Hochschule Luzern
Technik und Architektur

Bachelor Thesis

Gerät zur Überwachung von Umgebungslärm

Stefano Nicora

7. Juni 2024

Inhaltsverzeichnis

1 Einleitung **4**
1.1 Ausgangslage 4
1.2 Ziele 4
1.3 Vorgehen 5

2 Stand der Technik **6**

3 Mikrofon **7**
3.1 Grundlagen 7
3.2 Komponentenwahl 12
3.2.1 Kriterien 12
3.2.2 Vergleich 13
3.2.3 Fazit 13

4 Mikrocontroller **14**
4.1 Grundlagen 14
4.2 Komponentenwahl 16
4.2.1 Kriterien 16
4.2.2 Vergleich 17
4.2.3 Fazit 17

5 LED **18**
5.1 Grundlagen 18
5.2 Komponentenwahl 19
5.2.1 Kriterien 19
5.2.2 Vergleich 20
5.2.3 Fazit 20

6 Entwicklung **21**
6.1 Konzept 21
6.2 Hardware 23
6.2.1 PCB 23
6.2.2 Kosten 25
6.3 Software 26

7 Messungen **34**
7.1 Ausführungsgeschwindigkeit 34
7.2 Energieaufnahme 34
7.3 Mikrofon-Messungen 37
7.4 Vergleich 41

8 Validierung der Anforderungen **42**

9 Stolpersteine **43**

10 Fazit und Ausblick **44**

11 Anhang **46**
11.1 Fremdwörter & Abkürzungen 49
11.2 Messmittel 50
11.3 Messaufbau 51
11.4 Kalibrationsmessung 53
11.5 PCB-Versionen 54
11.6 Schema 56
11.7 Software 57
11.8 Aufgabenstellung 58
11.9 Anforderungskatalog 60
11.10 Risikoanalyse 61
11.11 Zeitplan 62
11.12 Journal 63

Abstract(DE)
Die nachfolgende Arbeit befasst sich mit der Entwicklung einer kombinierten Soft- und Hardware-Lösung zur Erfassung von Umgebungslärm. In einem ersten Schritt wurden die benötigten Bauteile identifiziert und evaluiert. Darauf folgte die Konzeptionierung, sowie die Entwicklung der eigentlichen Soft- und Hardwareplattform. Dabei hat die Hardware drei Iterationen durchlaufen. Im Anschluss wurde die Arbeit mit einem Referenzgerät verifiziert. Dieser Vergleich hat gezeigt, dass das Gerät zwar einsatzfähig ist, jedoch noch Verbesserungs- und Erweiterungspotential aufweist. Dazu wurden entsprechende Leitfäden formuliert.

Abstract(EN)
The following work deals with the development of a combined software and hardware solution for recording environmental noise. In a first step, the required components were identified and evaluated. This was followed by the conceptualisation and development of the actual software as well as the hardware. The hardware went through a total of three iterations. The work was then verified with a reference device. This comparison showed that although the device is operational, there is still potential for improvement and expansion. Guidelines were formulated for this exact purpose.

1 Einleitung

1.1 Ausgangslage

Die Firma hEar hat es sich zum Ziel gesetzt, gegen die, für unsere Ohren schädliche Lärmbelästigung, vorzugehen. Dazu wurde in der Masterarbeit von Sophie Mia Willener eine konzeptionelle Idee für ein Messgerät ausgearbeitet. Zusätzlich dazu wurde eine Marktanalyse durchgeführt, sowie ein erster Prototyp gebaut. Dieser ist jedoch noch unhandlich, nicht verifiziert und somit nicht für den Massenmarkt geeignet.

1.2 Ziele

Das Ziel dieser Arbeit ist es, in einem ersten Schritt, auf Basis des vorhandenen Prototypen, ein funktionales, kompaktes und portables Schalldruckpegel-Messgerät zu entwickeln. Dabei sollen folgende Rahmenbedingungen zwingend eingehalten werden:

- Die Laufzeit des Gerätes soll mindestens 12 Stunden betragen.
- Das Gerät wird mit einem Akku betrieben. Dieser wird via eines USB-C-Anschlusses aufgeladen.
- Der Schalldruckpegel wird mit einem MEMS-Mikrofon aufgezeichnet.
- Die Messdaten werden in regelmässigen Abständen auf dem Gerät gespeichert.
- Das Gerät verfügt über eine BLE-Schnittstelle um die Messdaten drahtlos an ein Zielgerät zu übertragen.
- Der aktuelle Schalldruckpegel wird auf der Vorderseite des Gerätes visuell dargestellt.

In einem zweiten Schritt, wird das Gerät ausgemessen und dessen Qualität mit auf dem Markt bereits vorhandenen Geräten verglichen.

1.3 Vorgehen

In einem ersten Schritt, wurden mit hEar die zu erreichenden Ziele der Arbeit ausgearbeitet. Auf Basis dessen, wurde der Anforderungskatalog (Anhang 11.9) erstellt. Dieser ermöglichte es anschliessend, eine umfangreiche Risikoanalyse (Anhang 11.10) sowie einen geeigneten Zeitplan (Anhang 11.11)zu erstellen. Nach der initialen Projektphase wurden die nötigen Grundlagen für Mikrofone (3), LEDs 5 sowie Mikrocontroller (4) aufgearbeitet und mit dem aktuellen Stand der Technik (2) abgeglichen. Anschliessend folgte die Wahl der benötigten Bauteile sowie die eigentliche Hardware- und Softwareentwicklung (6). Nach der agilen Entwicklung folgte zum Schluss die Verifizierung mit den dazugehörigen Messungen (7) und die daraus abgeleiteten Schlussfolgerungen.

2 Stand der Technik

Um den Schalldruckpegel messen zu können, existiert eine Vielzahl von Messgeräten von einer Vielzahl von unterschiedlichen Herstellern. Die Preise reichen von 10 CHF für ein nicht genauer spezifiziertes Gerät von AliExpress bis hin zu über 3000 CHF für ein XL2 von NTI Audio (inkl. M2211-Mikrofon). Eines haben jedoch alle diese Geräte gemeinsam. Sie weisen eine Frequenzbandbreite von mind. 10kHz auf, verfügen über ein Display um die Messwerte anzuzeigen, fangen den Schall mittels eines Kondensator-Mikrofons auf und implementieren mindestens den dB(A)-Gewichtungsfilter 3.1. Doch diese Geräte sind alle durch ihre klobige Bauweise sowie ihr, für nichttechnisches Personal ungeeignetes Design, nicht für den Massenmarkt geeignet. Nichtsdestotrotz soll das in dieser Arbeit entwickelte Gerät den Parametern eines hochwertigen Gerätes entsprechen. Dadurch wird es mit dem oben genannten XL2 mit den folgenden Parametern([1]) verglichen:

- Frequenzbereich: 5Hz - 20kHz
- Auflösung: 0.1dB
- Max. dBSPL: 144dBSPL
- Mikrofontyp: Kondensator

Abbildung 1: XL2 von NTI Audio | Quelle: [1]

3 Mikrofon

Als erste Kernkomponente gilt es, ein geeignetes Mikrofon zu evaluieren. Aus diesem Grund folgen anschliessend die benötigen Grundlagen, sowie die Komponentenwahl.

3.1 Grundlagen

MEMS

Mittels **M**icro **E**lectro-**M**echanical **S**ystems werden Bauteile bezeichnet, welche eine Strukturgrösse von wenigen μm aufweisen. Aus diesem Grund werden diese mehrheitlich auf Basis von Halbleitern hergestellt. Beispiele dafür sind Gyroskope (3), Drucksensoren, Oszillatoren oder eben, Mikrofone. Bedingt durch die Strukturgrösse, können hiermit kleinere Kräfte oder Drücke zuverlässiger detektiert werden. Bei Mikrofonen wird zwischen top- und bottom-port unterschieden (2), wobei je nach PCB-Layout nicht beide Varianten verwendet werden können.

Abbildung 2: Unterschied MEMS-Mikrofone | Quelle: [2]

Abbildung 3: Beispiel MEMS-Beschleunigungssensor | Quelle: [3]

I2S

Inter-**I**ntegrated **S**ound (I^2S) bezeichnet eine Bus-Schnittstelle, welche von Philips zur Übertragung von digitalen Audiosignalen entwickelt wurde. Ähnlich wie I^2C (Inter-Integrated Circuit) wird die Schnittstelle jedoch nur innerhalb des Gerätes verwendet. Dabei werden drei Pins zwischen Sender (hier das Mikrofon) und dem Empfänger (hier der Mikrocontroller) benötigt:

- **SCK** (Serial Clock)
  Generiert die Taktrate, welche gleichzeitig die Datenrate der Übertragung definiert. Die Taktrate wird vom Master (hier der Mikrocontroller) vorgegeben.

- **WS** (Word Select)
  Gibt vor, welcher Audiokanal (R, L) übertragen werden soll. Dies ermöglicht es, entweder ein Stereo-Signal oder zwei Mono-Signale, wie zum Beispiel zwei unabhängige Mikrofone, zu übertragen.

- **SD** (Serial Data)
  Beinhaltet den eigentlichen Datenstream mit der Datenrate, definiert durch **SCK**, und der Länge, definiert durch **WS**.

