

**Konzeption und Entwicklung eines AVAS für
rein elektrisch betriebene Fahrzeuge der
Fahrzeugklasse L1 & L3e**

Masterarbeit

im Studiengang

Audiovisuelle Medien im Schwerpunkt Ton

vorgelegt von

Bence Böröcz

Matr.-Nr.: 34898

am 22. November 2019

an der Hochschule der Medien Stuttgart

Erstprüfer/in: Prof. Oliver Curdt

Zweitprüfer/in: Prof. Jens-Helge Hergesell

Ehrenwörtliche Erklärung

Hiermit versichere ich, Bence Böröcz, ehrenwörtlich, dass ich die vorliegende Masterarbeit mit dem Titel: „Konzeption und Entwicklung eines AVAS für rein elektrisch betriebene Fahrzeuge der Fahrzeugklasse L1 & L3e“ selbstständig und ohne fremde Hilfe verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen wurden, sind in jedem Fall unter Angabe der Quelle kenntlich gemacht. Die Arbeit ist noch nicht veröffentlicht oder in anderer Form als Prüfungsleistung vorgelegt worden.

Ich habe die Bedeutung der ehrenwörtlichen Versicherung und die prüfungsrechtlichen Folgen (§ 26 Abs. 2 Bachelor-SPO (6 Semester), § 24 Abs. 2 Bachelor-SPO (7 Semester), § 23 Abs. 2 Master-SPO (3 Semester) bzw. § 19 Abs. 2 Master-SPO (4 Semester und berufsbegleitend) der HdM) einer unrichtigen oder unvollständigen ehrenwörtlichen Versicherung zur Kenntnis genommen.

Stuttgart, 21. November 2019

Kurzfassung

Diese Arbeit befasst sich mit der Entwicklung eines prototypischen „Acoustic Vehicle Alerting System“ (AVAS) für leichte elektrisch betriebene Fahrzeuge. Um die Sicherheit im Straßenverkehr, bei steigender Anzahl von Elektroautos, gewährleisten zu können sind AVAS seit 2019 Pflicht für alle neu angemeldeten PKW mit Elektroantrieb. Jedoch sind elektrisch betriebene Zweiräder davon ausgeschlossen. Hinsichtlich der zunehmenden Sharing-Anbieter und der rapiden Entwicklung im asiatischen Markt für Elektromotoren bis 45 km/h, zeigt diese Arbeit Lösungsansätze für ein akustisches Warnsystem für die entsprechende Fahrzeugklasse. Hierbei wird neben den technischen Anforderungen, ebenso die klangliche Gestaltung erforscht und erläutert.

Abstract

This paper deals with the development of a prototype for an “Acoustic Vehicle Alerting System” (AVAS) for light electro mobility. To ensure safety in traffic regardless of the rising numbers of electric vehicles AVAS for cars are obligatory since mid-2019 for every newly registered car with electric drive. However electrical driven two-wheelers are excluded of that rule. Regarding the rapid growth of sharing providers and the ever-growing Asian market for e-scooters till 45 km/h this paper provides solutions for acoustic alerting systems for the corresponding vehicle type. In this connection the technical requirements as well as the sound design process are researched and explained.

Inhaltsverzeichnis

Ehrenwörtliche Erklärung	2
Kurzfassung	3
Abstract	3
Inhaltsverzeichnis	4
Abbildungsverzeichnis	6
Tabellenverzeichnis	7
Abkürzungsverzeichnis	8
1 Einleitung und Problembeschreibung	9
2 Gesetzliche Grundlage des AVAS	10
2.1 Gesetzliche Vorgaben in der Europäischen Union	10
2.2 Innerhalb der Vereinigten Staaten von Amerika.....	13
3 Anforderungs- und Marktanalyse	15
3.1 Schallwahrnehmung des Menschen.....	15
3.1.1 Hörphysiologie.....	15
3.1.2 Psychoakustik und Hörpsychologie	17
3.2 Aktueller Stand der Technik	20
3.3 Spektrale und klangliche Analyse bestehender AVAS.....	21
3.4 Umfrage zu AVAS.....	25
4 Annäherung an die Problemstellung	31
4.1 Motivation	31
4.2 Use Cases	31
5 Technische Umsetzung des AVAS	33
5.1 Controller Area Network	33
5.2 Konzeption der Hard- und Softwarekomponenten.....	36
5.3 Herausforderungen und Probleme	38
5.4 Aufbau des Prototyps	40
5.4.1 Hardware	40
5.4.2 Software.....	43
6 Erstellung der Soundpatterns	50

6.1	Grundlagen der elektronischen Klangerzeugung	50
6.1.1	Grundelemente der Klangerzeugung	50
6.1.2	Wichtige Verfahren der elektronischen Klangerzeugung	51
6.2	Fahrgeräusch	53
6.3	„Friendly Horn“	64
6.4	Portierung auf ein eingebettetes System.....	67
7	Nutzerstudie.....	70
7.1	Vorgehen und Methodik	71
7.2	Ergebnisse.....	73
8	Fazit	76
Anhang A: Datenträger		77
Anhang B: „Proof of Concept – Sheet“		78
Anhang C.1: Automationskurven der Geschwindigkeit -Soundpattern 1.....		79
Anhang C.2: FMOD Profiler.....		80
Quellenverzeichnis.....		81

