

Pflichtenheft

Bluetooth-Mesh Plattform für IoT Anwendungen

PROJEKT 5 - ANKLIN, BOBST, HORATH
19. Februar 2020

Fachcoach:

Matthias Meier
Manuel Di Cerbo

Team:

Raffael Anklin
Robin Bobst
Cyrill Horath

Studiengang:

Elektro- und Informationstechnik

Semester:

Herbstsemester 2019

Inhaltsverzeichnis

1 Übersicht	1
1.1 Ausgangslage	1
1.2 Projektziele	2
1.3 Lieferobjekte	4
2 Lösungskonzept	5
3 Konzeptschema	5
3.1 UPN	5
3.2 GIS	5
3.3 HMI	5
4 Hardware	6
4.1 Bluetooth Mesh Node (BMN)	6
4.2 Gateway Interface System (GIS)	6
4.3 Energy Harvesting System (EHS)	7
4.4 Power Storage System (PSS)	8
5 Software	9
5.1 Software BMN	9
5.2 Software GIS	9
5.3 Human Machine Interface (HMI)	9
5.4 Security	9
5.5 Open Source Projekt	9
6 Projektvereinbarung	11
Literatur	12
A Konzeptschema	13
B Terminplanung	14
C Risikoanalyse	15

1 Übersicht

In diesem Kapitel soll eine Übersicht über den Inhalt des Projekts 5 des Studiengangs Elektro- und Informationstechnik gegeben werden. Dabei soll auch aufgezeigt werden, welche Ziele erreicht werden sollen und welche Lieferobjekte erstellt werden müssen.

1.1 Ausgangslage

Die Bluetooth Technik wurde im Jahr 1998 von der "Bluetooth Special Interest Group"(SIG) als Industriestandard für Datenübertragung herausgebracht. Ursprünglich wurde das Funkverfahren jedoch von Jaap Hartsen und Sven Mattisson für die Firma Ericsson entwickelt. Der Hauptzweck dieser Methode zur Datenübermittlung war das Ersetzen von Kabelverbindungen von Mobiltelefonen, Peripheriegeräten oder Computern. Der Name Bluetooth oder auf Deutsch Blauzahn kommt vom dänischen König Harald Blauzahn. Diesem König gelang es die verfeindeten Länder Dänemark und Norwegen dank seiner Kommunikationsfreudigkeit zu vereinen. Da die skandinavischen Firmen Nokia und Ericsson viel Aufwand in die Bluetooth Technologie gesteckt haben, wurde dieser Name sowie die Runen H (Harald) und B (Blauzahn) für das Logo übernommen.[1] Seit dem Start von Bluetooth gab es eine Vielzahl von Versionen, die von mehreren Firmen ständig weiterentwickelt werden. Im Dezember 2009 wurde von der SIG die Version 4.0 Smart vorgestellt. Mit dieser Version von Bluetooth war es möglich kleine und sparsame Geräte wie z.B. smarte Uhren, Brillen oder sogar Ringe herzustellen.[2] Ab dem Jahre 2017 ist es möglich Bluetooth Komponenten in einem Mesh-Netzwerk zu konfigurieren.[3] Dieses Netzwerk basiert auf einem "many-to-many pairing system" d.h. jeder Teilnehmer ist mit den anderen Teilnehmern verbunden. Dieses dezentralisierte System hat den Vorteil, dass es kein Master Element benötigt. Fällt ein Teilnehmer aus besteht das Netzwerk trotzdem weiter.[4] Genau hier soll das Projekt 5 ansetzen. Da die Programmierung eines Mesh-Netzwerkes sehr kompliziert ist, wird dafür eine "Open Source Software" geschrieben, die es ermöglicht ein Netzwerk vereinfacht aufzubauen und zu konfigurieren.

1.2 Projektziele

In den beiden Tabellen 1.1 und 1.2 sind die Pflicht- resp. Wunschziele für dieses Projekt festgehalten.

Pflichtziele		
Nr.	Ziel	Beschrieb
P1	Bluetooth-Mesh-Netzwerk	Eine variable Anzahl an BLE-Nodes bauen ein Mesh-Netzwerk auf um darin Datenaustausch zu ermöglichen.
P2	UPN	Der Universal-Peripheral-Node kann je nach Einsatz als Sensor oder Aktor konfiguriert und bestückt werden.
P3	Low Power	Die UPN sind bezüglich Hardware und Software energiesparend konzipiert um sie autonom betreiben zu können.
P4	Security	Das Mesh-Netzwerk ist gegen unerlaubten Zugriff und sonstigen Angriffen geschützt.
P5	Netzunabhängig	Durch Versorgung mittels Batterie und Energy-Harvesting können die UPN komplett netzunabhängig betrieben werden.
P6	Energy-Harvesting	Für die Versorgung der UPN werden verschiedene Varianten für das Energy-Harvesting entwickelt. Das Ergebnis wird eine Variantenstudie sein.
P7	Gateway	Zur Konfiguration des Bluetooth-Mesh-Netzwerks steht ein Gateway basierend auf Standard Hardware (Raspberry-Pi + nRF52840 USB Dongle o.ä.) zur Verfügung.
P8	LAN/WLAN	Für die Integration in TCP/IP basierte Systeme bietet der Gateway eine entsprechende Schnittstelle.
P9	CLI	Mittels Command-Line-Interface kann das Mesh-Netzwerk verwaltet werden.