Abbildung 4: Übersicht I2S-Signale

PDM

Pulse **D**ensity **M**odulation oder Pulsdichtemodulation bezeichnet die Darstellung eines analogen Signals in der digitalen Ebene. Dabei entspricht das Verhältnis zwischen der Anzahl der digitalen "1" und digitalen "0" der Amplitude des analogen Signales. Das PDM-Signal weist eine vielfach schnellere Takrate (meist Faktor 32 / 64), als das ursprünglich abgetastete Mikrofonsignal auf. Dadurch wird auftretendes Rauschen in einen höher gelegenen Frequenzbereich verschoben, welcher bei der anschliessenden Signalverarbeitung mit einem Tiefpassfilter abgeschnitten wird. Im Gegensatz zu I²S (3.1) werden lediglich zwei Pins zwischen Mikrofon und Empfänger benötigt:

- DAT (Data)
  Beinhaltet den als PDM-Codierten Datenstream.
- CLK (Clock)
  Generiert die Taktrate, mit welcher die Daten übertragen werden.

Abbildung 5: Beispiel PDM-Signal

Abbildung 6: Beispiel PCM-Signal

PCM

Bei **P**ulse **C**ode **M**odulation wird jeder analoge Messwert in einem digitalen Bereich abgebildet (Abbildung 6). Je nach benötigter Genauigkeit fällt dieser Bereich mal grösser, mal kleiner aus. Bei der Datenübertragung mittels I²S

(3.1) geschieht diese Umwandlung direkt im Sender und erzeugt somit das SD-Signal. Bei Sendern mit PDM-Ausgang (3.1) ist dies nicht der Fall. Hier ist im Empfänger zwingend eine Peripherie notwendig, welche das PDM-Signal mittels Unterabtastung in ein PCM-Signal umwandelt.

Schalldruck(pegel)

Damit wird der Bezug zwischen dem Schalldruck P, die Kraftauswirkung von Schallwellen auf ein Medium, und dem normierten Bezugswert P_0 von $20\mu Pa$ geschaffen. Oftmals wird er auch mit **S**ound **P**ressure **L**evel (SPL) angegeben. Da das menschliche Gehör Lautstärke jedoch logarithmisch wahrnimmt, wird dieser in der Akustik mittels dBSPL 1 angegeben.

$$dBSPL = 20 \cdot log_{10}(\frac{P}{P_0}) \tag{1}$$

Abbildung 7: Grundbegriffe Akustik | Beispiel: ICS-41351

AOP

Der **A**coustic **O**verload **P**oint bezeichnet den durch ein Mikrofon maximal messbaren SPL, ohne dass es dabei zu einem Klirrfaktor (THD) von mehr als 10% kommt. Dieser ist besonders dann wichtig, wenn Messungen in lauten Umgebungen vorgenommen werden müssen.

SNR

Signal to **N**oise **R**atio oder Signal Rausch Abstand beschreibt die Differenz zwischen dem Referenzschalldruckpegel von 94dBSPL bei 1kHz und dem Rauschboden. Je höher diese Distanz, desto exakter sind die getätigten Messungen.

Rauschboden

Dieser bezeichnet den kleinstmöglich noch messbaren Wert, welcher nicht im Grundrauschen untergeht. Er ist vom SNR des Mikrofons abhängig.

Dynamikbereich

Mit dem Dynamikbereich 7 wird der maximal messbare Schalldruck-Unterschied bezeichnet. Dieser wird folgendermassen berechnet:

$$DynamicRange(DR) = \underbrace{dBFS_{@1kHz}}_{Datenblatt[10]} + \underbrace{SNR}_{Datenblatt[10]} \tag{2}$$

Signalauflösung

Abbildung 8: Min. benötigte Auflösung in Bit für gewünschten max. SPL

Zeitgewichtung

TODO

Frequenzbewertung

Unser Gehör reagiert nicht nur nicht linear auf Lautstärkeveränderungen, sondern zusätzlich auch unterschiedlich Sensitiv auf unterschiedliche Frequenzen im hörbaren Bereich (20Hz-20kHz). Insbesondere im Bereich zwischen 3kHz bis 4kHz [4] ist das Gehör besonders anfällig auf äussere Stimulation. Abbildung 9, ursprünglich durch Fletcher & Munson, mittlerweilen auch durch den ISO226-Standard beschrieben, zeigt diesen Umstand. Um der Problematik aus Abbildung 9 sorge tragen zu können, wird ein sogenannten

Frequenzbewertungsfilter [1], Abbildung 10, eingesetzt. Dabei werden heutzutage nur noch die dB(A), sowie der dB(C)-Filter eingesetzt. Die beiden Filter unterscheiden sich folgendermassen:

- dB(A)
  Reproduziert die 40 Phon Kurve
  Widerspiegelt das menschliche Gehör besser, als der dB(C)-Filter

- dB(C)
  Reproduziert die 100 Phon Kurve.
  Filtert weniger tiefe und hohe Frequenzen als dB(A) heraus
  Wird verwendet, um besonders laute Umgebungen zu messen

Abbildung 9: Phon-Kurven im Vergleich | Quelle: [4]

Abbildung 10: Frequenzbewertungsfilter | Quelle: [5]

3.2 Komponentenwahl

3.2.1 Kriterien

- Standort der Mikrofon-Öffnung
- Ausgangssignal
- Maximaler Schalldruckpegel

[1] Weighting curves

- MEMS-Technologie
- Frequenzbereich

3.2.2 Vergleich

Typ	**Protokoll**	**Öffnung**	**SNR[dB]**	**AOP[dB]**	f_{min}**[Hz]**	f_{max}**[Hz]**
ICS-43434	I2S	Unten	65	120	60	20000
ICS-41351	PDM	Oben	65	120	50	20000
MMICT5838-00-012	PDM	Unten	68	133	27	20000
CMM-4737DT-26386-TR	PDM	Oben	58	120	100	10000
CMM-4737DT-26186-TR	PDM	Oben	58	120	100	10000
MP34DT05(-A)	PDM	Oben	64	122.5	100	10000
MP34DT06J	PDM	Oben	64	122.5	100	10000
SPH0645LM4H-B	I2S	Unten	65	120	20	10000

Tabelle 1: Vergleich Mikrofone

3.2.3 Fazit

$\frac{1}{3}$ der untersuchten Mikrofone weisen die Mikrofon-Öffnung auf der Unterseite auf. Dies ermöglicht es, einen höheren SNR gewährleisten zu können, jedoch ist diese Montageart für das Endprodukt nicht geeignet (siehe Kapitel 6). Bei den restlichen Mikrofonen sticht das ICS-41351 von TDK InvenSense mit seinem hohen SNR, AOP sowie der hohen Frequenzbandbreite heraus. Dadurch hat dieses das Rennen gewonnen.

4 Mikrocontroller

Der Mikrocontroller repräsentiert die zweite Kernkomponente des Systems. Aus diesem Grund folgen anschliessend die benötigen Grundlagen, sowie die eigentliche Komponentenwahl.

4.1 Grundlagen

DMA

Bei **D**irect **M**emory **A**ccess handelt es sich um eine Art Steuerbaustein, welcher unabhängig und parallel zum eigentlichen Rechenkern arbeitet. Dieser hat, wie der Rechenkern selbst, Zugriff auf den gesamten Datenbus. Dieser ermöglicht es, effizient Daten von Peripherien wie der UART, I^2S, SPI, ect. in den RAM und umgekehrt zu laden. Dadurch wird der Rechenkern entlastet und kann für wichtigere Aufgaben eingesetzt oder gar in einen Energiesparmodus versetzt werden. Dabei sind zwei Möglichkeiten vorhanden, mit dem DMA-Baustein zu interagieren:

- Bus Master
  Der DMA-Baustein wird einmalig durch den Rechenkern aufgesetzt und läuft danach autonom. So können beispielsweise via I^2S einkommende Daten automatisch aus dem Eingangsregister in den Speicher geladen werden. Bei genügend grosser Datenmenge (definiert durch den Entwickler), kann der Rechenkern wieder aus dem Energiesparmodus aufgeweckt werden und diese Daten verarbeiten.

- Programmed I/O
  Jeder Arbeitsprozess des DMA-Bausteins wird durch den Rechenkern gestartet. Da sich DMA und Rechenkern den Datenbus teilen, wird so garantiert, dass der Datentransfer zum gewünschten Zeitpunkt stattfindet.

Bootloader

Unter einem Bootloader versteht man Code, welcher grundsätzlich nach dem erstmaligen flashen, persistent im System bleibt. Er dient als Ausgangspunkt für die eigentliche Software und ermöglicht es beispielsweise auch, während der Laufzeit des Gerätes, Softwareupdates herunter zu laden und diese anschliessend einzuspielen. Insbesondere wenn Funkprotokolle wie Bluetooth

oder WLAN eingesetzt werden, ist ein Bootloader oftmals unerlässlich. Der Nachteil ist jedoch, dass dieser den ohnehin knappen Speicherplatz weiter reduziert.

RTC

Unter einer **R**eal **T**ime **C**lock versteht man ein Stück Hardware (intern oder extern), welche die Zeit ab Inbetriebnahme misst. Übliche RTCs haben, aufgrund ihrer Bauweise eine Genauigkeit von 100ppm. Da dies einer Zeitverschiebung von 4min und 19s @32kHz entspricht, enthalten die meisten RTCs Kompensationsmechanismen, um diese Zeit auf unter 100ms zu bringen. [6]

Low Power

Mit Low Power wird, insbesondere bei Embedded-Systemen, die reduzierte Energieaufnahme beschrieben. Oftmals verfügen die Systeme über mehrere Modi mit jeweils unterschiedlichen Eigenschaften. Das Ziel ist es, mit jedem tieferen Modus etwas mehr Energie zu sparen und somit die Akkulaufzeit zu verlängern. Wie in Abbildung 11 ersichtlich, sinkt zwar die Energieaufnahme mit jeder Stufe, jedoch werden dabei gewisse Peripherien abgeschaltet und die allgemeine Startzeit des Systems wird erhöht.