Abbildungsverzeichnis

Abbildung 1 Spektrum der AVAS Empfehlung der UN (Frequenz über Zeit)	12
Abbildung 2 Spektrum eines auf dem Klavier gespielten f.....	13
Abbildung 3 Querschnitt des menschlichen Ohrs	15
Abbildung 4 Menschliche Hörfläche	17
Abbildung 5 Hörschwelle und Isophone	18
Abbildung 6 AVA-System von Grewus	20
Abbildung 7 Spektrum des Fahrgeräuschs eines Citroen C-Zero	22
Abbildung 8 Spektrum des Fahrgeräuschs eines Renault Twizy	23
Abbildung 9 Spektrum des Fahrgeräuschs des BMW i8.....	24
Abbildung 10 Controller Area Network	33
Abbildung 11 CAN Pinbelegung auf D-SUB 9	34
Abbildung 12 Unterteilung eines CAN-Frames.....	35
Abbildung 13 Blockschaltbild des Signalflusses im AVAS	40
Abbildung 14 Anlegen der Event- und Parameterinstanzen	43
Abbildung 15 Aufrufen der Event- und Parameterinstanzen.....	44
Abbildung 16 Fehlermeldung und Events an Gameobject anfügen.....	44
Abbildung 17 Update und „OnDestroy“ Methoden	45
Abbildung 18 Definieren der notwendigen Variablen.....	46
Abbildung 19 Geschwindigkeit und Fahr-Modi	47
Abbildung 20 Fahrtrichtung und Fernsteuerung	48
Abbildung 21 Fahrpedalstellung	49
Abbildung 22 Blockdiagramm des Sounddesign Workflows	53
Abbildung 23 Einstellungen des Software Synthesizers.....	54
Abbildung 24 Automationskurven der Gaspedalstellung – Soundpattern 1	56
Abbildung 25 Oszillator des zweiten Soundpatterns.....	57
Abbildung 26 Automationskurven der Geschwindigkeit – Soundpattern 2.....	58
Abbildung 27 Automationskurven der Gaspedalstellung – Soundpattern 2	59
Abbildung 28 Oszillator des Grundgeräuschs für das zweite Soundpattern.....	60
Abbildung 29 Envelope Filter E1	61
Abbildung 30 Wellenform bei Envelope Maximum	61
Abbildung 31 Automationskurven der Geschwindigkeit – Soundpattern 3.....	62
Abbildung 32 Automationskurven der Gaspedalstellung – Soundpattern 3	63
Abbildung 33 Summenbearbeitung in FMOD.....	64

Abbildung 34 Softwareoberfläche von Kontakt mit geladenem Kalimba	65
Abbildung 35 Equalizer-Einstellungen der zweiten "freundlichen Hupe"	66

Tabellenverzeichnis

Tabelle 1 Anforderungen an den Mindestschallpegel.....	11
Tabelle 2 Auto-, Roller-, Motorradfahrer/innen.....	25
Tabelle 3 (links) Soll das Elektrofahrzeug nach einem klassischen Verbrennungsmotor klingen? (rechts) Soll das Elektrofahrzeug nach künstlich generierten Geräuschen klingen?	26
Tabelle 4 Vergleich der Adjektive die einen auffälligen bzw. unauffälligen Klang beschreiben.....	28

Abkürzungsverzeichnis

APK	Android Application Package
AVAS	Acoustic Vehicle Alerting System
CAN	Controller Area Network
CRC	Cyclic Redundancy Check
DAW	Digital Audio Workstation
EV	Electric Vehicle
HMI	Human-Machine Interface
LEM	Leichtelektromobil
MIDI	Musical Instrument Digital Interface
NHTSA	National Highway Traffic Safety Administration
OEM	Original Equipment Manufacturer
RTR	Remote Transmission Request
UN	Die Vereinten Nationen
UNECE	Die Wirtschaftskommission für Europa der Vereinten Nationen
QRTV	Quiet Road Transport Vehicles

1 Einleitung und Problembeschreibung

Der städtische Verkehr mit leichten Elektrofahrzeugen wird in Zukunft an Bedeutung gewinnen. Der urbane Lieferverkehr und Sharing-Anbieter nehmen stetig zu. Die Preise der E-Scooter werden durch den asiatischen Markt gedrückt, sodass die elektrisch betriebene Fortbewegung erschwinglicher wird. Rund 26 Millionen Elektroroller werden in China jährlich verkauft und auch in Deutschland steigen die Verkaufszahlen.¹ Durch die Lautlosigkeit der Elektroantriebe im niedrigen Geschwindigkeitsbereich ist die auditive Wahrnehmung für andere Verkehrsteilnehmer stark erschwert.

In Zusammenarbeit mit der Robert Bosch GmbH wurde in dieser Abschlussarbeit ein Prototyp für ein „Acoustic Vehicle Alerting System“ (AVAS) für elektrisch betriebene Roller der Fahrzeugklasse L1 & L3e entwickelt.

Im Zuge dessen werden die technischen Anforderungen überprüft und gleichzeitig die Erwartungen an ein solches System erforscht. Bei einem AVAS handelt es sich in der Regel um einen Fahrgeräuschgenerator für elektrisch betriebene Fahrzeuge. So dient das AVAS nicht nur zur besseren Erkennung durch Fußgänger, sondern kann im nächsten Schritt auch ein Wiedererkennungsmerkmal für Hersteller sein.

Zunächst sollen die gesetzlichen Anforderungen analysiert werden, um einordnen zu können, wie das Geräusch beschaffen sein muss. Weiterhin werden die Anforderungen des Systems hinsichtlich der menschlichen Wahrnehmung detaillierter bestimmt. Um die Rahmenbedingungen noch besser eingrenzen zu können, wird anschließend eine Marktanalyse durchgeführt und im Zuge dessen die Erwartungen an ein AVAS erfragt und bestimmt.

In der zweiten Hälfte der Arbeit wird die technische Umsetzung des Prototyps erläutert. Zuerst wird dabei auf die Konzeption der Komponenten und Softwaretools eingegangen. Im Folgenden wird Schritt für Schritt aufgezeigt, wie die Hardwarekomponenten ausgesucht werden und wie die dazugehörige Software programmiert wird.

Schlussendlich werden drei Soundpatterns für das AVAS erstellt und schrittweise in ihrer Erstellung dargestellt. Um eine Aussage über die Qualität der Geräusche treffen zu können, wird zuletzt eine Nutzerstudie mit den finalen Soundpatterns durchgeführt.

