Dokumentation zum Projekt des fünften Semesters

LoRaWAN

Ubiquitious Computing am Beispiel eines Tablettenautomaten

|  |  |  |
| --- | --- | --- |
| Projektbeteiligte: | Meike Berger  Julia Gnann  Alexej Mendler | Xxxxxx  6085431  7226480 |
| Projektbetreuer: | Prof. Dipl.-Inf. Till Hänisch  Heiko Hutschenreiter |  |
| Semester:  Studiengang:  Fakultät:  Hochschule: | WWI2016/5  Wirtschaftsinformatik  Wirtschaft  DHBW Heidenheim |  |

# Inhaltsverzeichnis

# Projektdefinition

Mit dem vorliegenden Dokument wird das Projekt „Ubiquitious Computing“ dokumentiert. Im Mittelpunkt dieses Projekts steht die Planung, Entwicklung und Dokumentation eines Systems zur Informationsverarbeitung mit einem Microcontroller. Fünf binäre Daten werden von einem Legacy System an einen Microcontroller geschickt, gespeichert und über LoRaWAN ans TTN geschickt. Dort können die Daten von einem Webserver abgerufen und dargestellt werden.

Dazu wird zuerst eine Anforderungsanalyse durchgeführt, um die Anwendungsfälle und die technischen Anforderungen zu ermitteln. Dadurch kann eine Technologieauswahl getroffen werden und die Fehlerbehandlung geplant werden. Daraufhin folgt eine Systembeschreibung in Form eines Grobkonzepts. …

# Anforderungsanalyse

Funktionale Anforderungen/Anwendungsfälle:

1. Gehäuse geöffnet, wenn TA befüllt wird  
   🡪 wird simuliert durch Kippschalter auf 1
2. Ausgabekorb belegt (Lichtschranke)  
   🡪 wird simuliert durch Kippschalter auf 2
3. Summer, wenn Einnahmetermin ansteht  
   🡪 wird simuliert durch Taster 3
4. Präsenztaste, wird für Entnahme gedrückt nach Summer oder, um nächsten Termin zu erfahren  
   🡪 wird simuliert durch Taster 4 (bei Simulation NUR Entnahmesituation, NICHT Abfrage)
5. LED der Präsenztaste leuchtet, um auf Termin aufmerksam zu machen oder bei Terminabfrage

Technische Anforderungen:

Legacy System

Echtzeitfähigkeit

# Systemauswahl und Grobkonzept

## Grobkonzept



Der Tablettenautomat hat 5 I/Os, die an den LoPy4 via binärer Schnittstelle gesendet werden können. Die 5 I/Os sind:

* Gabellichtschranke
* Summer
* Präsenztaste
* LED Präsenztaste
* Gehäuse geöffnet

Die empfangenen Signale werden auf dem LoPy4 gespeichert und erhalten bei Eingang einen TimeStamp. Da die Uhrzeit des Tablettenautomaten nicht als binärer I/O abrufbar ist, wird der TimeStamp vom LoPy4 gesetzt. Dazu benötigt dieser ein sogenanntes RTC-Modul, eine Real Time Clock.

Die auf dem LoPy4 gesammelten Daten werden einmal täglich via LoRaWAN an das sogenannte TTN – The Things Network – geschickt. Der Webserver ruft die Daten vom TTN ab. Diese werden in einer Datenbank gespeichert und können vom Benutzer über eine GUI abgefragt werden.

Datenbank: noch unklar. SQL oder Dokumentenbasiert mit Docker.

GUI: einfache http-basierte Darstellung oder Wordpress etc.

## Schnittstellenbeschreibung

Nachfolgend werden die Schnittstellen des Tablettenautomaten definiert.