Abbildung 11: Vergleich Low-Power-Modi STM | Quelle: [7]

Peripherie in Hardware oder Software

Bei allen gängigen Kommunikationsprotokollen ist die Frage, wie diese implementiert werden sollen. Dabei stehen folgende Möglichkeiten zur Auswahl:

- in **Hardware**
  + Die benötigte Infrastruktur (Register, CRC, etc.) sind bereits physikalisch implementiert. Die Kommunikation kann dementsprechend vom Rechenkern abgekoppelt durchgeführt werden und ermöglicht eine gewisse Parallelität von mehreren Prozessen mit wenig Rechenaufwand. Insbesondere in Verbindung mit DMA, kann so viel Rechenzeit eingespart werden.
  - Bei den meisten Mikrocontrollern ist die spezifische Hardware an einzelne Pins gebunden. Dies wirkt sich auf die Flexibilität des PCB-Layouts oder gar des Gesamtsystems (zu wenig Anschlüsse) aus.

- in **Software**
  + Die meisten Pins können flexibel für jedes Protokoll verwendet werden. Dadurch vereinfacht sich auch das PCB-Layout.
  - Bei Verwendung von Ein-Kern-Systemen, welche im Embedded-Bereich mehrheitlich anzutreffen sind, kann keine echte Parallelität von Prozessen erfolgen. Dies verringert die Ausführungsgeschwindigkeit.

Grundsätzlich ist, nach Möglichkeit und Verfügbarkeit, eine Hardware-Peripherie zu bevorzugen. Insbesondere bei batteriebetriebenen Systemen mit langer Laufzeit, ist diese ausschlaggebend.

4.2 Komponentenwahl

Es existieren eine Vielzahl von Mikrocontroller-Herstellern. Viele dieser verfügen über eine breite Palette an BLE-tauglichen Chips. Um den geeignetsten darunter zu finden, werden nachfolgend die benötigten Schnittstellen definiert und mehrere, vorselektionierte Mikrocontroller, miteinander verglichen.

4.2.1 Kriterien

- **I²S** + **PDM** zur Ansteuerung des Mikrofons.
- **SPI** zur Ansteuerung von möglichen Peripherien wie externe Speicherbausteine oder Displays.

- **I²C** zur Ansteuerung von Laderegler und LED-Treiber-IC.
- **DMA** zur energieeffizienten Verarbeitung der Daten.
- **RTC** um die gemessenen Werten mit einem Zeitstempel verknüpfen zu können.
- **BLE**-Fähiger Mikrocontroller mit vorgefertigter Antenne um die Integration zu vereinfachen.

4.2.2 Vergleich

Hersteller	Typ	I2S	I2C	SPI	DMA	BLE	PDM	RTC	ANT	Flash (kB)	uA/MHz	Zusatz
Infineon	CY8C63x6	x	x	x	x	x	x	x	-	<=512	40	Nicht via normale Distributoren
Infineon	CY8C63x7	x	x	x	x	x	x	x	-	<=512	40	Nicht via normale Distributoren
Microchip	BM83	x	x	-	-	x	x	-	-	-	-	Charge-controller für Akku **Not recommended for new Designs**
Nordic Semiconductor	nRF52832	x	x	x	x	x	x	x	x	<=512	58	ANT: BMD-350-A-R (u-blox)
Nordic Semiconductor	nRF52833	x	x	x	x	x	x	x	x	512	52	ANT: NINA-B406-00B (u-blox) BL653 (Lairdconnect)
Nordic Semiconductor	nRF52840	x	x	x	x	x	x	x	x	1024	52	ANT: NINA-B302-00B (u-blox) BMD-340-A-R (u-blox) BL654PA (Lairdconnect)
Nordic Semiconductor	nRF5340	x	x	x	x	x	x	x	x	1024	48	ANT: NORA-B106-00B (u-blox) BL5340 (Lairdconnect)
Silicon Labs	EFR32BG22	x	x	x	x	x	x	x	x	<=512	27	ANT: BGM220P Lyra P (Lairdconnect)
Silicon Labs	EFR32BG27	x	x	x	x	x	x	x	-	768	29	-
STmicro	STM32WB		x	x	x	x		x	x	1024	53	ANT: STM32WB5MMG
STmicro	BlueNRG-LP	x	x	x	x	x	x	x	-	256	18	-

Abbildung 12: Vergleich von möglichen Mikrocontrollern

4.2.3 Fazit

Wie dem Vergleich zu entnehmen ist, unterscheiden sich die Mikrocontroller meist nur in der integrierten Flash-Grösse sowie der Energieaufnahme bei einer gewissen Taktrate. Dadurch wird die Wahl nicht eingeschränkt, was die Evaluation nicht vereinfacht. Ausschlaggebend ist jedoch der Formfaktor. Bis auf Silicon Labs und STmicro muss für einen Mikrocontroller mit vorgefertigter Antenne auf Drittanbieter ausgewichen werden. In Zeiten von Katastrophen wie COVID oder Kriegen inmitten von Handelsrouten ist die Produktverfügbarkeit einer der zentraler Aspekte. Aus diesem Grund wird nachfolgend der EFR32BG22 von Silicon Labs bzw. dessen Modul-Variante, der **BGM220P**, eingesetzt.

5 LED

Die visuelle Darstellung des Schalldruckpegels kann auf verschiedene Arten umgesetzt werden. Da hEar explizit eine Darstellung mittels LED wünscht, werden nachfolgend die wichtigsten Charakteristiken eingeführt und anschliessend ein passendes Produkt evaluiert.

5.1 Grundlagen

Wellenlänge

Das für den durchschnittlichen Menschen sichtbare Lichtspektrum beginnt bei 380nm (blau) und endet bei 780nm (rot)[8]. Wie der Abbildung 13 zu entnehmen ist, entspricht die Sensitivität einer Gausskurve. Dabei erreicht diese ihr Maximum bei 555nm (grün) bei Betrachtung der Tageskurve. Dementsprechend kann beim Einsatz einer 555nm-LED im Gegensatz zu LEDs mit einer tieferen oder höheren Wellenlänge die zugeführte Leistung bei gleichwertiger Lichtempfindlichkeit reduziert werden.

Abbildung 13: Sensitivität des menschlichen Auges auf das Lichtspektrum bei Tag (**schwarz**) und Nacht (**blau**)| Quelle: [8]

Lichtleistung

Neben der Wellenlänge, ist die Lichtleistung ausschlaggebend für die wahrgenommene Helligkeit einer LED. Dabei wird die Lichtleistung in Candela [cd] angegeben, wobei 1 cd der Helligkeit einer Kerze entspricht. Bei einer typischen LED verhält sich die Lichtleistung linear mit dem zugeführten Strom.

Vorwiderstand / Strombegrenzung

Nach Anlegen einer Spannung, >= Schwellspannung der LED [2], wird diese grundsätzlich zu einem Kurzschluss. Um den Stromfluss durch die LED begrenzen zu können, wird im Regelfall ein Vorwiderstand verwendet. Der Nachteil des Vorwiderstandes ist es, dass, gegeben durch das ohmsche Gesetzt, über diesem elektrische Leistung in Wärme umgewandelt wird. Bei einem akkubetriebenen Gerät ist dies unerwünscht. Als Alternative kann ein Buck Converter, Abbildung 14, oder eine Konstantstromquelle eingesetzt werden. Dadurch entfällt der Vorwiderstand und die Effizienz des Teilsystems wird erhöht.

Abbildung 14: Buck Converter

5.2 Komponentenwahl

5.2.1 Kriterien

- Lichtleistung
- Wellenlänge

[2] variiert je nach LED Typ und Farbe

5.2.2 Vergleich

Hersteller	**Typ**	**Farbe**	**Intensität[mcd]**[3]	**Spannung[V]**
SunLED	XZCM2CRK53WA-8VF	Rot	180	2.1
SunLED	XZCDGK53W-8VF	Grün	250	3.1
SunLED	XZCFBB53W-8VF	Blau	30	3
Worldsemi	WS2018	Rot	55	2.2
Worldsemi	WS2018	Grün	110	3.1
Worldsemi	WS2018	Blau	20	3.4
RND	135-00184	Rot	7	1.6

Tabelle 2: Vergleich LED

5.2.3 Fazit

Wie im Kapitel 5.1 ersichtlich, ist die Abwägung zwischen Energieaufnahme der LED und der effektiven Photonenabgabe nicht trivial. Aus technischer Sicht macht dementsprechend eine LED mit einer Wellenlänge von 555nm (Grün) am meisten Sinn. Nach Rücksprache mit dem Industrieparter, hEar, dient das Gerät an erster Stelle als Warngerät. Warnungen werden üblicherweise mit einer Wellenlänge von 780nm (Rot) gekennzeichnet. Dadurch wird im Endprodukt die Low-Power-LED XZCM2CRK53WA-8V von SunLED eingesetzt.

[3] bei einem Strom von 2mA

6 Entwicklung

Die im Unterkapitel 1.2 definierten Ziele sollen nun umgesetzt werden. Dazu soll die von aussen sichtbare Vorderseite möglichst wenig Bauteile aufweisen, um das visuelle Design nicht zu beeinträchtigen. Mit einer angestrebten Gesamtgrösse von 60mm Durchmesser, können die gewählten Bauteile geradenoch auf dem PCB in einer symmetrischen Art platziert werden.

6.1 Konzept

Front Die Vorderseite weist folgende Bauteile auf:

- 8 LEDs
- Mikrofon
- LED-Treiber-IC

Zudem sind die LEDs sind ringförmig angeordnet, um der Anforderung eines visuellen Ladebalkens entsprechen zu können.