¹ <https://www.n-tv.de/wirtschaft/Markt-fuer-E-Roller-waechst-weiter-article20492467.html>

2 Gesetzliche Grundlage des AVAS

2.1 Gesetzliche Vorgaben in der Europäischen Union

Bereits am 20. Februar 2014 schuf der Rat der europäischen Union die gesetzliche Grundlage für akustische Warnsysteme in Elektrofahrzeugen. Initiiert wurde dies durch eine informelle Arbeitsgruppe der Wirtschaftskommission für Europa der Vereinten Nationen (UNECE). Sie hat Richtlinien für „Quiet Road Transport Vehicles“ (QRTV), also geräuschlose Transportfahrzeuge, entwickelt. Im Zuge dessen wurde auch der Begriff AVAS eingeführt. Mithilfe der Richtlinien, die im Folgenden noch detaillierter erläutert werden, sollen Fahrzeughersteller einen Rahmen zur Erstellung ihrer eigenen Systeme bekommen. Innerhalb dieser Richtlinien bleibt den Herstellern genug Spielraum, um individuelle Klänge zu erstellen und so markenspezifische Fahrgeräusche zu etablieren. Für rein elektrisch fahrende Autotypen der Fahrzeugklasse M und N, die ab dem 1. Juli 2019 typgeprüft werden, ist das AVAS verpflichtend. Ab dem 1. Juli 2021 muss jedes neu angemeldete elektrisch betriebene Fahrzeug der entsprechenden Fahrzeugklassen mit einem Warnsystem ausgestattet sein. Es besteht also keine Pflicht ältere Modelle ab diesem Zeitpunkt nachzurüsten. Die Anforderungen für ein mit einem AVAS ausgestatteten Fahrzeug sind in dem entsprechenden EU-Beschlüssen beschrieben und im Folgenden zusammengefasst.^{2 3}

Das AVAS muss zwischen dem Anfahren und einer Geschwindigkeit von 20 km/h und beim Rückwärtsfahren automatisch ein Schallzeichen erzeugen. Bei Hybridfahrzeugen, die auch mit einem Verbrennungsmotor ausgestattet sind, darf das AVAS nur dann ein Schallzeichen erzeugen, wenn der Verbrennungsmotor nicht im angegebenen Geschwindigkeitsbereich in Betrieb ist. Fahrzeuge, die von sich aus einen Gesamtschallpegel entsprechend den Anforderungen der UNECE-Regelung Nr.138 wiedergeben, müssen nicht mit einem AVAS ausgestattet werden. Die Regelung definiert den Mindest-

² <https://eur-lex.europa.eu/legal-content/DE/TXT/PDF/?uri=CELEX%3A32017R1576&from=DE&fbclid=IwAR3REqOR4HkL77Dmang8-cahdg2UvdSZ-2ufYPIZ9V6YnAznyUY5dtKetAw>

³ <https://eur-lex.europa.eu/legal-content/DE/TXT/PDF/?uri=CELEX:42017X0071&from>

schallpegel für Fahrzeuge im Straßenverkehr anhand eines vorgegebenen Messverfahrens. Dabei wird der Schallpegel in Abhängigkeit vom Frequenzbereich gemessen. Der Mindestschallpegel ist also nicht über alle Terzbänder gleich laut und fällt bei etwa 1500 Hz stetig ab (Tab.1). Der in Tabelle 1 erwähnte Absatz 3.3.2 der UNECE Regelung Nr.138 definiert die Fahrtbedingungen während der Messung. Die Prüfgeschwindigkeit muss beispielsweise mindestens 5 Sekunden lang konstant eingehalten werden.

Anforderungen an den Mindestschallpegel in dB(A)

Frequenz in Hz		Prüfung mit konstanter Geschwindigkeit nach Absatz 3.3.2 (10 km/h)	Prüfung mit konstanter Geschwindigkeit nach Absatz 3.3.2 (20 km/h)	Prüfung bei Rückwärtsfahrt nach Absatz 3.3.3
Spalte 1	Spalte 2	Spalte 3	Spalte 4	Spalte 5
Insgesamt		50	56	47
Terzbänder	160	45	50	X
	200	44	49	
	250	43	48	
	315	44	49	
	400	45	50	
	500	45	50	
	630	46	51	
	800	46	51	
	1 000	46	51	
	1 250	46	51	
	1 600	44	49	
	2 000	42	47	
	2 500	39	44	
	3 150	36	41	
4 000	34	39		
5 000	31	36		

Tabelle 1 Anforderungen an den Mindestschallpegel⁴

⁴ https://eur-lex.europa.eu/legal-content/DE/TXT/PDF/?uri=CELEX%3A42017X0071&from&fbclid=I-wAR2SyDjk-Xk5liPJKJF_63Y8siS-Vfr7qhqshLnyuPm0bgOkUmtMK4borME

Das AVAS muss mit einem Mechanismus ausgestattet sein, welcher es erlaubt, das System temporär abzuschalten. Diese „Pausenfunktion“ muss für den Fahrer leicht zugänglich sein. Zudem muss bei einem Neustart des Fahrzeugs das AVAS standardmäßig eingeschaltet sein. Falls der Fahrer das System abschaltet, liegt es in seiner eigenen Verantwortung, andere Verkehrsteilnehmer nicht zu gefährden.

Das Schallzeichen des AVAS muss kontinuierlich im erforderlichen Geschwindigkeitsbereich in Betrieb sein. Dabei soll die Art des Klanges so gestaltet sein, dass auf ein herannahendes bzw. sich in Bewegung befindliches Fahrzeug hingewiesen wird. Der Geräuschpegel kann sich an ein Fahrzeug mit vergleichbarer Größe orientieren, sollte aber auf keinen Fall höher sein.

Neben den eindeutig definierbaren physikalischen Größen gibt es weitere Bestimmungen, die die Art des Signals definieren. Das AVAS darf beispielsweise keine Musik oder sonstige Klänge, die nicht auf ein herannahendes Fahrzeug schließen lassen, wiedergeben. Die Klangart sollte sich also im Entferntesten auch am Klang eines Verbrennungsmotors orientieren. Zur leichteren Umsetzbarkeit hat die UN ein Beispielgeräusch erstellt, dessen Spektrum in der folgenden Abbildung abgebildet ist. Es handelt sich dabei um den Anfahrvorgang eines fiktiven Elektrofahrzeugs. Das dazugehörige Audiofile befindet sich in Anhang A.1.

