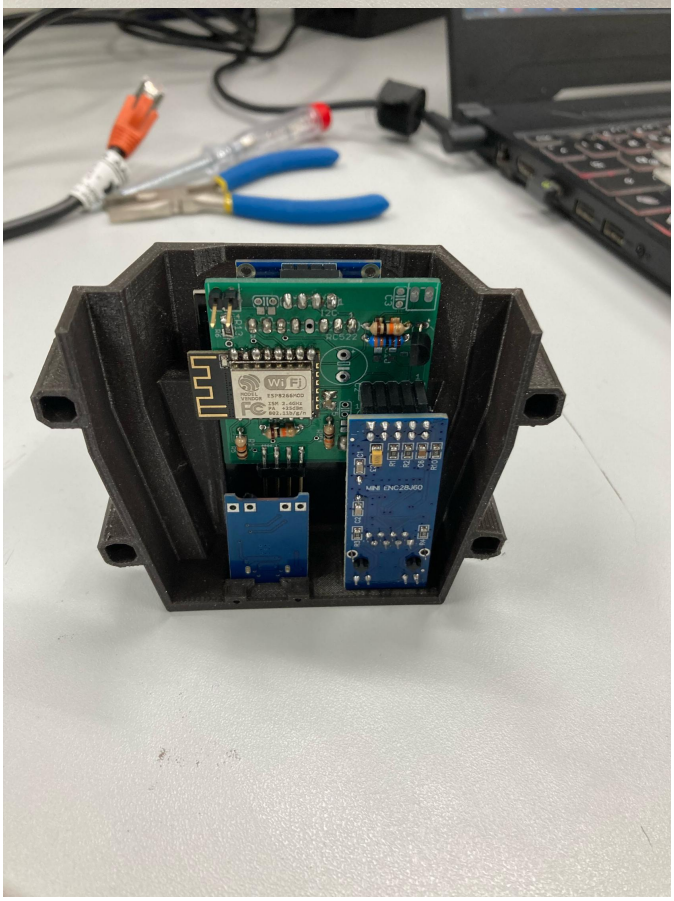
Was ist das?

Dieses Projekt bietet ein alternatives 3D-Druck Gehäuse für den FabReader v1 (abgekündigt). Das FabAccess Case wurde mit der FOS-Software Blender vom Hackwerk e.V. Aalen designed.









Stückliste

Sources

Die Quelldateien zum 3D-Drucken finden sich unter

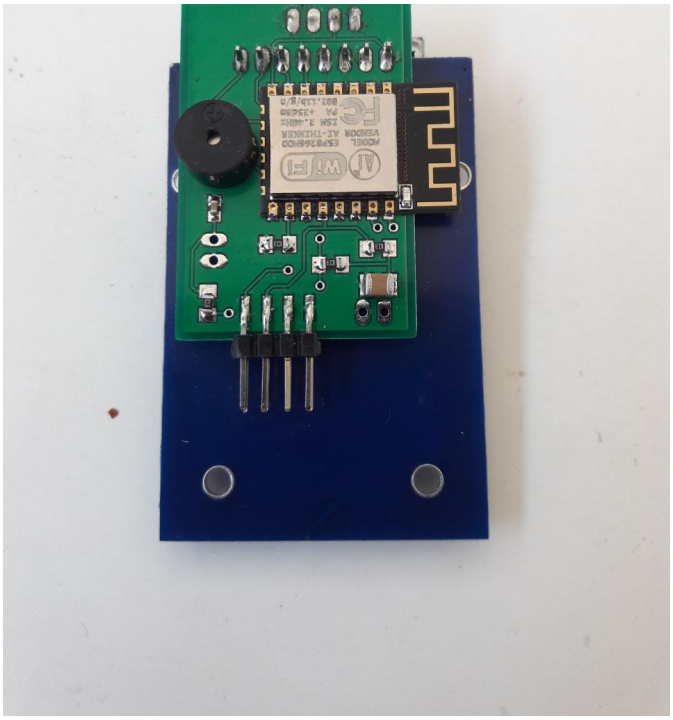
https://gitlab.com/sfz.aalen/projekte/fabaccess-case.