Back Die Rückseite hingegen, beinhaltet folgende Bauteile:

- USB-C-Ladeanschluss
- Polyfuse (Automatisch rückstellbarer Überstromschutz)
- Schalter um generell das Gerät ein- und auszuschalten
- BMS, um den Akku sicher auf- und entladen zu können
- Mikrocontroller, welcher sich aufgrund seiner Antenne am Rand des PCBs befinden muss
- Akkuhalter
- Optionaler Speicherbaustein

Abbildung 15: Konzept PCB-Layout

Abbildung 16: Konzept Softwareablauf

6.2 Hardware

6.2.1 PCB

Das PCB durchlief mehrere Iterationen. Nachfolgend treten die Änderungen mit den benötigten Zusatzinformationen chronologisch auf [4]. Zudem findet sich eine Übersicht, wieso, welches Bauteil eingesetzt wird.

V1-1
Lediglich kleinere Verdrahtungs- und Platzierungsoptimierungen.

V1-2
Folgende Anpassungen wurden vorgenommen:

- Akku-Verpolungsschutz mittels P-Kanal MOSFET.
- Testpads für I^2C, SPI und PDM entfernt.
- USB-C 16 Pin Buchse durch 6 Pin Buchse (power only) ersetzt.
- Rotation der LEDs um 40° im Gegenuhrzeigersinn, um die Reihenfolge der LEDs zu gewährleisten. Zusätzliche Optimierung der Platzierung der LEDs, um die Verdrahtung zu vereinfachen.
- TRIG-Pin des LED-Treiber-ICs auf Masse geschaltet.
- Spannungsteiler zur Messung der Speisespannung via USB-C.
- Optionaler Pullup-Widerstand an DRV_EN um den Anschluss nicht immer manuell schalten zu müssen.

V1-3
Auf Wunsch von hEar wurde zusätzlich noch ein PCB entwickelt, welches funktionsidentisch mit dem Vorgänger ist. Einzig die LEDs sind nun nicht mehr ringförmig angeordnet, sondern erstrecken sich neu auf einer vertikalen Linie.

[4] Detailierte Abbildungen der Versionen befinden sich im Anhang 11.5

Sicherung
Der Fall eines Kurzschlusses auf Geräteseite wird mittels einer rückstellbaren Sicherung (PPTC[5]) der Stromfluss unterbrochen. Die gewählte Sicherung blockiert bei einem Strom von 400mA komplett und setzt sich automatisch wieder zurück, sobald die Speisung abgetrennt wird.

LED-Treiber
Die Aufgabe des Treibers ist es, den LED-Strom möglichst Energieeffizient zu regeln (siehe 5.1). Der gewählte Treiber ermöglicht es, alle 8 LEDs individuell in 100uA-Schritten zu regeln. So kann die Energieaufnahme des Gesamtsystems kontrolliert werden.

Akku-Laderegler | BMS
Das **B**attery **M**anagement **S**ystem wurde aufgrund folgender, benötigter Eigenschaften gewählt:

- Möglichst wenige externe Bauteile
- Über- und Unterspannungsschutz
- Überstromschutz
- Keine direkte Ansteuerung mittels Mikrocontroller nötig
- Möglichkeit zur visuellen Darstellung des Ladestatus
- Unterstützung von Li-Po und Li-Ion Akkus

Spannungsregler
Der Akku weist eine maximale Nennspannung von 4.2V (Typ. 3.6V) auf. Alle gewählten Halbleiter erlauben jedoch maximal eine Speisespannung von 3.6V. Dadurch wird ein möglichst effizienter und kostengünstiger Spannungsregler verwendet. Wie auch bereits das BMS, weist der Spannungsregler möglichst wenig externe Bauteile auf.

[5] **P**olymeric **P**ositive **T**emperature **C**oefficient, automatisch rückstellbare Sicherung von Littlefuse

USB-C-Buchse
Die in V1-0 verwendete Buchse (16 Pin) wird durch eine 6 Pin (power only) ersetzt. Dies, da die ursprünglich geplante Möglichkeit, den Mikrocontroller via USB zu programmieren, wieder verworfen wurde. Dadurch kann eine kostenoptimiertere Buchse verwendet werden.

P-Kanal MOSFET
Für den Verpolungsschutz kann im Grunde auch eine simple Diode (oder Schottky-Diode[6]) verwendet werden. Der daraus folgende Spannungsabfall ist jedoch für ein akkubetriebenes Gerät nicht ideal. Die Lösung bietet ein P-Kanal-MOSFET, wie in Abbildung 17 ersichtlich. Die Z-Diode, sowie der Gate-Widerstand, werden jedoch weggelassen, da die Batteriespannung die maximale Source-Gate-Spannung des gewählten FETs nicht überschreitet.

Abbildung 17: Verpolungsschutz mittels P-Kanal MOSFET

6.2.2 Kosten

Die Tabelle 3 beinhaltet alle anfallenden Materialkosten exkl. Versand. Zudem wurde JLCPCB als PCB-Hersteller gewählt, da zum einen die Qualität bei 2-Layer-PCBs ausreicht und zum anderen mit der Anbindung an LCSC die Bestückung der PCBs direkt im selben Werk stattfindet. Dadurch verkürzt sich die Fertigung des Endproduktes.
Die vorhandenen Kondensatoren, sowie die benötigten Widerstände werden in der Tabelle 3 unter Diverse zusammengenommen. Dies aus dem Grund,

[6]Die Schwellspannung entspricht nur ca. 0.2V

dass keines dieser Bauteile einer besonderen Toleranz unterliegt. Dadurch kann bei der PCB-Bestellung eine kosten-optimierte Wahl getroffen werden.

Hersteller	**Typ**	**Bezeichnung**	**Preis/100**
JLCPCB	-	PCB inkl. Versand	0.394
Texas Instruments	BQ21040DBVR	Batterie-Management	0.516
Microchip Technology	MIC5504-3.3YM5-TR	3V3 LDO	0.095
RS PRO	LIR2477	Akku	4.085
Littlefuse	1210L020WR	Rückstellbare Sicherung	0.205
Texas Instruments	LP55231SQX/NOPB	9-Kanal LED-Treiber	1.35
SunLED	XZCM2CRK53WA-8VF	LED RED CLEAR	0.2072
C&K	JS102011SAQN	Schalter 0.3A@6V	0.453
Keystone Electronics	1025-7	Batteriehalter	1.26
STMicroelectronics	M95P16-IXMNT/E	16MBit SPI EEPROM	0.843
Silicon Labs	BGM220PC22WGA2R	Mikrocontroller	6.33
TDK InvenSense	ICS-41351	MEMS-Mikrofon	1.05
GCT	640-USB4125-GF-A	USB-C Buchse 6 Pin	0.381
Toshiba	SSM3J334R,LF	P-Channel MOSFET	0.109
Diverse	Capacitor & Resistor	Abmessung: 0603 [Inch]	0.6
			17.90 CHF

Tabelle 3: Materialliste | Stand: 01.04.2024

6.3 Software

Im Kern besteht die Software, wie in Abbildung 16 gezeigt, aus einem einzelnen, sich wiederholenden Kreis. Durch den einfach gehaltenen Ablauf, wird auf den Einsatz eines RTOS verzichtet. Um diesen Abschnitt übersichtlich zu halten sind Timing-Diagramme sowie die Software im Anhang 11.7 zu finden.

Timer

Die Timer-Implementation dient als Hauptschleife, um die Datenaggregation, Verarbeitung und Ausgabe steuern zu können.
Das Makro **BAT_MEASUREMENT_INTERVAL** ermöglicht es, auf wenige

Millisekunden genau die Schlaufe zu steuern. Die Programmabfolge ist in Abbildung 18 ersichtlich.

Abbildung 18: Übersicht zu (LE)Timer-Funktionalität

Filterdesign

Wie im Unterkapitel 3.1 erläutert, reagiert unser Gehör nicht auf alle Frequenzen gleich. Um diese Frequenzbewertung in die Schalldruckpegel-Berechnung

einfliessen zu lassen, wird ein digitales, diskretes Filter benötigt. Die benötigten Frequenzbewertungsfilter sind jedoch ausschliesslich in der S-Ebene (Laplace-Bereich, zeitkontinuierlich) erhältlich. Dabei stehen uns, wie in Abbildung 19 ersichtlich, diesbezüglich zwei Wege offen:

- **Frequenzbereich**
  Hier wird das PCM-Codierte Signal mittels einer Fast Fourier Transformation (FFT) in den Frequenzbereich transformiert. Anschliessend können die Formeln 3 oder 4 auf das Signal angewendet werden, um die gewünschte Gewichtung vorzunehmen. Um anschliessend den Schalldruckpegel berechnen zu können, muss das Signal mittels einer inversen FFT wieder in den Zeitbereich transformiert werden. Je genauer die Frequenzbewertung sein soll, desto kleiner sind die Abstufungen von FFT und IFFT zu wählen, was zu einer Erhöhung des Berechnungsaufwandes auf dem Mikrocontroller führt.

- **Zeitbereich**
  Im Gegensatz zum Frequenzbereich muss das PCM-Codierte Signal nicht erst in den Frequenzbereich transformiert werden, sondern es kann direkt durch das zeitdiskrete Filter die Gewichtung vorgenommen werden. Dabei treten jedoch zwei Hauptprobleme auf:

  - Ein diskretes Filter im Zeitbereich ist immer nur eine Approximation eines Filters im Frequenzbereich. Dies verringert die Genauigkeit des implementierten Filters.
  - Die Umwandlung eines zeitkontinuierlichen Filters in einen zeitdiskreten ist nicht trivial. Dadurch muss dieser mit beispielsweise der bilinaren Transformation [9] vom Frequenz- in den Zeitbereich transformiert werden.