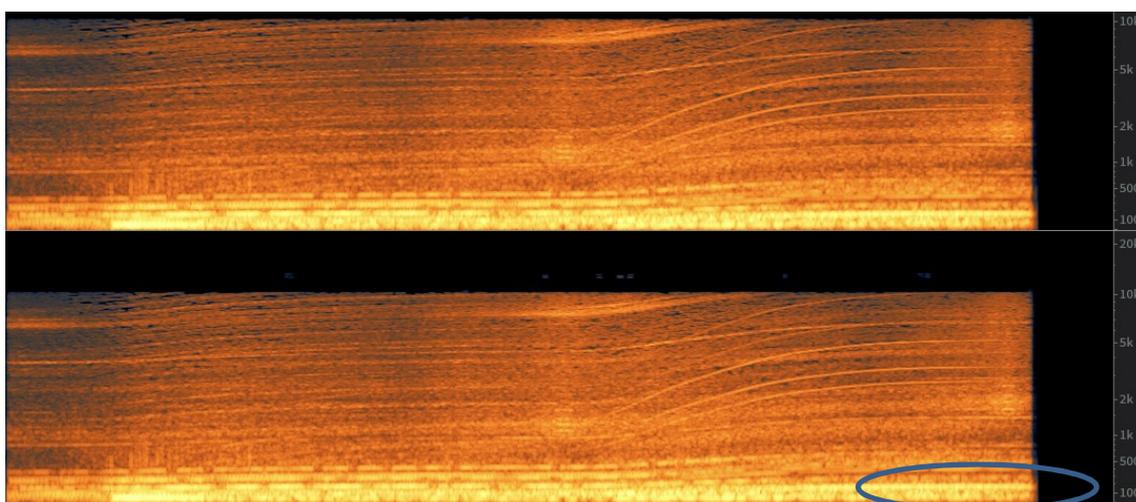


Abbildung 1 Spektrum der AVAS Empfehlung der UN (Frequenz über Zeit)⁵

⁵ <https://www.auto-motor-und-sport.de/tech-zukunft/sound-fuer-elektroautos/>

Im Spektrum des Beispielsignals ist gut zu erkennen, dass es sich besonders in der ersten Hälfte um ein breitbandiges Signal handelt. Kurz nach der Hälfte setzt ein etwas lauterer harmonischer Anteil ein. Dieser Anteil nähert sich der blau markierten Grundfrequenz bei etwa 170 Hz an, was nach der deutschen Notation einem f entspricht.

Auf diesem Ton aufbauend sind sehr gut dessen Harmonische zu erkennen. Die Ober-tonstruktur des Signals ähnelt dem eines Klaviers, welches ein f spielt.

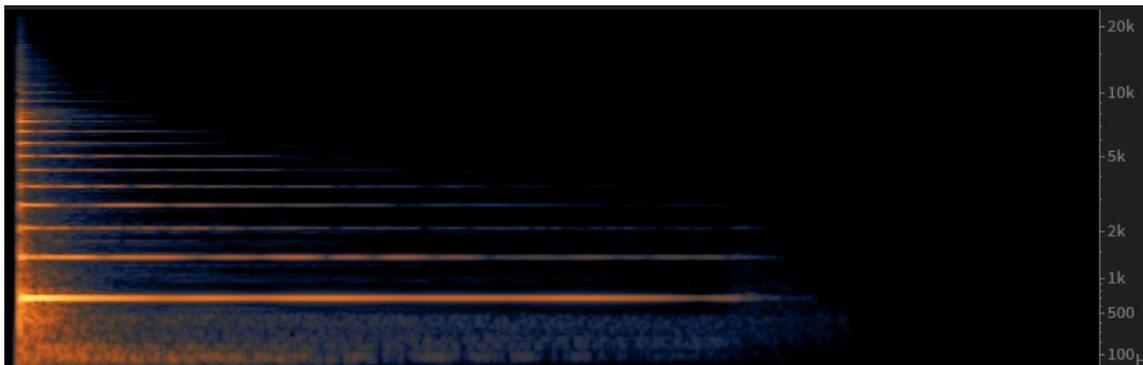


Abbildung 2 Spektrum eines auf dem Klavier gespielten f

Die Entscheidung, eine musikalische Obertonstruktur nachzubilden, hat einen psychologischen Hintergrund. Denn Töne aus der Musik, welche mit bekannten Instrumenten gespielt werden, werden unterbewusst als angenehm und vertraut wahrgenommen.

Anhand dieser Vielzahl an Vorgaben, haben die Hersteller bereits mehrere Anhaltspunkte für die Erstellung eines Geräusches für ihr AVAS.

2.2 Innerhalb der Vereinigten Staaten von Amerika

In den USA wurde bereits 2011 ein Gesetz verabschiedet, welches die Sicherheit von Fußgängern im Straßenverkehr hinsichtlich der geräuscharmen Elektro- und Hybridfahrzeuge, sicherstellen soll.⁶ Zu diesem Zeitpunkt wurde das Gesetz allerdings noch nicht weiter konkretisiert, so waren der Geschwindigkeitsbereich und die Lautstärke der aktiven Geräuscherzeugung nicht geregelt.

⁶ <https://web.archive.org/web/20101219073150/http://www.smartplanet.com/business/blog/smart-takes/hybrid-electric-vehicles-to-become-louder-for-pedestrian-safety/13077/>

In 2013 legte die „National Highway Traffic Safety Administration“ (NHTSA), als zuständige Behörde für diesen Bereich fest, dass bis zu einer Geschwindigkeit von 18,6 mph – was etwa 30 km/h entspricht – ein Fahrgeräusch aktiv wiedergegeben oder verstärkt werden muss. Die NHTSA einigte sich auf diesen Wert, da ab dieser Geschwindigkeit auch EVs durch Abrollgeräusche einen ähnlich hohen Schallpegel wie Verbrennungsmotoren erzeugen. So soll gewährleistet sein, dass Fußgänger die Fahrzeuge trotz der Umgebungsgeräusche zuverlässig wahrnehmen können. Die Anforderungen für die „Federal Motor Vehicle Safety Standards“ wurden sehr konkret in einem Dokument der NHTSA beschrieben.⁷

Letztlich ist auf der Grundlage dieser Reglementierungen nach mehrfachem Aufschub im Februar 2018 das Gesetz in den USA in Kraft getreten.