Anleitung

Hackwerk hat druckbare STL-Dateien und mehrere Fotos inklusive Zusammenbaureihenfolge als animierte GIF-Datei nutzerfreundlich abgebildet.

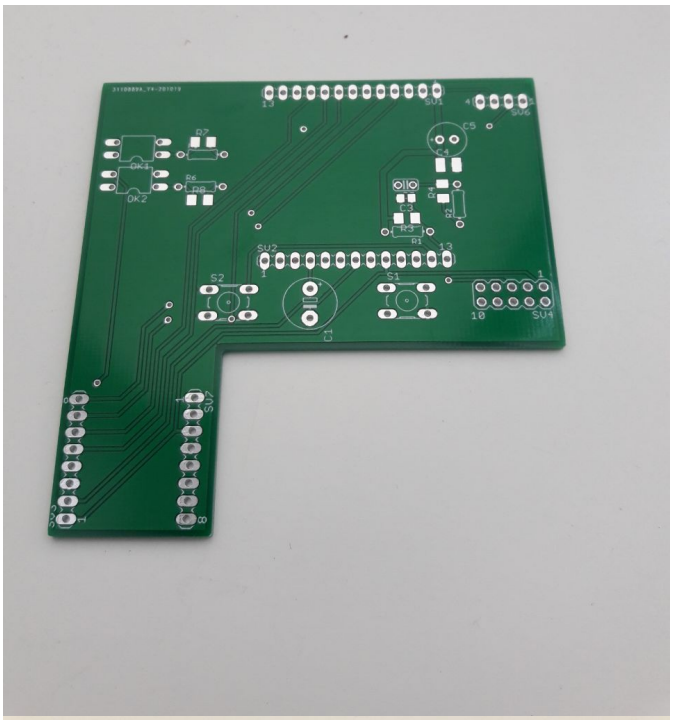
FabReader v1 (abgekündigt)