  Die benötigten A- und C-Gewichtungsfilter bereits sind in Matlab implementiert. Somit können die Filterkoeffizienten direkt exportiert und in C-Code implementiert werden.

$$G_A(s) = \frac{7.39705 \cdot 10^9 \cdot s^4}{(s+129.4)^2 \cdot (s+676.7) \cdot (s+4636) \cdot (s+76617)^2} \tag{3}$$

$$G_C(s) = \frac{5,91797 \cdot 10^9 \cdot s^2}{(s+129,4)^2 \cdot (s+76617)^2} \tag{4}$$

Abbildung 19: Unterschied Zeit- und Frequenzbereich zur Frequenzgewichtung

Filterimplementation

Die aus Matlab exportieren Filterkoeffizienten weisen einen entscheidenden Unterschied zu gewöhnlichen IIR-Filterkoeffizienten auf. Der Gesamtfilter wird in Teilfilter, sogenannte BiQuads, aufgeteilt, welche anschliessend kaskadiert werden. Dies, weil die Stabilität von Filtern mit tiefer Ordnung leichter kontrolliert werden kann, als bei jenen mit einer hohen Ordnung. Dadurch besteht ein einzelnes BiQuad-Filter-Element jeweils aus einem IIR-Filter zweiter Ordnung. Dabei existieren, wie in Abbildung 20 aufgezeigt, vier verschiedene Arten der BiQuad-Implementation. Oftmals verfügen Mikrocontroller bereits über eine Hardware-Implementation einer dieser Arten. Der in dieser Arbeit gewählte, jedoch nicht. Dadurch sind wir frei in der Wahl der Implementation. Aus diesem Grund wurde die "Transposed Direct Form 2" gewählt. Dies, da diese zwar mehr Variablen zur als Zwischenspeicher, jedoch weniger Rechenoperationen, benötigt.

Abbildung 20: Übersicht BiQuad-Filter

PDM Clock

Das Mikrofon benötigt einen exakten Clock, um die PDM-Daten an den Mikrocontroller bzw. dessen PDM2PCM-Periphiere zu übertragen. Um das Mikrofon im "HIGH PERFORMANCE MODE" laufen lassen zu können, wird gemäss Datenblatt ([10]) ein Clock von 3.072MHz benötigt. Da dieser nicht direkt aus einem der bereits verfügbaren Clocks abgeleitet bzw. herunter geteilt werden kann, muss dieser mittels dem integrierten PLL erzeugt werden. Der Prozess ist in Abbildung 21 ersichtlich, wobei der effektive PLL-Clock mittels der Formel 5 berechnet werden muss.

$$Frequenz_{PLL} = \underbrace{Frequenz_{ref}}_{LFXO} \cdot \frac{\overbrace{N+1}^{300<=N<=4095}}{\underbrace{M+1}_{0<=M<=4095}} \tag{5}$$

Abbildung 21: Clock-Muxing für 3.072MHz PDM

PDM CIC-Filter

Um aus dem 1-Bit PDM-Datenstream ein verarbeitbares PCM-Signal zu erhalten, wird die integrierte CIC-Filterstruktur verwendet. Gemäss [11], 7 sowie [10] muss diese folgendermassen konfiguriert sein:

- Dezimierungsrate: 64
- Wortbreite: 24 Bit
- Verstärkung 7: 0 dB
- Filterordnung: 5
- Clock-Dezimierung 6: 0
- Abtastrate: 48 kHz

$$Dezimierung = \frac{Taktfrequenz}{Abtastrate \cdot Dezimierungsrate} - 1 \tag{6}$$

$$Verstärkung = 30 - ceil\left(\frac{log_{10}\left(Dezimierungsrate^{Filterordnung}\right)}{log_{10}(2)}\right) \tag{7}$$

I²C

Abbildung 22: Softwareablauf I²C

SPI

Da der SPI-Treiber hauptsächlich mittels der SPIDRV-Bibliothek von Silicon Labs implementiert ist, fällt der Treiber schlank aus. Es existieren lediglich jeweils eine Funktion um eine neue Seite (512 Bytes) im Speicher zu beschreiben und um diese auch wieder auszulesen bzw. zu löschen.

RTC

Die RTC wird, nicht wie die restliche Peripherie, nicht selbst aufgesetzt und betrieben. Dies ist darauf zurückzuführen, dass der BLE-Stack von Silicon Labs zum Zeitmanagement denselben Zeitgeber verwendet. Dadurch ist der Zugriff auf die RTC limitiert und kann nur über die nachfolgende Funktion abgerufen werden:

$$sl_sleeptimer_get_time()$$

Dabei wird die Zeit auf die Sekunde genau zurückgegeben und kann somit für die Erzeugung des Zeitstempels verwendet werden.

LEDs

Auf Wunsch des Industriepartners, werden die SPL-Werte so auf die 8 LEDs abgebildet, dass der 85dB(A)-Wert auf LED 4 zu liegen kommt mit jeweils einer Veränderung von ± 3dB(A) auf die nachfolgenden LEDs. Dabei entspricht ein Schritt von +3dB, gemäss Formel 1, einer Lautstärkeverdoppelung. Somit sieht die momentane Skalierung folgendermassen aus:

Abbildung 23: Aufteilung dBSPL auf 8 LEDs

7 Messungen

Ein System kann qualitativ nur mit Messungen auf seine Einsatzfähigkeit untersucht werden. Dadurch erfolgen nachfolgend theoretische und praktische Messwerte.

7.1 Ausführungsgeschwindigkeit

Um die Dauer einer Funktion messen zu können, wurde ein GPIO-Pin zu Beginn sowie am Ende der jeweiligen Funktion umgeschaltet und mit den Logic-Analyzer ausgewertet. Aus 20 Funktionsaufrufen wurde anschliessend der Mittelwert berechnet und in Tabelle 4 aufgeführt.
Die Messungen zeigen, dass hauptsächlich das messen der Werte, sowie die Filterung dieser, die meiste Zeit in Anspruch nimmt. Gleichzeitig kann jedoch so verifiziert werden, dass ein Messintervall von 125ms (Messmodus: Fast) möglich ist.

Funktionsname	**Zeit [ms]**	**Bemerkung**
app_init	10.4	RTC, SPI, I2C, PDM, Timer, GPIO
PDM_readMicrophone	20.8	128 Samples; Low-Power
RTC_convertTimeToString	0.16	Format: 00:00:00
PDM_applyAWeightingFilter	15.2	128 Samples
PDM_convertPCMTodBSPL	0.81	128 Samples
PDM_convertSPLToString	0.28	
I2C_enableLedRange	0.62	
LOGGING_generateLogString	0.01	Länge: 13 char
SPI_writePage / SPI_readPage	4.4	512 Bytes
Total	52.68 ms	
Total pro Messintervall	**42.28** ms	

Tabelle 4: Ausführungsgeschwindigkeit der einzelnen Funktionen

7.2 Energieaufnahme

Die Energieaufnahme eines Systems zu quantifizieren, stellt sich als durchaus komplex dar. Viele Variablen, Abwägung von Verarbeitungsgeschwindigkeit

mit der gegensätzlichen Energieeffizienz und nicht genau spezifizierte Parameter erschweren die Festlegung der Rahmenbedingungen. Bei der Auslegung der nachfolgenden Parameter wird stets die energieeffizientes Möglichkeit gewählt, ohne damit die Leistungsfähigkeit massgeblich zu beeinflussen.

Mikrocontroller
Die Energieaufnahme des Mikrocontrollers ist nicht nur von der eigentlichen Taktrate abhängig, sondern auch welche Peripherieblöcke und Low-Power-Modi gerade aktiv sind. Dadurch lässt sich die Energieaufnahme nur sehr grob abschätzen. Mit den benötigten Peripherien aktiv, jedoch ohne eine Berechnung durchzuführen und in einen Low-Power-Modus zu wechseln, könnte die Energieaufnahme folgendermassen aussehen:

$$\approx \underbrace{27\mu A/MHz \cdot 76.8MHz}_{\mu C} + \underbrace{10\mu A/MHz \cdot 76.8MHz}_{Peripherie} = 2842\mu Ah \quad (8)$$

Mikrofon
Wie der Formel 9 entnommen werden kann, gibt es zwei Möglichkeiten das Mikrofon zu betreiben:

- Das Mikrofon wird konstant im Performance-Modus betrieben. Durch den Einsatz von DMA sind die benötigten Messewerte jeder Zeit verfügbar und die 20ms Startzeit des Mikrofons entfällt. Dadurch sind höher aufgelöste Messintervalle möglich. Dazu sind die Messungen der einzelnen Funktionen in Tabelle 4) ersichtlich.