## Schaltplan

# Fehlerbehandlung

## Übertragung Fehlerbehandlung

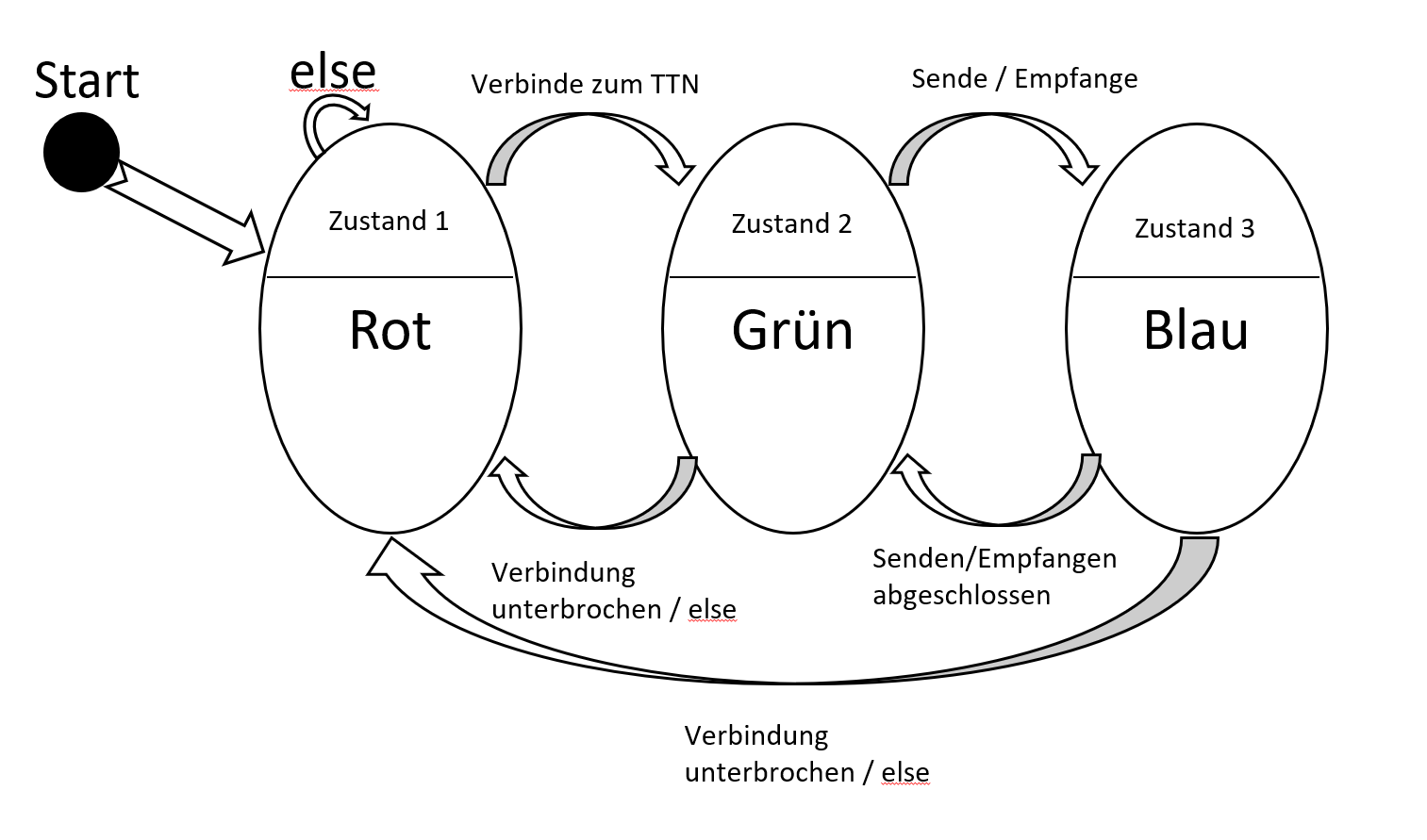
Es gibt einige Fehler, die während dem Übertragen der Daten passieren können. Um diese Fehler möglichst abfangen zu können, muss eine Fehlerbehandlung stattfinden.

Ein Problem kann sein, dass das Gerät nicht erreichbar ist und somit keine Daten an das TTN schicken kann oder die Datenübertragung fehlerhaft war.

Da der LoPy Daten speichern kann, soll bei der Fehlerbehandlung der Timestamp und der Status eine gewisse Zeit (ca. 7 Tage) gespeichert werden. Bei der nächsten Übertragung werden die im Speicher erfassten Daten über das TTN auf den Server weitergeleitet. Dieser stellt wiederum eine SQL-Abfrage an die Datenbank und gleicht ab, ob der Timestamp schon erfasst ist und gegebenenfalls übereinstimmt.

Bei fehlerhaften Daten soll bei einer Abfrage der Datensatz eingefügt werden. Ansonsten wird eine Rückmeldung an den LoPy geschickt, dass der Datensatz schon vorhanden ist. Danach soll der Datensatz im LoPy gelöscht werden.

Um Verbindungsprobleme auf dem LoPy direkt zu erkennen, wird die auf dem LoPy vorhandene LED in folgende Zustände unterteilt:



Rot: LoPy hat keine Verbindung zum TTN

Grün: LoPy hat eine bestehende Verbindung zum TTN

Blau: LoPy schickt oder empfängt Daten vom TTN

🡪 Statusmeldung, wann letzte Verbindung auf Server

🡪 Speichern auf LoPy, im Fall, dass kein Empfang. Später: Validierung, dass auf Server angekommen.

## Technische Fehlerbehandlung

* Technische Fehlerbehandlung: Tasten entprellen:

🡪 nach 1. Interrupt Timer stellen (20ms)

* Schnittstellen noch nicht definiert, ob high oder low

🡪 Inverterfunktion

## Logische Fehlerbehandlung

* Entnahmezeit berechnen:

Verwechslung zwischen Entnahme und Terminabfrage (negative Entnahmezeit ausschließen)

🡪 LED und Präsenztaste

* Entnahme von Becher dauert über eine halbe Stunde: Deadlock-Situation in Tablettenautomat: Folgetermine können nicht wahrgenommen werden.

🡪 Abfangen, aber wie? Z.B. indem auf Webserver angezeigt wird?

🡪 Versäumte Einnahmetermine darstellen!

Anwendungsbeschreibung

Lösungsbeschreibung

Umsetzungsbeschreibung