Tabelle 1.1: Pflichtziele

Wunschziele		
Nr.	Ziel	Beschrieb
W1	UPN Konfiguration via Mesh	Einstellungen des UPN können via Mesh Netzwerk angepasst werden und somit z.B. die Peripheriekonfiguration verändert werden.
W2	Firmwareupgrade via Mesh	Die Firmware der UPN wird via Mesh-Netzwerk aktualisiert.
W3	BLR und BLE	Bluetooth Long Range (BLR) und Bluetooth Low Energy (BLE) ergänzen das Bluetooth Mesh um die Reichweite zu vergrößern oder den Energieverbrauch nochmals zu vermindern.
W4	Dedizierte Hardware UPN	Das UPN ist als dedizierte Hardware realisiert und somit einsatzbereit.
W5	Datenschnittstelle	Mittels passender Datenschnittstelle auf dem Gateway können Fremdsysteme wie Apple Homekit, Google Home oder KNX angebunden werden.
W6	Datenschnittstelle ohne Zwischenspeicherung	Damit keine Daten auf dem Gateway zwischen gespeichert werden müssen können die Nodes mittels verbindungslosem Protokoll (MQTT, CoAP, usw.) direkt aus dem Mesh Netzwerk mit einem Fremdsystemen kommunizieren.
W7	HMI	Ein Human-Machine-Interface in Form einer Webapplikation unterstützt den User bei der Konfiguration des Mesh-Netzwerks und ermöglicht die Anbindung an Fremdsysteme.
W8	Dedizierte Gateway Hardware	Der Gateway ist auf einer dedizierten Hardware umgesetzt.
W9	Onboard Bluetooth	Da der Raspberry-Pi 4 bereits ein Bluetooth 5 Chip besitzt soll direkt dieser verwendet werden anstelle eines angeschlossenen Dongles.
W10	Mobiltelefon	Anstelle oder ergänzend zum Gateways kann ein Mobiltelefon ins Mesh-Netzwerk eingebunden werden um Konfigurationen vorzunehmen oder Daten aus zu lesen.
W11	GSM/LTE	Für Feldanwendungen besitzt der Gateway ein GSM/LTE Modul.
W12	Versuchsaufbau Energy-Harvesting	Erfolg versprechende Energy-Harvesting-Systeme werden in einem Versuchsaufbau auf deren Tauglichkeit weiter geprüft.

Tabelle 1.2: Wunschziele

1.3 Lieferobjekte

Zusätzlich zu den Projektzielen, folgen in diesem Kapitel die Lieferobjekte mit dem jeweiligen Datum. In der Tabelle 1.3 sind diese aufgelistet.

Nr.	Datum	Lieferobjekt
1	07.10.2019	Abgabe Pflichtenheft, 1. Version
2	14.10.2019	Abgabe Pflichtenheft, definitive Version
3	13.01.2020	Projektpräsentation
4	13.01.2020	Abgabe Fachbericht
5	13.01.2020	Abgabe Testaufbau Mesh-Netzwerk

Tabelle 1.3: Lieferobjekte

2 Lösungskonzept

Im Zentrum soll die Entwicklung einer *Bluetooth Mesh* Plattform stehen. Diese soll für ein weiterführendes Projekt einsetzbar sein (*Home Automation*, *Agricultur* oder *Industrie*).

3 Konzeptschema

In Abbildung A ist das Blockschaltbild ersichtlich, welches alle Teilsysteme und Einheiten darstellt. Es ist modular gegliedert und bietet eine Übersicht der Schnittstellen zwischen den einzelnen Modulen.

Das Konzeptschema besteht aus physikalisch getrennten Systemen, dem *Universal Peripheral Node* (UPN), dem *Gateway Interface System* (GIS), dem *Human Machine Interface* (HMI) und über das Internet verbundene Webserver, im Konzept als *Platform Bindings* bezeichnet.

3.1 UPN

Der *UPN* beinhaltet einen *Bluetooth Mesh Node* (BMN), welcher die Anbindung an das Mesh Netzwerk ermöglicht. Als Stromversorgung dient das *Power Storage System* (PSS), sowie das *Energy Harvesting System* (EHS). Diese Systeme verfügen über externe angebundene Energiequellen (z.B. Solarzellen). Zusätzlich werden die Aktoren und Sensoren des *UPN* über eine *Universal Peripheral Unit* (UPU) angesteuert bzw. eingelesen.