Im Vergleich der gesetzlichen Vorgaben der EU und der USA ist auffällig, dass sich die Geschwindigkeitsbegrenzungen trotz ähnlicher Argumentation um 10 km/h unterscheiden. Das hat zur Folge, dass sich Hersteller an der amerikanischen Grenze an 30 km/h orientieren, da es in der EU nicht verboten ist, auch jenseits der Geschwindigkeitsbegrenzung ein Geräusch wiederzugeben. Jedoch darf das Geräusch dann nicht lauter sein als ein vergleichbares Fahrzeug mit Verbrennungsmotor.

⁷ https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/documents/quietcar_finalrule_11142016.pdf

3 Anforderungs- und Marktanalyse

3.1 Schallwahrnehmung des Menschen

Um die Anforderungen an ein AVAS besser eingrenzen zu können, wird im folgenden Kapitel die Schallwahrnehmung genauer erläutert. Diese teilt sich in die Bereiche Hörphysiologie und Hörpsychologie auf. Über die Gegebenheiten der menschlichen Wahrnehmung und psychoakustische Phänomene lassen sich Rückschlüsse auf die notwendige Beschaffenheit des Klangs erschließen.

3.1.1 Hörphysiologie

Den wichtigsten Bereich in der Hörphysiologie stellt das Ohr dar. Es ist ein Sinnesorgan, welches sowohl für die Wahrnehmung von Schall als auch als Gleichgewichtsorgan dient. In Abb.3 ist ein Querschnitt des menschlichen Gehörgangs zu sehen.

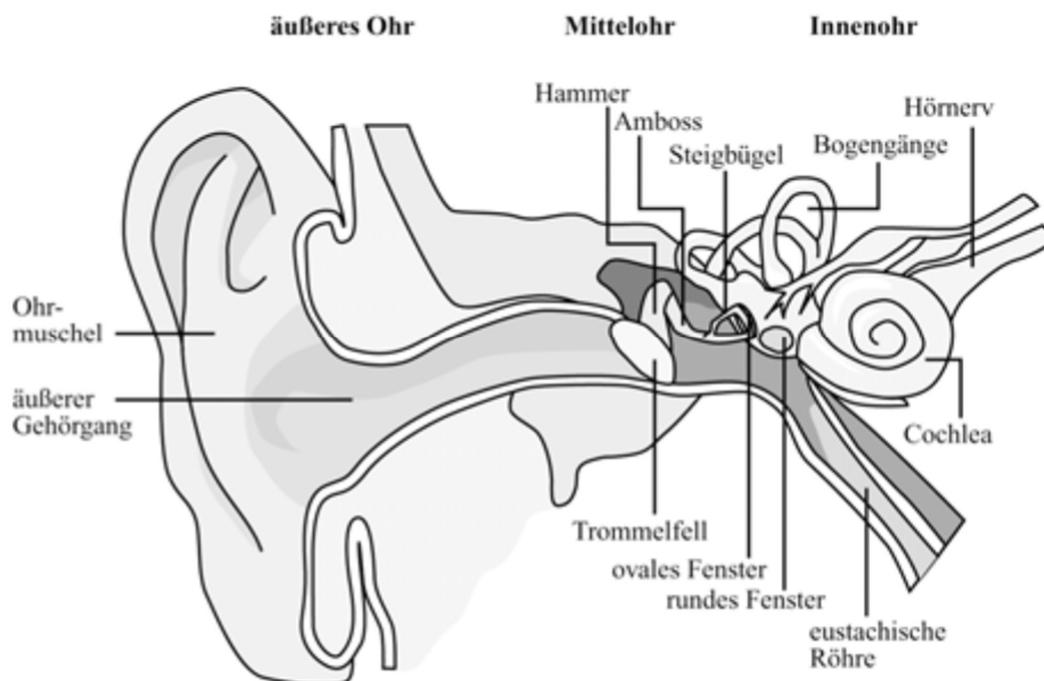


Abbildung 3 Querschnitt des menschlichen Ohrs⁸

⁸ Vgl. Dickreiter, S. 115

Im Wesentlichen lässt sich das Hörorgan in das Außen-, Mittel-, und Innenohr aufteilen. Über die Ohrmuschel am äußeren Ohr wird der Schall aus der Umgebung aufgefangen und über den etwa 3-4cm langen Gehörgang an das Trommelfell weitergeleitet. Hier werden, ähnlich wie bei einem Mikrofon, die Luftschwingungen in mechanische Schwingungen umgewandelt. Die Gehörknöchelchen Hammer, Amboss und Steigbügel dienen im Grunde als „Impedanzwandler“ im Ohr, sie übertragen außerdem die Auslenkung des Trommelfells über das ovale Fenster an das Innenohr.⁹

Im Innenohr befindet sich die mit einer Flüssigkeit gefüllte Cochlea. Hier wird der Schall schlussendlich in Nervenimpulse umgewandelt und über den Sehnerv an das Gehirn weitergeleitet. Der Innenraum der Cochlea besteht aus drei parallelen Kanälen, welche durch Membranen voneinander getrennt sind. Eine davon ist die Basilarmembran, welche in Wanderwellen schwingt. Durch diese Wellen entsteht auch die sogenannte Frequenz-Orts-Transformation. Sie gibt den Zusammenhang zwischen dem Ort des Erregungsmaximums auf der ausgebreiteten Basilarmembran, der Frequenz und der empfundenen Tonhöhe, an. Darauf wird später im Kapitel noch genauer eingegangen.¹⁰

Schallwellen aus unserer Umgebung sind eindeutig physikalisch definierbar. So hat zum Beispiel ein Sinuston eine eindeutig bestimmbare Frequenz und ebenso einen eindeutigen Schalldruckpegel. Dieses eindeutige Schallereignis wird durch den komplexen Gehörgang verarbeitet, über den Sehnerv ans Gehirn geleitet und erst dann entsteht eine Empfindung. Diese Empfindung ist im Gegensatz zu dem Schallereignis nicht mehr eindeutig. Sie kann nicht mehr durch die physikalischen Größen, die das Schallereignis definieren, wiedergegeben werden. An dieser Stelle setzt die Psychoakustik an, sie stellt den Zusammenhang zwischen Reiz und Empfindung her.¹¹

Es führt nicht jeder Reiz zu einer Empfindung. Die Hörfläche beschreibt den Bereich, in dem in Abhängigkeit von der Frequenz und dem Schalldruckpegel ein Hörereignis ausgelöst wird. Der Hörbereich hat mit der sogenannten Hörschwelle eine untere Grenze

⁹ Vgl. Raffaseder, S. 103-14

¹⁰ Vgl. Weinzierl, S. 42-44

¹¹ Vgl. Dickreiter, S. 116-118

und eine obere Grenze in Form der Schmerzgrenze. Diese Grenzen sind jedoch ebenfalls frequenzabhängig und unterscheiden sich je nach Frequenz um bis zu 40dB.