- Das Mikrofon wird nur dann aktiviert, wenn es benötigt wird. Dadurch sinkt die benötigte Leistung von $580\mu Ah$ auf 87.2 μAh (9). Dadurch erhöht sich jedoch die benötigte Messzeit um 20ms. Dies aus dem Grund, dass keine DMA verwendet wird und das Mikrofon vom Standby-Betrieb (20 μA) in den Performance-Modus (580 μA) wechselt.

$$Leistung_{Mikrofon} \approx \underbrace{580\mu A \cdot 120ms}_{Performance-Modus} + \underbrace{20\mu A \cdot 880ms}_{Standby} = 87.2\mu Ah \quad (9)$$

EEPROM
Der Chip weist eine vielzahl von unterschiedlichen Leistungsmodi auf. Die Energieaufnahme variiert zwischen dem niedrigen Standby-Modus ($35\mu Ah$)

und dem schreiben einer Seite ($3mAh$) von 512 Bytes. Gemäss folgender Mischrechnung (10) kann die ungefähre Energieaufnahme approximiert werden.

$$\approx \underbrace{3mA \cdot 4.5ms}_{Schreiben} + \underbrace{1.5mA \cdot 2ms}_{Lesen} + \underbrace{35\mu A \cdot 993.5ms}_{Standby} = 51.3\mu Ah^{7} \tag{10}$$

Übersicht berechnet

Bauteil	**Strom**	**Bemerkung**
Mikrocontroller	2842 μAh	
LEDs	800 μAh	100μAh pro LED $\cdot$ 8 LEDs
LED-Treiber	600 μAh	Alle Ausgänge EIN
EEPROM	51.3 μAh	
Mikrofon	87.2 μAh	Low-Power-Modus
Total	4.4 mAh	
Total Arbeitstag à 12h	**52.6** mAh	

Tabelle 5: Berechnete Energieaufnahme des Gesamtsystems

Übersicht gemessen

Bauteil	**Strom**	**Bemerkung**
Mikrocontroller	1760 μAh	Alle Peripherien EIN
LEDs	780 μAh	Alle LEDs EIN
LED-Treiber	820 μAh	Alle Ausgänge EIN
EEPROM	70 μAh	
Mikrofon	103 μAh	Messintervall: 250ms
Total	3.5 mAh	
Total Arbeitstag à 12h	**42.40** mAh	

Tabelle 6: Gemessene Energieaufnahme des Gesamtsystems

Wie insbesondere der Tabelle 6 zu entnehmen ist, kann die Einsatzdauer von 12 Stunden mit einem 50mAh-Akku abgedeckt werden.

[7] Alle Zeiten gemäss Datenblatt

7.3 Mikrofon-Messungen

Das MEMS-Mikrofon, sowie das M2211 von NTI, wurden beide jeweils vor jeder Messung mittels dem CAL200 auf 94dBSPL kalibriert. Da die Abstände zwischen den Mikrofonen und CAL200 nicht identisch sind, wurden zudem noch eine Kalibrationsmessung durchgeführt. Genauere Angaben sind dem Anhang 11.2 sowie 11.4 zu entnehmen.

Messaufbau SPL

- Messfrequenz: 1 kHz
- Amplitude: 1 Vrms
- Signalform: Sinus
- Gewichtung: Z
- Distanz zu Signalquelle: 100 mm ... 800 mm
- Erfassung: Slow
- Visualisierung: 29

Messaufbau Frequenzgang

- Messfrequenz: 10 Hz ... 20 kHz
- Amplitude: 1 Vrms
- Signalform: Sinus
- Gewichtung: A, C, Z
- Erfassung: Slow
- Distanz zu Signalquelle: 200 mm
- Visualisierung: 28

Abbildung 24: Frequenz vs Schalldruckpegel-Plot der ungewichteten Messung | dB(Z)

Wie in Abbildung 24 ersichtlich, kann die Messung grob in drei Bereiche eingeteilt werden, wobei jeder Abschnitt eine konkrete Einzelheit aufweist.

- Bereich <80Hz

- Bereich >80Hz und <2kHz
  Dieser ist unproblematisch, da er die Kennlinie des Referenzmikrofons widerspiegelt. Die vorhandenen Nichtlinearitäten sind auf den verwendeten Messlautsprecher sowie den dazugehörigen Verstärker zurück zu führen.

- Bereich >2kHz
  Auch hier folgen die beiden Mikrofone der Nichtlinearität des Messaufbaus. Jedoch reagiert das MEMS-Mikrofon sensitiver auf diesen Frequenzbereich, als das Referenzprodukt. Dieses Phänomen ist auf zwei physikalische Effekte zurückzuführen:

  - Helmholtz Effekt
    Dieser tritt, bauartbedingt, ab einer gewissen Frequenz bei allen MEMS-Mikrofonen auf. Ersichtlich ist dieser in [10], Figure 4. Dabei schwingt das gesamte Mikrofon mit der Membrane mit und erzeugt somit einen höheren Schalldruck.
  -

Abbildung 25: Frequenz vs Schalldruckpegel-Plot der gewichteten Messung | dB(A)

Abbildung 26: Frequenz vs Schalldruckpegel-Plot der gewichteten Messung | dB(C)

Abbildung 27: Distanz vs Schalldruckpegel-Plot

7.4 Vergleich

TODO

8 Validierung der Anforderungen

TODO

9 Stolpersteine

Stolpersteine gab es in dieser Arbeit einige. Auf gewisse werde ich, nachfolgend genauer eingehen.

RTC
Der Einsatz der RTC hat auf zweierlei Arten zu Problemen geführt. Wie sich während der Arbeit herausgestellt hat, benötigt der BLE-Stack von Silicon Labs für das Zeitmanagement die RTC. Dadurch kann diese nicht frei und flexibel eingesetzt werden, sondern muss über spezielle Funktionen abgegriffen werden. Gleichzeitig setzt sich die RTC bei jedem Neustarts des Systems zurück. Dadurch kann kein anhaltender Zeitstempel, mit Referenz zur lokalen Zeit, erzeugt werden. Dies ist bei der aktuellen Implementation so angedacht, für ein fertiges Produkt jedoch nicht ideal.

BLE
Der BLE-Stack weist eine hohe Komplexität auf. Dadurch wird dieser grundsätzlich nicht selbst implementiert, sondern vom Hersteller des Mikrocontrollers zur Verfügung gestellt. Dies birgt jedoch die Gefahr, dass gewisse Peripherien einen Doppelnutzen erfahren wie dies beispielsweise auch bei der RTC der Fall ist. Genauso war es auch beim PLL, welcher zur Erzeugung des Mikrofon-Clocks benötigt wird, der Fall. Auch dieser muss mit dem BLE-Stack "geteilt" werden, was die Implementation erschwert hat.

Mikrofon
Mikrofone mit digitalem Ausgang sollten grundsätzlich einfacher in der Handhabung sein als jene mit analogen. Dies konnte ich jedoch nicht feststellen. Unklarheiten im Bezug zum Ausgangssignal des Mikrofons, sowie des anschliessenden PCM-Peripherieblocks haben die Arbeit erheblich erschwert. Ich bin auch nach Abschluss dieser Arbeit noch nicht komplett davon überzeugt, dass das ausgegebene PCM-Signal einer vorzeichenbehafteten 16-Bit-Zahl (int16_t) entspricht. Dies aus dem einfachen Grund, dass die mitgelieferte HAL-Funktion, gemäss Implementation, einen unit32-Wert zurückliefert. Rückfragen an den Hersteller blieben unbeantwortet.

10 Fazit und Ausblick

Mit dieser Arbeit habe ich ein Produkt geschaffen, welches zwar nicht ganz an ein NTI XL2 herankommt, jedoch in seiner Preiskategorie durchaus seinen Platz verdient hat. Der Prototyp erfüllt alle in 11.9 gewünschten Anforderungen, ist soweit einsatzbereit und kann in einer nachfolgenden Arbeit weiterentwickelt werden. In dieser sollten folgende Themenbereiche behandelt werden:

- BLE
  Die Hardware ist soweit für die Datenübertragung vorbereitet. Nun soll die Datenübertragung mittels zugehörigem BLE-Stack implementiert werden.

- Interface
  Die Messdaten befinden sich momentan nur auf dem EEPROM. In Verbindung mit der BLE-Implementation ist eine visuelle Applikation auf dem Zielgerät, z.B. Smartphone, zu implementieren. Neben dem herunterladen der Messdaten können so auch Einstellungen wie LED-Helligkeit, Mess-Intervall oder Aufteilung der Werte auf die LEDs auf dem Messgerät vorgenommen werden.

- RTC
  Die momentane Implementation ermöglicht es nicht, einen Zeitstempel im UNIX-Format den Messdaten anzuhängen. Dies aus dem einfachen Grund, dass bei einem leeren Akku diese Zeit nicht weiter gezählt wird. Als Alternative könnte der initiale Zeitstempel via BLE auf das Messgerät übertragen und für die neue Messung verwendet werden.

- Low-Power
  Bei der Entwicklung wurde ein besonders Augenmerk auf die Energieeffizienz gelegt. Nichts desto trotz, kann das System noch energieeffizienter ausgelegt werden. Insbesondere ein Mikrocontroller mit integrierter BiQuad-Hardware würde die hohe Berechnungszeit 4 reduzieren.

- Bluetooth-Zertifizierung
  Gemäss Spezifikation von Bluetooth, ist jedes Gerät zu zertifiziert ([12]). Dementsprechend auch Geräte, welche aus bereits zertifizierten Geräten zusammengestellt werden. Ohne Zertifizierung darf ein Gerät nicht verkauft werden. Die Zertifizierung ist jedoch teuer([13]).

Danksagung

Als aller erstes möchte ich mich bei bei meinem Industriepartner, hEar, bedanken, ohne welchen das Projekt niemals in meine Hände gelangt wäre. Gleichzeitig möchte ich mich auch bei meinen beiden Betreuern, Filippo Parisi und Patric Eberle, für die Unterstützung während des Semesters bedanken. Ein besondere Dank gilt zudem folgenden Personen:

- Pascal Jund und Jürgen Wassner für die tatkräftige Unterstützung bei der Implementation von zeitdiskreten Filtern auf einer Embedded-Plattform.
- Manuel Isenegger für die Unterstützung bei Akustikthemen aller Art.
- Katharina Wick für die Korrektur, sowie der konstruktiven Kritik meiner gesamten Arbeit.
- Meinen Kommiliton:innen für die unzähligen konstruktiven Diskussionen und Möglichkeiten des Austausches im BAT-Raum.