3.2 GIS

Als Zentrale Einheit des *GIS* dient das *Central Control System* (CCS). Zur Anbindung an das Mesh Netzwerk steht ebenfalls ein *BMN* zur Verfügung. Die Stromversorgung erfolgt über das lokale Stromnetz mithilfe eines Netzteils. Zusätzlich wird die Internetverbindung über das *Mobile Network Interface* (MNI) abgesichert.

3.3 HMI

Das *HMI* bildet die Benutzerschnittstelle über ein internetfähiges Gerät. Es kommuniziert mittels WLAN/LAN mit dem *GIS*. Über dieses Gerät kann der Status visualisiert, sowie die Konfigurationen durchgeführt werden.

4 Hardware

In diesem Kapitel wird die Hardware vom *Gateway Interface System*, dem *Universal Peripheral Node*, dem *Energy Harvesting System* und dem *Power Storage System* beschrieben.

4.1 Bluetooth Mesh Node (BMN)

Im Bluetooth Mesh Protokoll gibt es zwei verschiedene Geräte, ein *“unprovisioned device“* und einen *“node“*. Das *“unprovisioned device“* ist ein Teilnehmer, der für das Mesh Netzwerk unbekannt ist und deshalb keine Rechte besitzt. Wird dieses Gerät nun in das Netzwerk aufgenommen, so wird das *“unprovisioned device“* zu einem *“node“*. Dieses vorgehen nennt sich *“provisioning“*. [5] Die Hardware für den *“node“* besteht bei allen Geräten aus dem gleichen SoC. Der nRF52840 von Nordic Semiconductor eignet sich aus folgenden Gründen perfekt für diese Anwendung. Die *“nodes“* dürfen, um eine lange Laufzeit zu garantieren, sehr wenig elektrische Leistung beziehen. Der nRF52840 benötigt im Ruhemodus nur wenige $[\mu A]$. Ein weiterer Grund ist die sehr gute Dokumentation der Software von Nordic Semiconductor. Die gesamte Software ist im Infocenter erhältlich und frei zugänglich. Weitere Vorteile befinden sich in der Tabelle 4.1: [6]

Vorteile des nRF52840

Bluetooth 5	-95 dBm Sensitivität
Multiprotokoll (Thread, Zigbee, usw)	+8 dBm Ausgangsleistung
Geringer Stromverbrauch (wenige $[\mu A]$)	USB 2.0
12bit ADC	NFC
1 MB flash und 256kB RAM Speicher	ARM M4F Cortex

Tabelle 4.1: Vorteile des nRF52840

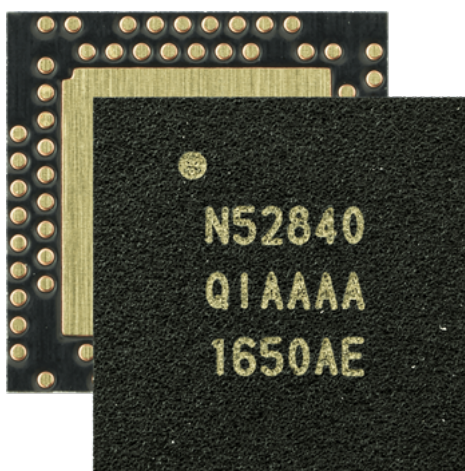


Abbildung 4.1: nRF52840 SoC [7]

4.2 Gateway Interface System (GIS)

Der Bluetooth-Mesh-Gateway soll die Plattform gegenüber Fremdsystemen öffnen und damit die Schnittstelle zu IOT Anwendungen (Internet of Things) bilden. Damit die Plattform einfach zu betreiben ist, wird beim Gateway in erster Linie auf den Einplatinen-Computer Raspberry-Pi

4 (siehe Abschnitt 4.2) gesetzt. Andere Einplatinen-Computer wären ebenfalls denkbar wobei die Raspberry-Pi-Plattform bereits sehr weit verbreitet ist und somit oft bereits verfügbar ist.

Der Raspberry-Pi 4 besitzt neben Ethernet und WLAN Schnittstellen sowie USB 3 Ports auch von Grund auf mit einem Bluetooth 5 Modul ausgestattet. So könnte er direkt ins Bluetooth-Mesh-Netzwerk integriert werden. Um jedoch die volle Integration zu erreichen, wird in erster Linie der oben erwähnte nRF52840 (siehe Abschnitt 4.1) in Form eines Dongle-Development-Boards 4.3 eingesetzt. Via serieller Schnittstelle wird dieser mit dem Raspberry-Pi 4 verbunden.

Als Ergänzung oder Alternative zum Gateway soll beispielsweise ein Mobiltelefon welches ein Bluetooth Modul besitzt ins Mesh Netzwerk integriert werden können. Darüber sollen dann Konfigurationen sowie die Ein- und Ausgabe von Parametern und Daten möglich sein.