In Abbildung 4 ist die Hörfläche eines gesunden Ohrs zu sehen.

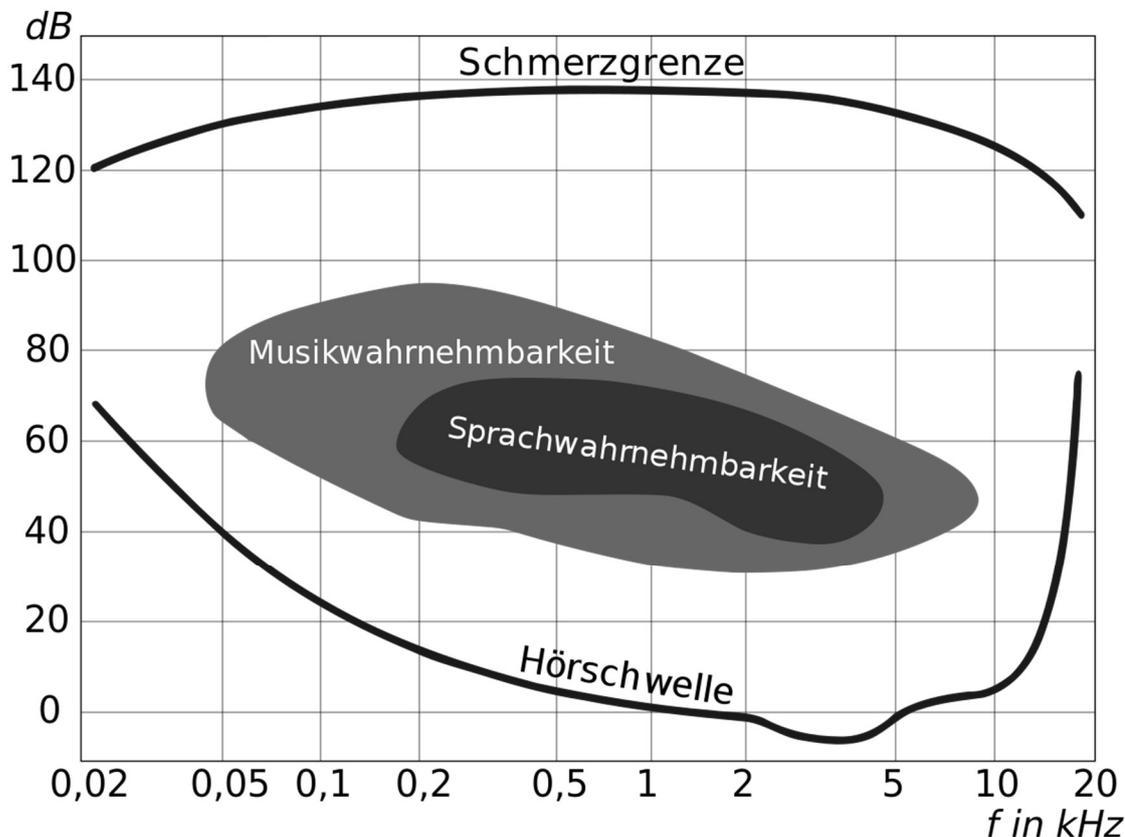


Abbildung 4 Menschliche Hörfläche¹²

Außerdem ist gut zu erkennen, dass das Ohr in den Mitten zwischen 500Hz und etwa 10kHz besonders empfindlich ist. So lässt sich beispielsweise bereits ableiten, dass ein Fahrgeräusch, welches sich im Frequenzbereich der Sprache befindet, leichter wahrgenommen wird als ein sehr tieffrequentes Signal. Gleichzeitig besteht hier auch das Risiko, dass das Geräusch schneller eine Inakzeptanz erreicht, da es besonders zwischen 2 kHz und 5 kHz sehr aufdringlich wirken kann.

3.1.2 Psychoakustik und Hörpsychologie

Die Hörschwelle knüpft an der Psychoakustik an, da es sich hier bereits um das tatsächlich wahrgenommene Signal handelt. Um eine bessere Vergleichbarkeit von Lautstärke

¹² <https://de.wikipedia.org/wiki/H%C3%B6rf%C3%A4che#/media/Datei:Hoerflaeche.svg>

in Bezug auf Audiosignale zu schaffen, gibt es die Kurven gleicher Lautstärke oder auch Isophone genannt. Sie geben mithilfe einer Kurvenschar an, wie sich der Schallpegel bei gleichlaut empfundenen Tönen über die Frequenz verhält.¹³

