

Weekly Reports

Luftqualität in Innenräumen - Gruppe 1

17. Januar 2022

Name	Matrikel Nr.	Arbeitsaufwand (h)
Friedrich Just	1326699	
Stipe Knez	1269206	19,00
Lucas Merkert	1326709	20,00
Achim Glaesmann	1309221	17,50
Max-Rene Konieczka	1211092	18,00
Can Cihan Nazlier	1179244	14,00

Tabelle 1: Arbeitsaufwand dieser Woche

1 Überblick

1.1 Friedrich Just

1.2 Stipe Knez

Vergangene Woche wurde sich dazu entschieden, das Format, in dem die Daten der ZigBee Module an das Backend geschickt werden, zu überarbeiten. Anstatt die Daten in Form von JSON Strings an das Backend zu schicken, sollten diese nun in Form von regulären Strings, bei denen die Werte mit einem Semikolon getrennt werden, geschickt werden. Diese Funktionalität konnten wir im Verlauf der Woche mit Hilfe eines entsprechenden Parsers erfolgreich im Backend implementieren. Als nächstes wurde begonnen sich dem Übertragen der Daten vom Back- in das Frontend und der Erstellung eines Dashboards mit Hilfe der D3.js Library zuzuwenden.

1.3 Lucas Merkert

Diese Woche wurde der CCS811 Sensor zum laufen gebracht. Hierbei haben wir uns mithilfe der Informationen von Dienstag einen geeigneten Zustandsautomaten wie in Abbildung 1 überlegt und diesen dann umgesetzt. Hierbei wurde der Sensor über die Adresse 0x5A angesprochen angesprochen. Im ersten Zustand wechselt der Sensor in den BOOT Modus damit ein Software-Reset durchgeführt

werden kann und die Hardware ID überprüft werden kann. Danach wechselt der Sensor wieder in den APP Modus. In das MEAS_MODE Register wird dann der Befehl geschrieben, dass der Sensor jede Sekunde misst. Zuletzt wird noch im Status Register überprüft ob Daten bereit liegen und diese werden dann ausgelesen. Dies geschieht in einem Zyklus von einer Sekunde.

1.4 Achim Glaesmann

Diese Woche wurde dazu verwendet die von den Verschiedenen Teammitgliedern übernommenen Aufgaben bezüglich der Auslesung des Sensors und der Übertragung der Daten über das Zigbee Protokoll zusammenzuführen. Zunächst wurde sich persönlich getroffen, um in der Gruppe Probleme beim Auslesen des CCS811 und dem Aufbau des Funknetzes zu lösen. Nach testen des Sensors am Arduino und dem Übertragen der Programmlogik der anderen Sensoren auf den AVR-Chip, war es trotz mehrfachen Prüfen des Codes nicht möglich eine I2C-Kommunikation zum CCS aufzubauen. Selbst nach ausführlichem Testen und mehrmaligen umschreiben einzelner Softwarekomponenten, konnte das Problem nicht behoben werden. Es stellte sich heraus, dass es kein softwareseitiges Problem war. Nach austauschen des Funkmoduls war es möglich eine Kommunikation zwischen Sensor und Mikrokontroller zu etablieren. Der Mikrokontroller wurde markiert und wird nochmal überprüft. Nachdem sowohl das Auslesen des CCS811, als auch Aufbau des Funknetzwerkes realisiert wurden, war es nötig die einzelnen Softwarelösungen für die Sensoren und das Netzwerk zu kombinieren. Dies wurde erfolgreich erledigt. Als nächste Aufgabe ist es nun nötig, die Übermittelten Daten an das Backend der Desktop-Applikation zu übermitteln.

1.5 Max-Rene Konieczka

Im Verlauf der letzten Woche wurde daran gearbeitet, Strings über Atmel Studio zu versenden und diese anschließend in unserer Datenbank anzeigen zu lassen. Nachdem dies gelungen ist, wurde angefangen sich mit dem Dashboard auseinanderzusetzen. Es wurde überlegt, die D3.js Library von Javascript [1] zu verwenden. Mit D3.js ist es möglich einfache Formen bis hin zu detaillierten Diagrammen zu erstellen. Die Daten der Sensoren sollen nun mithilfe der Axis API von D3 in einem Koordinatensystem dargestellt werden.