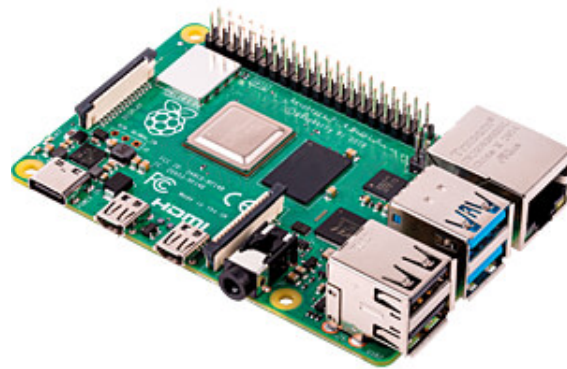


Abbildung 4.2: Raspberry-Pi 4 [8]

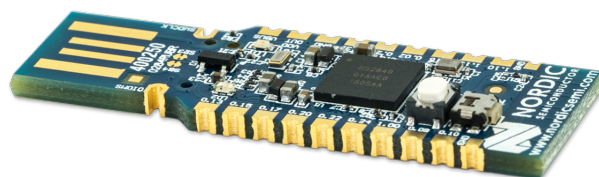


Abbildung 4.3: nRF52840 USB Dongle [9]

4.3 Energy Harvesting System (EHS)

Das *EHS* beinhaltet unterschiedlichen Methoden um Energie aus der Umgebung aufzufangen. Diese sind in der folgenden Tabelle mit den wichtigsten Kenndaten aufgefassen.

Durch Berechnungen, Simulationen und Testaufbauten soll gezeigt werden, welche Methode sich als genügend ertragreich für unser System erweist. Dafür muss der Energiebedarf des Systems gemessen werden. Die Messungen erfolgen unter verschiedenen Konfigurationen. Mithilfe der Ergebnisse soll es möglich sein, die optimale Konfiguration situationsbedingt zu wählen.

Source	Power density	Harvesting tech.	Advantages	Disadvantages
Solar	Indoor: 10 $\mu\text{W}/\text{cm}^2$ Outdoor: 10 mW/cm^2	Photovoltaic	High power density Mature	Not always available Required exposure to light (not implantable) Expensive
Vibration	Human: 4 $\mu\text{W}/\text{cm}^2$ Industrial: 100 $\mu\text{W}/\text{cm}^2$	Piezoelectric Electrostatic Electromagnetic	Implantable High efficiency	Not always available Material physical limitation
Thermal	Human: 30 $\mu\text{W}/\text{cm}^2$ Industrial: 1–10 mW/cm^2	Thermoelectric Pyroelectric	High power density Implantable	Not always available Excess heat
RF	GSM: 0.1 $\mu\text{W}/\text{cm}^2$ Wi-Fi: 1 mW/cm^2	Antenna	Always available Implantable	Low density Efficiency inversely proportional to distance

Abbildung 4.4: Tabelle Umgebungsenergie [10]

4.4 Power Storage System (PSS)

Das *PSS* ist für die Energiespeicherung zuständig. Untersucht wird die Speicherung der gewonnenen Energie aus dem *EHS* mithilfe Sekundärer Zellen (Akkus) oder mit *Supercaps*. Der Einsatz eines Tiefendladungsschutzes ist bei verwenden der sekundären Zellen angedacht. Zusätzlich soll die Versorgung aus primären Zellen untersucht werden. Mithilfe der Ergebnisse soll es möglich sein, die optimale Konfiguration situationsbedingt zu wählen.

5 Software

Das Kapitel Software befasst sich mit der Programmierung des Mikrocontroller *nRF52840* sowie des *GIS*.

5.1 Software BMN

Die Software auf dem *BMN* (siehe Abschnitt 4.1) wird mit Hilfe der folgenden *SDKs* von *Nordic Semiconductor* entwickelt:

- *nRF5 SDK*[11]
- *nRF5 SDK for Mesh*[12]
- *Zephyr*[13]

Diese *SDKs* enthalten eine sehr gut dokumentierte Bibliothek, die für den Quellcode des *BMN* benötigt werden.

5.2 Software GIS

Mit dem Ziel den Gateway Open Source und Open Hardware zu realisieren, soll eine einfache Linux-Distribution eingesetzt werden. Beim Raspberry-Pi 4 (siehe Abschnitt 4.2) eignet sich dafür besonders das Debian basierende Betriebssystem Raspbian da es eigens für die Raspberry-Pi Familie entwickelt wurde.

Basierend auf dieser Oberfläche können nun Anbindungen an Fremdsysteme wie *Apple Homekit* oder *KNX* mit den passenden Software Bausteinen realisiert werden. Welche Bausteine dies sein werden ist Teil der Umsetzung für eine spezifische Anwendung welche noch nicht festgelegt ist. Daher wird in erster Linie mit einfachen Sprachen wie beispielsweise *Node.js* oder *Python* eine Ein- und Ausgabe von Daten oder Befehlen umgesetzt.