11 Anhang

References

[1] "XL2 Schallpegelmesser & Akustik-Analysator | NTi Audio." [Online]. Available: https://www.nti-audio.com/de/produkte/schallpegelmesser/xl2-audio-akustik-analysator

[2] G. Feiertag, M. Winter, and A. Leidl, "Flip chip packaging for MEMS microphones," *Microsystem Technologies*, vol. 16, pp. 817–823, May 2010.

[3] "Peripherer Beschleunigungssensor." [Online]. Available: https://www.bosch-mobility.com/de/loesungen/sensoren/peripherer-beschleunigungssensor/

[4] N. R. C. U. C. o. D. D. f. I. w. H. Impairments, R. A. Dobie, and S. V. Hemel, "Basics of Sound, the Ear, and Hearing," in *Hearing Loss: Determining Eligibility for Social Security Benefits*. National Academies Press (US), 2004. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK207834/

[5] "Frequenzbewertung," Oct. 2023, page Version ID: 238201232. [Online]. Available: https://de.wikipedia.org/w/index.php?title=Frequenzbewertung&oldid=238201232

[6] G. Dighe, "TPS65950 Real-Time Clock Timing Compensation," Oct. 2008. [Online]. Available: https://www.ti.com/lit/an/swca024/swca024.pdf

[7] "STM32L4 Series - PDF Documentation," Mar. 2024. [Online]. Available: https://www.st.com/en/microcontrollers-microprocessors/stm32l4-series/documentation.html

[8] "V-Lambda-Kurve," Dec. 2023, page Version ID: 240001066. [Online]. Available: https://de.wikipedia.org/w/index.php?title=V-Lambda-Kurve&oldid=240001066

[9] A. V. Oppenheim, *Zeitdiskrete Signalverarbeitung*. R. Oldenbourg, 1989.

[10] "https://invensense.tdk.com/wp-content/uploads/2020/07/DS-000157-ICS-41351-v1.4.pdf." [Online]. Available: https://invensense.tdk.com/wp-content/uploads/2020/07/DS-000157-ICS-41351-v1.4.pdf

[11] L. Silicon, "EFR32xG22 Wireless Gecko Reference Manual," 2022. [Online]. Available: https://www.silabs.com/documents/public/reference-manuals/efr32xg22-rm.pdf

[12] "Bluetooth Zertifizierung." [Online]. Available: https://www.bluetooth.com/develop-with-bluetooth/qualification-listing/

[13] "Bluetooth Zertifizierungskosten." [Online]. Available: https://www.bluetooth.com/fee-schedule/

Abbildungsverzeichnis

1 XL2 von NTI Audio | Quelle: [1] 6
2 Unterschied MEMS-Mikrofone | Quelle: [2] 7
3 Beispiel MEMS-Beschleunigungssensor | Quelle: [3] 7
5 Beispiel PDM-Signal 9
6 Beispiel PCM-Signal 9
7 Grundbegriffe Akustik | Beispiel: ICS-41351 10
8 Min. benötigte Auflösung in Bit für gewünschten max. SPL . 11
9 Phon-Kurven im Vergleich | Quelle: [4] 12
10 Frequenzbewertungsfilter | Quelle: [5] 12
11 Vergleich Low-Power-Modi STM | Quelle: [7] 15
12 Vergleich von möglichen Mikrocontrollern 17
13 Sensitivität des menschlichen Auges auf das Lichtspektrum bei Tag (**schwarz**) und Nacht (**blau**)| Quelle: [8] 18
14 Buck Converter 19
15 Konzept PCB-Layout 22
16 Konzept Softwareablauf 22
17 Verpolungsschutz mittels P-Kanal MOSFET 25
18 Übersicht zu (LE)Timer-Funktionalität 27
19 Unterschied Zeit- und Frequenzbereich zur Frequenzgewichtung 29
20 Übersicht BiQuad-Filter 30
21 Clock-Muxing für 3.072MHz PDM 31
22 Softwareablauf I^2C 32
23 Aufteilung dBSPL auf 8 LEDs 33
24 Frequenz vs Schalldruckpegel-Plot der ungewichteten Messung | dB(Z) 38
25 Frequenz vs Schalldruckpegel-Plot der gewichteten Messung | dB(A) 39
26 Frequenz vs Schalldruckpegel-Plot der gewichteten Messung | dB(C) 40
27 Distanz vs Schalldruckpegel-Plot 40

11.1 Fremdwörter & Abkürzungen

Abkürzung	Beschreibung
BLE	Bluetooth Low Energy
BMS	Battery Management System | Akku-Laderegelung
CRC	Cyclic Redundancy Check
Embedded	In das Gerät integriert
FFT	Fast Fourier Transformation
FIR	Finite Impulse Response
Flashen	Software auf Mikrocontroller laden
I2C	Inter-Integrated Circuit (Bus-System)
IC	Integrated Circuit | Integrierter Schaltkreis
IFFT	Inverse Fast Fourier Transformation
IIR	Infinite Impulse Response
Li-Ion	Lithium-Ionen
Li-Po	Lithium-Polymer
Peripherie	Gruppe von Funktionen um z.B. das PCM-Signal zu generieren
Pins	Physikalischer Anschluss am Mikrocontroller
PLL	Phase Locked Loop
ppm	Parts Per Million
PPTC	Polymeric Positive Temperature Coefficient
Register	Schnellste Speichereinheit im Mikrocontroller
RTOS	Real Time Operating System | Betriebssystem auf Mikrocontroller
SPI	Serial Peripheral Interface (Bus-System)
THD	Total Harmonic Distortion | Klirrfaktor
UNIX-Format	Zeitformat gezählt ab 01.01.1970
WLAN	Wireless Local Area Network

11.2 Messmittel

Farblich gekennzeichnete Geräte und Software gehören zusammen.

- Saleae Logic Analyzer - Logic 8
  S/N: T&A 36 093
- Logic 2
  Version: 2.4.14
- Audio Precision 2700
  S/N: SYS2-31822
- AP2700 Control Software
  Version: 3.30 (Build 118)
- MCUXpresso IDE
  Version: 11.8.1 [Build 1197] [2023-10-27]
- NXP MCU-Link Pro CMSIS-DAP
  Version: V3.108
  S/N: FDS0PNBVSQUQZ
- NTI XL2
  S/N: A2A-17216-E0
- M2211 Messmikrofon
  S/N: 7579
- Larson Davis CAL200 (1kHz / 94dBSPL Referenzquelle)
  S/N: 17902
- XL2 Projector PRO
  Version: 1.54
- Simplicity Studio
  Version: SV5.8.1.1
- Gecko SDK Suite
  Version: 4.4.1

11.3 Messaufbau

(a) MEMS hinten

(b) NTI hinten

(c) MEMS Vogelperspektive

(d) NTI Vogelperspektive

Abbildung 28: Messaufbau Frequenzgang

(a) Messobjekt hinten

(b) Messobjekt Vogelperspektive

(c) Reflexionsarme Messkammer

Abbildung 29: Messaufbau SPL

11.4 Kalibrationsmessung

Abbildung 30: Kalibrationsmessung über die Distanz

Distanz [mm]	**MEMS**	**Referenz**	**Delta**
100	83.7	87.1	3.4
200	78	82.7	4.7
300	76.3	80.6	4.3
400	74.5	79.9	5.4
500	73.1	78.2	5.1
600	70.8	77.5	6.7
700	70.7	77.1	6.4
800	69.1	75.5	6.4

11.5 PCB-Versionen

Abbildung 31: V1-0 Front

Abbildung 32: V1-0 Back

Abbildung 33: V1-1 Front

Abbildung 34: V1-1 Back

Abbildung 35: V1-2 Front

Abbildung 36: V1-2 Back

Abbildung 37: V1-3 Front

Abbildung 38: V1-3 Back

11.6 Schema

USB-C & Debug & Programming
Microcontroller
LED
EEPROM
Microphone
Power
Mechanical
Testpoints
Auxiliary components
Hochschule Luzern, Technik & Architektur
Title: Schalldruckpegelmessgerät
Size: A4
Date: 2024-05-04
KiCad E.D.A. 8.0.0
Rev: 1.2
Id: 1/1

11.7 Software

https://github.com/Froster2406/BAT2024

Oder alternativ im digitalen Anhang unter:

Entwicklung/Software/custom

11.8 Aufgabenstellung

Horw, 19. Februar 2024
Seite 1/2

Bachelor Thesis im Studiengang Elektrotechnik und Informationstechnologie

Aufgabe für Herrn Nicora Stefano

Entwicklung einer PCB zur Analyse von Umgebungslärm

Fachliche Schwerpunkte
Embedded Systems, Akustik, PCB-Design, Signalverarbeitung

Einleitung
hEar ist ein Start-up im Bereich Design, Akustik und Prävention von Problemen im Zusammenhang mit dem Gehör. Nach ersten Machbarkeitstests streben wir die Entwicklung eines Geräts an, das kontinuierlich den Geräuschpegel während des gesamten Tages überwacht. Durch ein System von Lichtindikatoren wird sofortiges Feedback zur Umgebung gegeben, um ein höheres Bewusstsein zu fördern und präventive Massnahmen für die Gesundheit des Gehörs zu unterstützen.