1.6 Can Cihan Nazlier

Durchgehen der Dokumentation von d3.js. Websockets vorbereiten, sodass server side events gesendet werden können. Warum Websockets? Antwort: das frontend und das backend laufen als zwei separate anwendungen und müssen sich untereinander austauschen. Das geschieht eigentlich über http. Das Problem jedoch ist, für http muss der Client dem Server request schicken um Daten zu empfangen. Da die Sensordaten im Backend ankommen und wenn die Kommunikation nur mittels http geschieht, so müsste der client alle 10 minuten eine Anfrage an den Server senden und es könnte durchaus dazu kommen, dass wir einzelne

Messungen verpassen. Deshalb ist es wichtig, dass der server dem Client Daten sendet sobald welche da sind. Dazu wurde nun die Grundlage gelegt.

Literatur

- [1] *D3 Data-Driven Documents*. URL: <https://d3js.org/>.

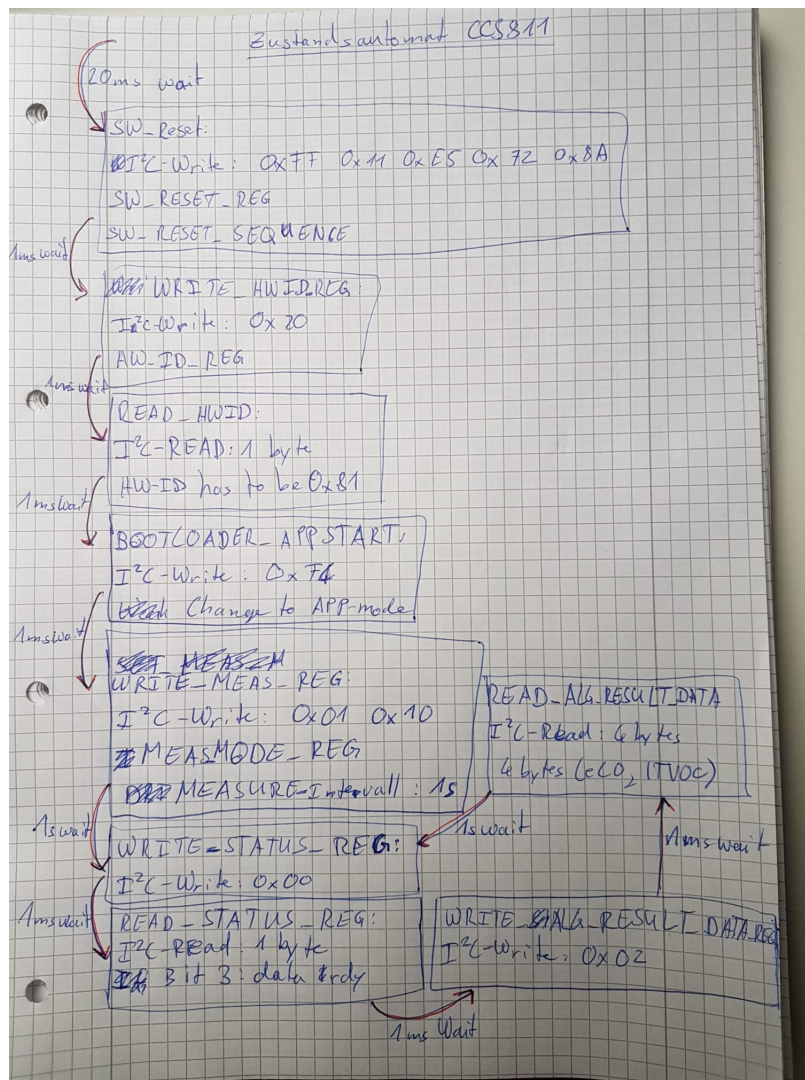


Abbildung 1: Zustandsautomat CCS811