RFID-RC522



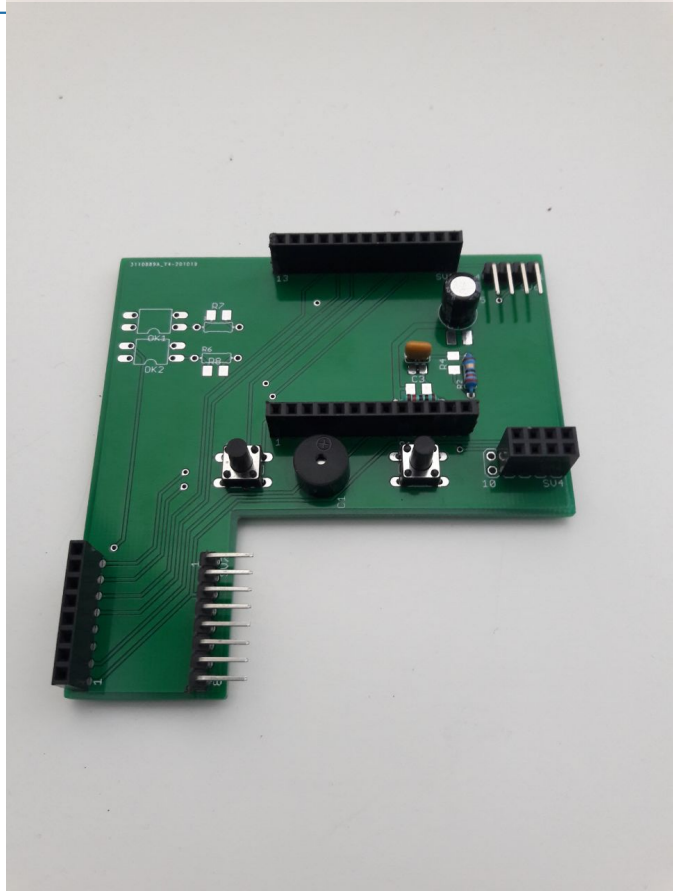

HanRun
HR911105A
15/10


connecting...
V0.03

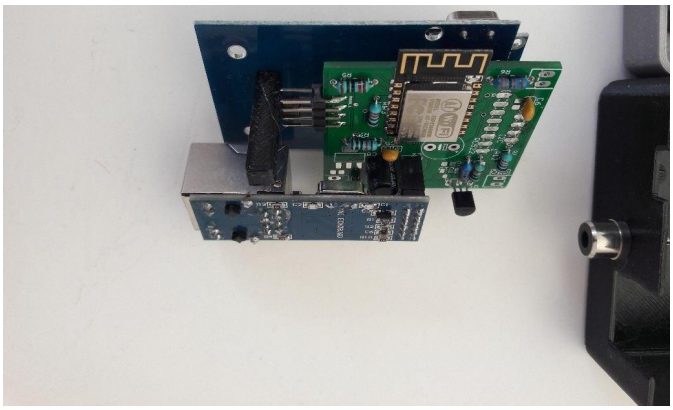





ID 003
LAN
connecting...
RFID-RC522
HNI ECN28J60










LAN
connecting...


ID 003
LAN
connecting...


ID 003
LAN
connecting...

Was ist das?

Das Projekt beschreibt ein RFID Lesegerät welches über MQTT in einer IT-Infrastruktur für offene Werkstätten eingebunden werden kann, z. B. für die Freischaltung von Ressourcen.

Mit einer RFID-Karte sich eine Person an einem Lesegerät in einem MQTT-Netzwerk identifizieren. In Abhängigkeit der damit verbundenen Anwendungen können hiermit Ressourcen freigeschaltet, Türen / Fächer aufgeschlossen oder Programme angesteuert werden. Dabei stellt das Lesegerät nur die Kommunikation mit der RFID-Karte und mit dem Anwender (über ein Display) zur Verfügung. Aktuatoren und Netzschalter sind nicht vorgesehen.

Features

- Ethernet RJ45 Buchse
- e-Ink Display
- SD-Kartenslot
- 2 Outputs für Optokoppler -> wurde nie zu Ende entwickelt
- Serielle Schnittstelle
- I2C (2-Wire) Schnittstelle
- WiFi Support
- Spannungsversorgung über Ethernet (PoE)
  - Wir hatten mal überlegt passives POE über einen einfachen Splitter und Injektion vorzusehen. Das wäre dann aber außerhalb des Gehäuses gewesen (bzw. hätte man das Gehäuse vergrößern müssen). Das verwendete LAN-Modul ließ es leider nicht zu direkt an den Pins zu gehen, die für passives POE verwendet werden. Sollte man den Lan-Anschluß nochmal diskret auf der Platine nachbauen, kann man da eine andere Buchse verwenden die die Pins rausführt.
- Unterstützung für MIFARE Classic, Ultralight C und DESFire (nur OTA)

Stückliste

- ESP32
- RC522 Modul - macht erhebliche Probleme mit DESFire Karten und die Platine darunter schluckt viel Sendeleistung

Sources (inkl. PDF-Aufbauanleitung)

https://gitlab.com/fabinfra/fabhardware/fabreader

Hinweise zum Status des FabReader v1

> Der FabReader ist nie über den Draft-Status hinaus gekommen und hat keine relevante praktische Anwendung in Werkstätten erfahren. Selbst Joris hat trotz produktiven FabAccess die FabReader nie wirklich verwendet. Hackwerk e.V. aus Aalen hat diese Reader widerrum verwendet, jedoch mit einer eigenen Firmware.

FabReader v2 (abgekündigt)

Wichtige Hinweise

Der FabReader v2 war ein lange Zeit nicht öffentlicher Entwurf eines FabReader v1 Nachfolgers und gilt als abgekündigtes Projekt. Dieser Entwurf ist jedoch ebenso nicht über das Entwurfsstadium gekommen. Das Projekt wurde in GitLab am 11.02.2025 öffentlich gesetzt, **archiviert** und entsprechend kommentiert (

https://gitlab.com/fabinfra/fabhardware/FabReader2)

> Aktuell (seit Ende 2024) wird durch die Community aktiv am FabReader v3 entwickelt.