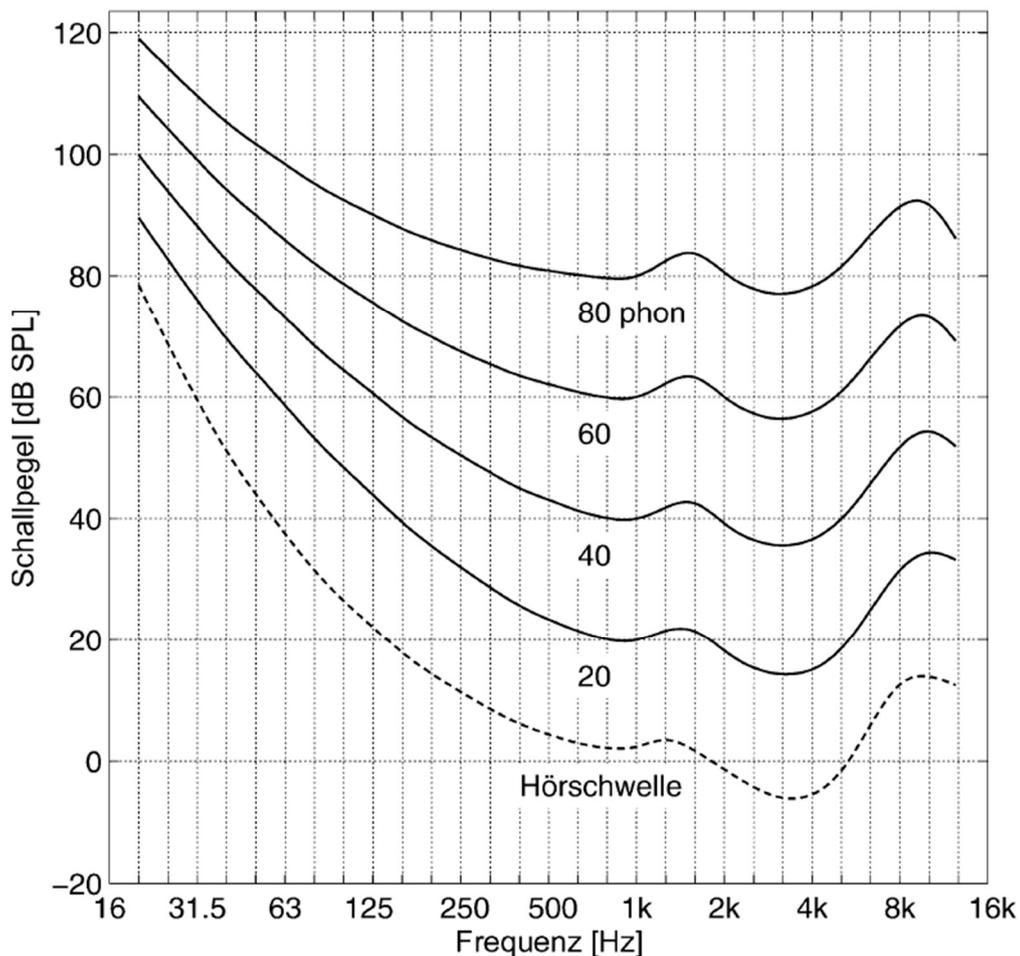


Abbildung 5 Hörschwelle und Isophone¹⁴

Die Isophone beziehen sich auf Sinustöne, der entsprechenden Frequenzen und werden in „phon“ angegeben. Daher sollten sie sich für komplexere Geräusche und Klänge lediglich als grober Anhaltspunkt verstehen.

Ein weiteres psychoakustisches Phänomen sind Verdeckungs- oder Maskierungseffekte, welche besonders auch im Straßenverkehr zum Tragen kommen können. Weiterhin muss bei den Maskierungseffekten zwischen Simultan- und Nach- bzw. Vormaskierung unterschieden werden. Bei der Simultanmaskierung verdecken Schallereignisse

¹³ <https://www.spektrum.de/lexikon/physik/kurven-gleicher-lautstaerke/8648>

¹⁴ Vgl. Weinzierl, S. 54

mit hohem Pegel, solche mit niedrigem. Je näher sich dabei die Frequenzen der Ereignisse sind, desto stärker wirkt dieser Effekt.

Bei der Vor- bzw. Nachmaskierung wird nach einem lauten akustischen Ereignis, ein leises Schallsignal unter Umständen maskiert. Das Ohr braucht je nach Schallpegel des lauten Ereignisses bis zu 100ms um sich zu erholen und für leisere Signale wieder empfindlich zu sein.¹⁵

Außerdem spielen auch die Lautstärke und die Dauer eines Schallereignisses eine Rolle. Das Ohr ist durchaus dazu in der Lage lange anhaltende Hintergrundgeräusche aufgrund von Ermüdung auszublenden. Gleichzeitig werden sehr laute, anhaltende Schallsignale nach einer Adaptionszeit leiser wahrgenommen.

Diese Maskierungseffekte sind beim Sounddesign für das AVAS zu berücksichtigen, da es sich gegen den bestehenden Lärmpegel im Straßenverkehr durchsetzen muss. So kann man aus einer Kombination der hörpsychologischen Erkenntnisse einige Rahmenbedingungen für den Klang und die technischen Voraussetzungen des AVAS festlegen und möglicherweise die Durchsetzungsfähigkeit des Klanges auf dessen Grundlage vereinfachen. Beispielsweise wäre ein eher tieffrequentes Signal nicht sinnvoll, da ein höherer Schallpegel notwendig wäre um die gleiche empfundene Lautheit zu erzielen.

¹⁵ Vgl. Raffaseder, S.112-113

3.2 Aktueller Stand der Technik

Durch die Gesetzgebung sind inzwischen bereits mit einem AVAS ausgestattete Fahrzeuge auf den europäischen Straßen unterwegs. Die verbauten Soundmodule haben einen recht kleinen Formfaktor und spielen voreingestellte Geräusche entsprechend der Fahrgeschwindigkeit ab.

Zur Analyse eines weit verbreiteten AVAS wurde ein System von der Firma Grewus aus Hamburg näher betrachtet. In Abbildung 6 ist ein Foto eines solchen AVAS zu sehen.



Abbildung 6 AVA-System von Grewus

Das Modul ist in etwa würfelförmig und knapp 8x8x8cm klein. Es benötigt eine Betriebsspannung zwischen 9 und 16 Volt. Daher wird es direkt an das 12V Bordnetz des Fahrzeugs angeschlossen. Außerdem gibt es noch eine „Controller Area Network-Schnittstelle“ (CAN), so können die Fahrzeugdaten aus dem System in das AVAS übertragen werden. Innerhalb der wasserdichten Hartplastikschale des AVAS befindet sich ein kleiner Lautsprecher und eine Platine. Diese darf aus betrieblichen Gründen hier nicht gezeigt werden.