5.3 Human Machine Interface (HMI)

Das *HMI* wird kein primäres Ziel sein. Angedacht wird ein einfaches *Webinterface* zur Konfiguration des *Mesh-Netzwerkes*.

5.4 Security

Der Bluetooth-Mesh-Standart von (*SIG*) stellt vier verschiedene Sicherheitspakete zur Verfügung.[14]

- Authentifizieren durch z.B. LED die auf Board drei Mal blinkt
- Zwei Level AES-CCM Verschlüsselung mit 128-bit Schlüssel
- Verbergen der Metadaten durch einen privaten Schlüssel
- Laufnummer zur Verhinderung von wiederholenden Nachrichten

5.5 Open Source Projekt

Die gesamte Software im Projekt 5 wird als “Open Source Software“ deklariert. Damit die Software global zur Verfügung steht, wird diese unter der “General Public License Version 3“ (GPLv3) lizenziert. Die GPL beinhaltet ein starkes “copyleft“, d.h. das Software-Projekt muss öffentlich und gebührenfrei zugänglich sein und der Quellcode muss jedem ausgehändigt werden, der danach fragt. Das starke “copyleft“ bringt aber den Vorteil, dass die Lizenz vom Projekt

nicht verändert werden kann. Wird die Software von jemandem weiterentwickelt muss dieser die GPL Lizenz weiterführen und kann kein “closed source“ Projekt daraus erstellen. Ein weiterer Vorteil im Hinblick auf die Bachelorarbeit ist, dass das “copyright“ einer “open source“ Lizenz beim Entwickler bleibt. Das bedeutet die Grundlage der Software kann weiterentwickelt werden z.B. als eine Software die Hausautomation steuert und somit als “closed source“ verwendbar ist.[15]

6 Projektvereinbarung

Projektcoach

Meier Matthias

Ort, Datum:

Di Cerbo Manuel

Ort, Datum:

Unterschrift:

Unterschrift:

Projektteam

Anklin Raffael

Ort, Datum:

Bobst Robin

Ort, Datum:

Horath Cyrill

Ort, Datum:

Unterschrift:

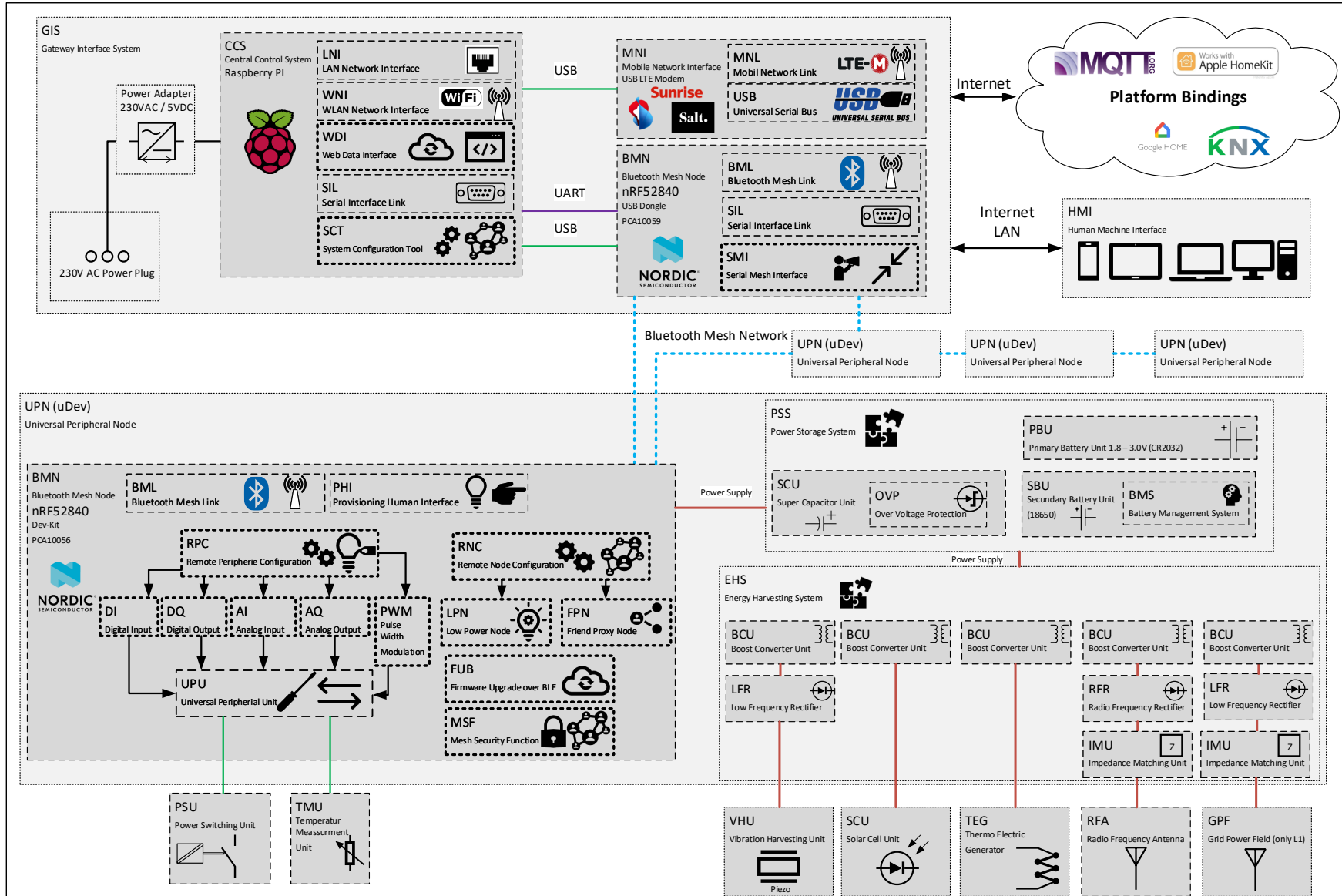
Unterschrift:

