Grip Firmware

1.0

Purpose

The purpose of this device is to allow students in a school to register their attendance to a class. They can do this by using their smartphone or by using a passive NFC tag. This interaction is then sent to a central server, where the attendance is recorded and stored.

Hardware

Processor: ESP32 wroom 32U

Nfc: pn532

Storage: micro Sd card

Firmware

Components:

- pn532: Responsible for controlling the pn532 module.
- Rtc: Responsible for setting and reading the hardware real time clock.
- Sd: Responsible for data storage on a micro sd card.
- Wifi: Responsible for connecting to WIFI AP and communicating with the Grip server and time servers.
- Nfc: Responsible for high level NFC communication. (Emulation, Scan)
- IO: Low level GPIO handling.
- HMAC: Responsible for token signing.
- Config: Responsible for storing and reading configuration.

Communication

The device has two main way of communication: NFC and WIFI.

- **NFC**
  The NFC module used is the pn532 module. (https://www.nxp.com/docs/en/nxp/data-sheets/PN532_C1.pdf)
  This way of communication has two main purposes: Communicate with smartphones and read passive tags.
- **WIFI**
  Wifi communication is used to notify the server, whenever an attendance was registered with a passive tag, when the device needs a secret key for token signing, or when the internal clock needs to be adjusted to the current time.

Opeation

Startup

On startup the controller initializes the FREERTOS tasks in the following order:

1. Sd task
2. Config task
3. Wifi task
4. RTC task
5. NFC task

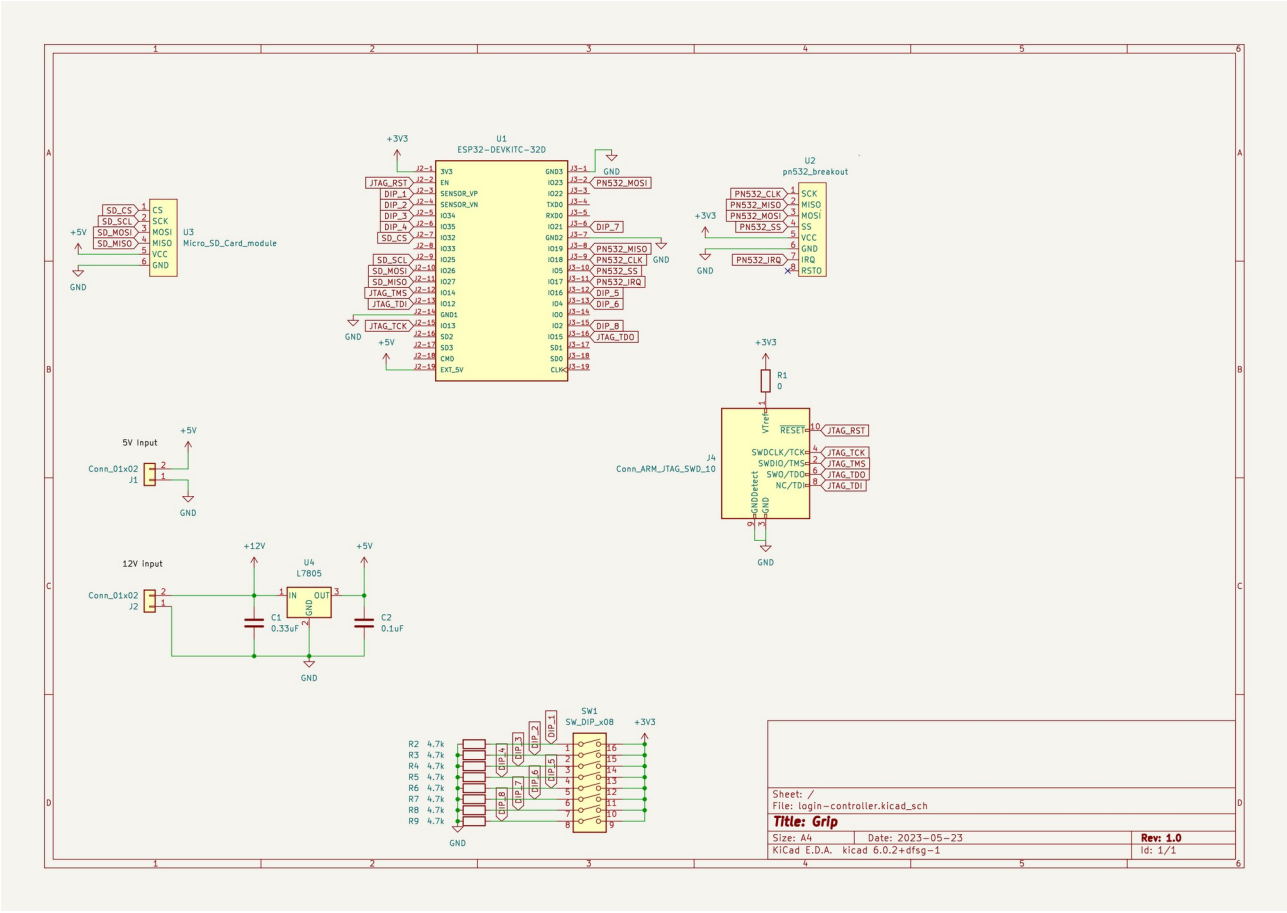
Operation

After startup the following events can trigger different to normal behaviour:

1. While emulating an nfc tag, a scanner (smartphone) activates and reads the tag. This causes the controller to write a response to the scanner, that contains information about the scan in the folowing format: <station id>_<unix time>_<salt>
   In this configuration the <station id> stands for the identifyer of the station set by the DIP switch, the <unix time> stands for the time of the scan in unix format, and <salt> stands for a random generated integer, for token security purposes. The message is accompanied by a token that is the message signed by the stations secret key. These two records are transmitted to the scanning device, and these can be used to verify a scans validity.
2. Periodically the RTC module need to be synced to the real world time.
3. When the device is in scanning mode and a passive tag is present within scanning range the device should initiate communication and read the serial number of the tag. This information than will be sent to the server.

PCB

Schematic:



Wiring