Aufgabenstellung
- Einarbeitung in die Akustik
- Analyse des Stands der Technik und der Wissenschaft
- Vertiefung der Techniken der Schallmessung und -analyse
- Kritische Prüfung von MEMS-Mikrofonen, Mikrocontrollern und LEDs zur Optimierung des Gleichgewichts zwischen Leistung und Energieeffizienz
- Entwurf und Herstellung von Prototypen von Leiterplatten (PCB)
- Softwareentwicklung für die Schallanalyse
- Durchführung von Tests und Überprüfung der Systemleistung durch Vergleich mit den auf dem Markt befindlichen Produkten
- Analyse des Schalls in verschiedenen Situationen und Bestimmung des Risikos für das menschliche Gehör
- Dokumentation in einem Bericht
- Die genauen Aufgaben werden zu Beginn des Projekts besprochen

Termine

Start der Arbeit:	Montag 19.2.2024
Zwischenpräsentation:	Zu vereinbaren im Zeitraum 8.4. – 3.5.2024
Abgabe Schlussbericht:	Freitag 7. Juni 2024, vor 16:00 im D311
Abgabe Digitale Doku:	Gemäss separater Anweisung der Studiengangleitung
Abschlusspräsentation:	Zu vereinbaren im Zeitraum 10.6. – 28.6.2024
Diplomausstellung:	Mittwoch 3. Juli 2024 (Teilnahme obligatorisch!)

Dokumentation

Der gebundene Schlussbericht enthält keine Selbständigkeitserklärung und ist in einfacher Ausführung zu erstellen. Er enthält zudem zwingend

- einen sehr kurzen, englischen Abstract.
- Ein Titelblatt, gleich hinter dem Deckblatt, gemäss Weisungen der Studiengangleitung
- Eine SD-Hülle, innen, auf der Rückseite des Berichtes für den Betreuer

Alle Exemplare des Schlussberichtes müssen komplett und termingerecht gemäss Angaben der Studiengangleitung abgeben werden. Zusätzlich muss eine SD-Speicherkarte mit dem Bericht (inkl. Anhänge), dem Poster und den Präsentationen, Messdaten, Programmen, Auswertungen, usw. unmittelbar nach der Präsentation abgeben werden.
Die gesamte Dokumentation ist zudem gemäss Anweisungen der Studiengangleitung elektronisch auf einen Server zu laden. Sämtliche abzugebende Teile der Dokumentation sind Bestandteile der Beurteilung.

Fachliteratur/Web-Links/Hilfsmittel

Wird bei Projektstart definiert

Geheimhaltungsstufe: Vertraulich

Verantwortlicher Dozent/Betreuungsteam, Industriepartner

Dozent	Prof. Dr. Patric Eberle	patric.eberle@hslu.ch
Industriepartner	hEar St.Anna Strasse 38 6006 Luzern Fr. S. Willener sophie.willener@bluewin.ch Tel. +41 79 856 14 81	
Experte	Hr. Reto Jäggi reto.jaeggi@ch.mullermartini.com	

Hochschule Luzern
Technik & Architektur

Prof. Dr. Patric Eberle

11.9 Anforderungskatalog

Nr.	M/K/N	Thema	Beschreibung
1	M	Speisung	Das Gerät verfügt über einen wiederaufladbaren Akku. Dieser wird mittels USB-C-Buchse aufgeladen.
2	M	Formfaktor	Die Bauform des PCB ist rund. Die genauen Dimensionen werden mit hEar definiert.
3	M	Laufzeit	Die Laufzeit soll mindestens 12 Stunden betragen. Dies deckt einen gängigen Arbeitstag ab.
4	M	Kommunikation	Das Gerät weist eine Bluetooth low energy (BLE) Schnittstelle auf. Durch diese werden die Messdaten an ein Empfangsgerät übertragen. Die Implementation dieser ist jedoch nicht Teil der Arbeit.
5	M	Messdaten-Logging	Die Messdaten werden in einem noch zu definierendem Zeitintervall lokal gespeichert.
6	M	Zeitstempel	Die Messdaten sollen über einen Zeitstempel verfügen.
7	M	Visuelle Anzeige	Der gemessene Schalldruckpegel wird visuell auf der Front des Gerätes dargestellt.
8	M	Kosten	Die Gesamtkosten entsprechen maximal 50 CHF.
9	M	Mikrofon	Der Schalldruckpegel soll mittels eines MEMS-Mikrofons aufgenommen werden.
10	K	LEDs	Die visuelle Darstellen des Schalldruckpegels erfolgt über LEDs auf der Front des Gerätes.
11	N	Bauteile	Die Bauteile sollen von Hand lötbar sein. Gleichzeitig sollen die Platinen maschinell bestückt werden können.

M: Muss, K: Kann, N: Nice to have

11.10 Risikoanalyse

Risikoanalyse

Risiko	Beschreibung	W'keit	Auswirkung	Auswirkung	Massnahmen	Risikowert
Nicht genügend Akku-Nennladung	Die Energieaufnahme der gesamten Schaltung übersteigt die mögliche Energieleistungsdichte der kommerziell erhältlichen Akkus	3	Das Produkt kann im gewünschten Rahmen nicht umgesetzt werden	4	Hauptursache finden und mit Alternativen lösen	12
Leistungsaufnahme der LEDs	Die Leistungsaufnahme der LEDs reduziert die Einsatzdauer des Produktes	3	Die Mindestlaufzeit des Gerätes kann nicht eingehalten werden	4	Alternativen in der Evaluationsphase finden	12
Gerät kann nicht programmiert werden	Die Programmierschnittstelle kann nicht verwendet werden	2	Die Software kann nicht auf das Gerät geladen werden	4	Genaue Prüfung der Kriterien bei der Auswahl	8
PCB wird nicht rechtzeitig bestellt / geliefert	Infolge Lieferschwierigkeiten wird das PCB erst viel später geliefert als geplant	2	Der Projektplan kann nicht eingehalten werden	4	PCB wird so entwickelt, dass dieses auch in der Schweiz produziert werden kann (mit Aufpreis)	8
Messgeräte defekt	Messungen der Energieeffizienz oder des Schalldruckpegels können nicht durchgeführt werden, da die Messgeräte defekt sind	2	Der Projektplan kann nicht eingehalten werden	3	Alternative Messgeräte eruieren	6
PCB-Layout ist nicht rechtzeitig fertig	Das PCB-Layout wird nicht bis zum geplanten Termin fertig	2	Der Projektplan kann nicht eingehalten werden	3	Spielraum bereits im Projektplan einberechnen	6
Kein passendes MEMS-Mikrofon	Die auf dem Markt vorhandenen Mikrofone erfüllen die gewünschten Anforderungen nicht	1	Alternativen zu MEMS-Mikrofonen müssen geprüft werden	4	Abklärung der Anforderungen bereits bei der Recherche	4
Halbleiterknappheit	Die benötigten Bauteile sind nicht mehr lieferbar	1	Das Produkt kann nicht fertig gebaut und getestet werden	4	Mehrere Lieferanten für das selbe Bauteil, Standard-Bauteile einsetzten	4
Krankheitsbedingter Ausfall	Die Arbeit muss krankeitsbedingt unterbrochen werden	1	Der Projektplan kann nicht eingehalten werden	4	Spielraum bereits im Projektplan einberechnen	4
Kostendach überstiegen	Die Anschaffung der Bauteile übersteigt das vorgegebene Kostendach	1	Die Arbeit könnte im geplanten Massstab nicht durchgeführt werden	3	Detailierte Kostenaufstellung zu Beginn des Projektes	3
Der falsche Mikrocontroller wurde eingesetzt	Für den Einsatzzweck wurde der falsche Mikrocontroller eingesetzt	1	Das Projekt kann nicht im beschriebenen Rahmen durchgeführt werden	3	Genaue Prüfung der Kriterien bei der Auswahl	3

W'keit: 1 = Tief, 3 = Hoch

Auswirkung: 1 = Tief, 4 = Hoch

Risikowert: 1 = minimal, 12 = maximal

11.11 Zeitplan

Zeitplan

	19.02 SW 01	26.02 SW 02	04.03 SW 03	11.03 SW 04	18.03 SW 05	25.03 SW 06	01.04 SW 07	08.04 SW 08	15.04 SW 09	22.04 SW 10	29.05 SW 11	06.05 SW 12	13.05 SW 13	20.05 SW 14	27.05 SW15	03.06 -	
						Ostern											
Vorbereitung, Recherche und Doku																	
Kickoff-Meeting	19.02																
Recherchen																	
Evaluation Microcontroller																	
Evaluation MEMS Mikrofon																	
Evaluation visuelle Anzeige																	
Zeitplan & Anforderungsliste erstellt		26.02															
Dokumentation weiterführen																	
Dokumentation beendet																02.06	
Prototyping & Testing																	
Arbeiten mit dem Evaluationsboard																	
Ausarbeitung des Konzeptes																	
Entwicklung PCB																	
Entwicklung Software																	
Zwischenpräsentation vorbereiten																	
Zwischenpräsentation										26.04							
Finalisierung																	
Optimierung des Codes																	
Bereinigung der Codebase																	
Reserve																	
Vorbereitung / Abgabe Plakat																	
Schlussabgabe Dokumentation																07.06	
Abschlusspräsentation vorbereiten																	
Abschlusspräsentation																	10.06-28.06

Projekt-Meilensteine

Projektstart	19.02.2024
Zwischenpräsentation	TBD
Abgabe Schlussbericht	**07.06.2024**
Abschlusspräsentation	10.6.-28.6.
Diplomausstellung	03.07.2024

kritische Tätigkeiten sind rot hervorgehoben

11.12 Journal