Unterschrift:

Literatur

- [1] K. Michna, *Entwicklungsgeschichte der Bluetooth-Technologie | Wissen*, de, Jan. 2019. Adresse: <https://www.bluetoothtest.de/wissen/geschichte/> (besucht am 30. Sep. 2019).
- [2] Bluetooth SIG, *Our History*, en-US, Bluetooth, Jan. 2019. Adresse: <https://www.bluetooth.com/about-us/our-history/> (besucht am 30. Sep. 2019).
- [3] M. Eckstein, *Neue Bluetooth-SIG-Gruppe für Mesh-Vernetzung im Smart Home*, de, Jan. 2019. Adresse: <https://www.elektronikpraxis.vogel.de/neue-bluetooth-sig-gruppe-fuer-mesh-vernetzung-im-smart-home-a-789161/> (besucht am 9. Okt. 2019).
- [4] M. Woolley, *An Intro to Bluetooth Mesh Part 1*, en-US, Juli 2017. Adresse: <https://www.bluetooth.com/blog/an-intro-to-bluetooth-mesh-part1/> (besucht am 30. Sep. 2019).
- [5] M. Afaneh, *The Ultimate Bluetooth Mesh Tutorial (Part 1)*, en-US, Sep. 2018. Adresse: <https://www.novelbits.io/bluetooth-mesh-tutorial-part-1/> (besucht am 30. Sep. 2019).
- [6] Nordic Semiconductor, *nRF52840 - Nordic Semiconductor*, en, Jan. 2019. Adresse: <https://www.nordicsemi.com/en/Products/Low%20power%20short-range%20wireless/nRF52840> (besucht am 2. Okt. 2019).
- [7] —, *nRF52840-QIAA.png (350×350)*, Jan. 2019. Adresse: <https://www.nordicsemi.com/-/media/Images/Products/SoC/SoCs-dobble-top/nRF52840-QIAA.png?h=350&la=en&mw=350&w=350&hash=5BEC8854006161B24DC902D49AC6D28BEEBE66A8> (besucht am 7. Okt. 2019).
- [8] R. E. G. Č. KG, *RASP PI 4 B 4GB - Raspberry Pi 4 B, 4x 1,5 GHz, 4 GB RAM, WLAN, BT*, de. Adresse: <https://www.reichelt.de/raspberry-pi-4-b-4x-1-5-ghz-4-gb-ram-wlan-bt-rasp-pi-4-b-4gb-p259920.html> (besucht am 5. Okt. 2019).
- [9] Nordic Semiconductor, *nRF52840-Dongle-promo.png (1159×500)*, Jan. 2019. Adresse: <https://www.nordicsemi.com/-/media/Images/Products/DevKits/nRF52840-Dongle-promo.png?h=500&la=en&mh=500&w=1159&hash=9DA47D0F2A8ABE0D96A002961ABE74B6D9246BA5> (besucht am 5. Okt. 2019).
- [10] L.-G. Tran, H.-K. Cha und W.-T. Park, „RF power harvesting: a review on designing methodologies and applications“, en, *Micro and Nano Systems Letters*, Jg. 5, Nr. 1, S. 14, Dez. 2017, ISSN: 2213-9621. DOI: 10.1186/s40486-017-0051-0. Adresse: <http://mns1-journal.springeropen.com/articles/10.1186/s40486-017-0051-0> (besucht am 7. Okt. 2019).
- [11] Nordic Semiconductor, *nRF5 SDK for Mesh*, en, Jan. 2019. Adresse: <https://www.nordicsemi.com/en/Software%20and%20Tools/Software/nRF5%20SDK%20for%20Mesh> (besucht am 5. Okt. 2019).
- [12] —, *nRF5 SDK*, en, Jan. 2019. Adresse: <https://www.nordicsemi.com/en/Software%20and%20Tools/Software/nRF5%20SDK> (besucht am 5. Okt. 2019).
- [13] Zephyr Project, *Zephyr*, en-US, Jan. 2019. Adresse: <https://www.zephyrproject.org/> (besucht am 9. Okt. 2019).
- [14] Nordic Semiconductor, *Nordic Semiconductor Infocenter - Security*, Juli 2019. Adresse: https://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.meshsdk.v3.2.0%2Fmd_doc_introduction_basic_concepts.html (besucht am 9. Okt. 2019).
- [15] T. Jaeger, A. Metzger, O. Koglin, T. Kreutzer, S. Labesius und D. Jansen, *Was ist die GPL? | ifrOSS*, de, Jan. 2018. Adresse: <https://www.ifross.org/was-gpl> (besucht am 5. Okt. 2019).