FabReader v3

Intro

Der FabReader v3 ist ein akiv entwickeltes Projekt der FabInfra Community. Es ist eine lokale in einem Werkstattbereich oder einer Maschine installiertes Terminal.

Interesse am Projekt mit zu wirken? Dann komm in unseren Chat. Es gibt eine spezielle FabReader-Gruppe: https://matrix.to/#/#fabreader:vow.chat

Icon

Der offizielle Icon des FabReader Projektes:



Sources

Die Projektdateien finden sich unter https://gitlab.com/fabinfra/fabhardware/FabReader3

Vorgängermodelle

Die Vorgängermodelle von FabReader findest du hier:

- FabReader v1 (abgekündigt)
- FabReader v2 (abgekündigt)

Projektstatus

> Der FabReader v3 ist aktuell nicht fertig, aber in Arbeit.

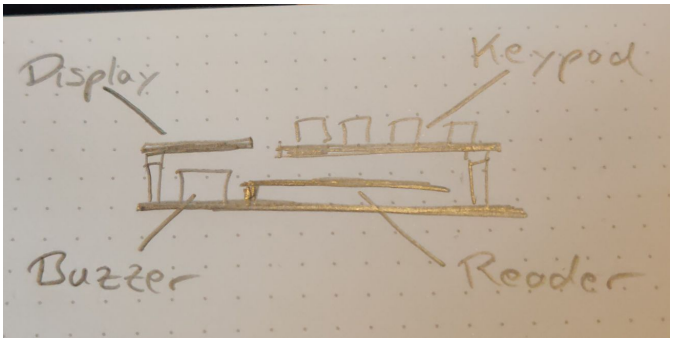
Anbindung an FabAccess per FabFire Tools

Die FabReader werden als Adapter an den Diflouroborane Server angebunden. Eine Anleitung findet sich unter FabFire Tools. Benutzt werden die Reader mit entsprechendne Karten.

Notizen

- Platine wird bei JLCPCB hergestellt und bestückt
- ist der aktuelle Prototyp, der in Entwicklung ist
- Protokoll: https://docs.fab-access.org/books/meeting-notes-pads/page/23102024-fabaccess-pcb-v2
- ToDos: https://todo.makerspace-gt.de/share/jyF9pO0uo5LS0RrSNijVyGdag9PHmQAQsnp8zBVi/auth?view=29
- Ziel: fertiges FabReader-Teileset bei AZDelivery (igami)

Skizze



Stückliste (BOM)

- ESP32-WROOM-32 (4,19 €)

  https://de.aliexpress.com/item/1005007816637789.html
- PN532 v3 RFID/NFC Reader (1,49 €)

  https://de.aliexpress.com/item/1005005633496559.html
- SSD1306 0,96" OLED (1,09 €)

  https://de.aliexpress.com/item/1005006425021544.html
- Buzzer (0,04 €)

  https://de.aliexpress.com/item/4000203107423.html
- LAN8720 (1,59 €)

  https://de.aliexpress.com/item/1005007821900741.html
- POE Spliiter USB C (2,79 €)

  https://de.aliexpress.com/item/1005006017042745.html
- 4x3 Matrix Keypad (1,25 €)

  https://de.aliexpress.com/item/1005006498970326.html

Tag Reader Modul: PN532 v3

https://de.aliexpress.com/item/1005005633496559.html

```
20:01:40    [D]    [pn532:367]    Mifare ultralight
20:01:40    [W]    [pn532.mifare_ultralight:020]    Not NDEF formatted
20:01:40    [D]    [pn532:200]    Found new tag '04-4F-58-6A-E5-18-90'
20:01:40    [D]    [pn532:379]    Waiting to read next tag
```

ACS ACR122U - USB NFC Reader



http://www.acs.com.hk/en/products/3/acr122u-usb-nfc-reader

OpenUnitState

https://github.com/homeofmaking/OpenUnitState