A Konzeptschema



Erstellt von Raffael Ankin	Datum 04.10.2019	Überarbeitet am Überarbeitet am	nw Fachhochschule Nordwestschweiz Hochschule für Technik	Projektbezeichnung: Bluetooth Mesh Plattform	Zeichenblatt-1 Bluetooth Mesh Plattform Konzeptzeichnung		Rev Dok. Nr.	Zeichenblatt 1 von 1
-------------------------------	---------------------	------------------------------------	--	---	---	--	-----------------	-------------------------

B Terminplanung

Hauptaufgabe	Aufgabe	Anfang	Ende	Wer	Review	30. Sep 19	7. Okt 19	14. Okt 19	21. Okt 19	28. Okt 19	4. Nov 19	11. Nov 19	18. Nov 19	25. Nov 19	2. Dez 19	9. Dez 19	16. Dez 19	23. Dez 19	30. Dez 19	6. Jan 20	13. Jan 20
Fachbericht	Strukturierung / Disposition	04.11.2019	18.11.2019	TBD																	
Fachbericht	Schreiben	18.11.2019	06.01.2020	Alle	Alle																
Fachbericht	Korrigieren / Gegenlesen	06.01.2020	13.01.2020	Alle	Alle																
Fachbericht	Drucken	13.01.2020	13.01.2020	CH																	
Fachbericht	Abgabe Fachbericht	13.01.2020	13.01.2020	Alle																	
Präsetation	Präsentation erstellen	06.01.2020	10.01.2020	Alle																	
Präsetation	Präsentation halten	13.01.2020	13.01.2020	Alle																	
Pflichtenheft	Pflichtenheft verfassen	30.09.2019	14.10.2019	Alle																	
Pflichtenheft	Abgabe Pflichtenheft	14.10.2019	14.10.2019	Alle																	
Vorstudie	Lösungsvarianten recherchieren	14.09.2019	21.10.2019	Alle																	
Vorstudie	Varianten Vergleichen in Tabelle und Entscheidung	07.10.2019	14.10.2019	Alle																	
Energie Harvsting	Berechnungen und Simulationen	07.10.2019	11.11.2019	RA																	
Energie Harvsting	Planen der Testaufbauten	21.10.2019	18.11.2019	RA																	
Energie Harvsting	Material Bestellungen für Testaufbauten	11.11.2019	11.11.2019	RA																	
Energie Harvsting	Testaufbauten fertigstellen	25.11.2019	30.11.2019	RA																	
Energie Harvsting	Messungen Durchführen und Vergleichen	25.11.2019	09.12.2019	RA																	
Energie Harvsting	Dokumentieren und evt. Optimierungen durchführen	09.12.2019	23.12.2019	RA																	
Bluetooth Firmware	IDE aufsetzen	30.09.2019	07.10.2019	BOB																	
Bluetooth Firmware	Mesh Netzwerktest mithilfe Nordic Examples	07.10.2019	21.10.2019	BOB																	
Bluetooth Firmware	Low Power node und Freind Node Energiebedarf messen	14.10.2019	21.10.2019	BOB/RA																	
Bluetooth Firmware	Firmware Remote Node Configuration Konzept erstellen	14.10.2019	28.10.2019	CH/BOB																	
Bluetooth Firmware	Firmware Remote Node Configuration programmieren	21.10.2019	25.11.2019	BOB																	
Gateway	Gateway Aufbauen und Basic Setup durchführen	30.09.2019	07.10.2019	CH																	
Gateway	Mesh Interface mithilfe Example Nordic ausprobieren	07.10.2019	25.10.2019	CH																	
Gateway	Webserver (Node Red) aufsetzen	07.10.2019	25.10.2019	CH																	
Gateway	Programm Remote Configuration Konzept erstellen	14.10.2019	28.10.2019	CH/BOB																	
Gateway	Programm Remote Configuration programmieren	21.10.2019	25.11.2019	CH																	
Hardware uDev	Schema und PCB Layout erstellen	25.11.2019	09.12.2019	tbd																	
Hardware uDev	Bestellung Material und PCB	12.12.2019	23.12.2019	tbd																	
Hardware uDev	Fertigen und Funktionstest	30.12.2019	06.01.2020	tbd																	
Platform Bindings	Anbindungs Konzepte erstellen	25.11.2019	09.12.2019	tbd																	
Platform Bindings	Tests durchführen und Dokumentieren	09.12.2019	06.01.2020	tbd																	
Validierung	Gesamttest Aufbauen (Gateway+Nodes+Energy Harv.)	02.12.2019	16.12.2019	Alle																	
Validierung	Gesamttest durchführen (Gateway+Nodes+Energy Harv.)	16.12.2019	23.12.2019	Alle																	
Validierung	Resultate dokumentieren	23.12.2019	06.01.2020	tbd																	
Validierung	Gesamtsystem für Presentation aufbauen und abgeben	06.01.2020	13.01.2020	Alle																	

C Risikoanalyse

Ereignis				Risiko ohne Massnahme			Prävention	Risiko mit Massnahme			Verantwortlich	Indikator
id	Risiko	Ursachen	Konsequenzen	si	pi	pi		si	pi	pi		
A	Teammitglied fällt kurzfristig aus	Unvorhergesehener Termin, leichte Krankheit, leichter Unfall	Weniger Personalressourcen, kleiner Mehraufwand	3	2	6	Reservezeit einplanen, Transparenter Informationsfluss im Team	1	2	2	CH	Abwesenheit
B	Teammitglied fällt längerfristig aus	Militärdienst, schwere Krankheit, Studienabbruch, schwerer Unfall	Grössere Umplanung, Neuverteilung der Arbeiten	3	2	6	Strukturierte Datenablage, guter Kommunikationsfluss	1	2	2		Abwesenheit
C	Datenverlust oder Zugriffsprobleme	Löschung der Projektdaten, Unzugänglichkeit von Onedrive, keine Internetverbindung	Zugriff auf Daten nicht möglich, Sämtliche Projektdaten nicht mehr vorhanden	2	1	2	Regelmässige Backups, Dokumente zusätzlich lokal abspeichern	1	1	1		Arbeiten auf dem Stand des letzten Backups
D	Software kann nicht mehr ausgeführt werden	Datenverlust, Softwareupdate	Schlimmstenfalls Verlust der gesamten Arbeit, vorübergehende Arbeitspause bis Update komplett	2	3	6	Fertige Softwareteile werden zusätzlich im Onedrive gespeichert (Revisionsverwaltung) / Github	2	1	2		Fehlermeldung
Ereignis	Softwarekonzept nicht ausführbar	Mangelnde Vorkenntnisse, schlechte Planung	Überdenken der Arbeit, Verzug der Arbeiten	2	2	4	Mit Software-Fachcoach besprechen	1	1	1		Nicht funktionierendes Skript
F	Softwareprojekt von Node Gerät und provisioner Gerät nicht verknüpfbar	Schnittstelle wurde nicht korrekt eingehalten	Verzögerung der Arbeit, Mehraufwand	2	2	4	Kommunikation zwischen Softwareteam	1	1	1		Softwareteam können Vorhaben nicht weiterführen
G	Zu komplizierter Sachverhalt	Inhalt kann nicht umgesetzt werden	Stillstand der Arbeit, Projekt nicht durchführbar	1	3	3	Früzeitige Besprechung mit Fachcoaches	1	1	1		Kein Weiterkommen
H	Soziale Spannungen im Team	Meinungsverschiedenheiten, schlechte Arbeitsaufteilung, keine Kompromissbereitschaft	Motivation sinkt, Arbeitsmoral sinkt, schlechte Projektarbeit, unzufriedener Arbeitgeber	2	2	4	Gegenseitige Kontrolle, Fehler offen im Team besprechen, Konstruktive Kritik	2	1	2		Schlechte Arbeitsmoral
I	Mangelnde Kommunikation	zu wenig Sitzungen, Angst vor Demütigung	Schlechtes Zusammenspiel, schlechtere Arbeit	2	1	2	Häufigere Sitzungen, höhere Wertschätzung der einzelnen Teammitglieder	1	1	1		Zurückhaltung
J	Nicht Termingerechte Abgabe der Arbeiten	Faulheit, mangelnder Einsatz, falsche Prioritäten, schlechte Projektführung	Terminplan kann nicht eingehalten werden	2	2	4	striktere Projektführung, gegenseitige Kontrolle, frühzeitiges Melden	2	1	2		Schlechte Arbeitsmoral
K	Qualitativ minderwertige Arbeit	Faulheit, mangelnder Einsatz, schlechter Teamgeist	Mehraufwand, Qualitativ ungenügende Arbeit, Zeitliche Probleme	2	2	4	Gegenkontrolle der Arbeiten	2	1	2		Schlechte Arbeitsmoral
L	Schlechte Terminplanung	Aufwand unterschätzt, keine Reserve eingeplant	Mehraufwand, Überarbeitung des Terminplans, Engpässe	3	2	6	Genug Reservezeit einplanen	2	1	2		Terminverzug

si= Eintrittswahrscheinlichkeit

pi= Auswirkung