Es ist ein simples Verstärkermodul verbaut, sowie die dazugehörige Elektrik für die Stromversorgung. Des Weiteren ist ein kleiner Flashspeicher verbaut, hier werden vermutlich die Geräusche, die das Fahrzeug wiedergeben soll, aufgespielt. Hinzu kommt ein kleiner Controllerchip, auf dem die Software läuft, welche aus den CAN-Signalen vom Fahrzeug, Informationen für die dynamische Veränderung des wiedergegebenen Fahrgeräuschs, erzeugt. Das Modul erzeugt so einen Schallpegel zwischen 50 und

90dBA.¹⁶

Um ein Gefühl zu bekommen, ob und wie das AVAS auf ein Fahrzeug der Fahrzeugklasse L übertragbar ist, wurde das Modul probeweise in einen Roller verbaut. Der Hersteller hat aufgrund der kleineren Größe des Fahrzeugs bereits angekündigt, dass das Modul etwas leiser eingestellt sein wird. Bei einer Testfahrt zeigte sich schnell, dass die Übertragbarkeit auf ein kleineres Fahrzeug kein Problem darstellt. Allerdings war in diesem Fall das Modul etwas zu leise eingestellt, sodass es bei geringer Geschwindigkeit nur in unmittelbarer Nähe zum Fahrzeug wahrgenommen werden konnte.

Das wiedergegebene Geräusch erinnert an einen relativ simpel gefilterten Sinuston, welcher außerdem eine Art sehr schnellen Tremolo-Effekt hatte. So klang das Geräusch etwas nach einem digitalen Telefonklingeln, welches die Lautstärke und Tonhöhe mit zunehmender Geschwindigkeit erhöht. Laut des Herstellers kann hier aber nach Rücksprache jedes beliebige Geräusch eingepflegt werden und auch auf die dynamische Gestaltung des Geräuschs kann Einfluss genommen werden. Bei diesem System ist dies immer mit einem Programmieraufwand verbunden, welcher nur durch den Hersteller zu bewerkstelligen ist. Die Anpassung des Geräuschs ist also nur über Grewus möglich, daher bestand keine Möglichkeit verschiedene Geräusche auszuprobieren. Zudem hielt sich der Hersteller in diesem Punkt aus betrieblich-, strategischen Gründen eher zurück.

Nach Angaben von Grewus befinden sich diese Systeme bereits in einigen Fahrzeugen. Da die Klänge individualisierbar sind, lässt sich als Passant keine Aussage über das verbaute System treffen. Nachfolgend werden die Spektren von AVAS-Fahrgeräuschen von verschiedenen Herstellern verglichen.

3.3 Spektrale und klangliche Analyse bestehender AVAS

In diesem Abschnitt werden drei Spektren von bereits ausgelieferten Elektrofahrzeugen miteinander verglichen. Das Bildmaterial wurde mithilfe von iZotope RX7 auf Grundlage der sich in Anhang A.1 befindlichen Audiodaten erzeugt. Dabei wird aus Gründen der Redundanz immer nur ein Kanal betrachtet, da die Signale alle in Mono

¹⁶ <https://www.grewus.de/produkte/avas-soundmodul/>

wiedergegeben werden. Die Fahrsituation unterscheidet sich zum Teil, nicht alle Fahrzeuge bremsen am Schluss des Audiofiles wieder ab, sondern fahren entweder davon oder hören schlagartig auf. Alle zum Vergleich herangezogenen Beispiele stammen aus einem Zusammenschnitt der „Auto Motor Und Sport“.¹⁷

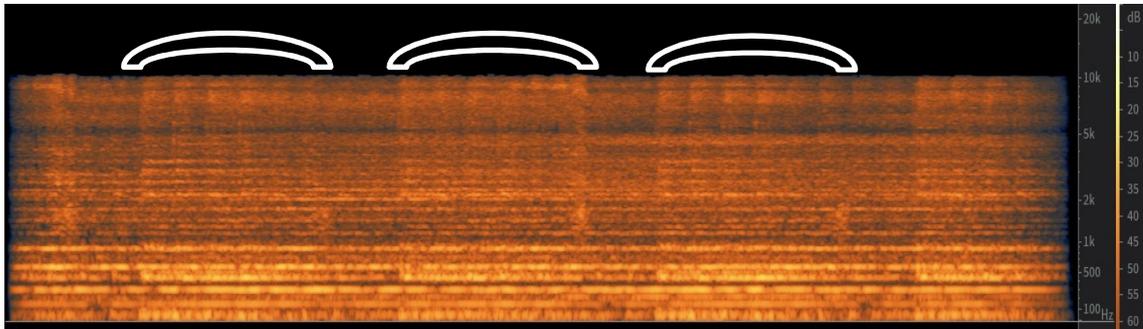


Abbildung 7 Spektrum des Fahrgeräuschs eines Citroen C-Zero

Bei dem Citroen C-Zero handelt es sich vermutlich um den experimentellsten Sound von allen in diesem Vergleich herangezogenen Modellen. Wenn man das Spektrum betrachtet, ist aber auch hier, ähnlich wie bei der EU-Empfehlung aus Kapitel 2.1, zu erkennen, dass es sich um ein harmonisches Signal handelt. Ebenfalls ist eine gewisse rhythmische Komponente zu sehen – besonders zwischen 1kHz und 10kHz gut zu erkennen. Sie wiederholt sich in den markierten Blöcken und hat innerhalb der Blöcke etwa 4 Ausschläge. Wenn man sich das Signal anhört, befinden sich an diesen Stellen vokal klingende Stöße. Sie klingen choral und feminin, allerdings ist es bei diesem Beispiel schwer zu beurteilen wie und ob sich das Geräusch der Fahrgeschwindigkeit anpasst oder nicht. Das liegt mitunter daran, dass sich der Sound aus diesem Beispiel über die Zeit kaum bis gar nicht verändert. Das Hörbeispiel gibt aber trotzdem einen Anhaltspunkt für die Herangehensweise des Sounddesigners und die Art des Klanges, die der Hersteller anstrebt.

Als Basis dient eine Mischung aus einem Synthesizer und einem synthetisch klingenden Chor, der zwischen 50Hz und 1kHz harmonisch das Spektrum ausfüllt. Wie eingangs beschrieben gibt es darauf aufbauend die rhythmischen Ergänzungsklänge, die im Spektrum etwas höher angesetzt sind und sich deshalb etwas vom Grundklang abheben.

¹⁷ <https://www.auto-motor-und-sport.de/tech-zukunft/sound-fuer-elektroautos/>

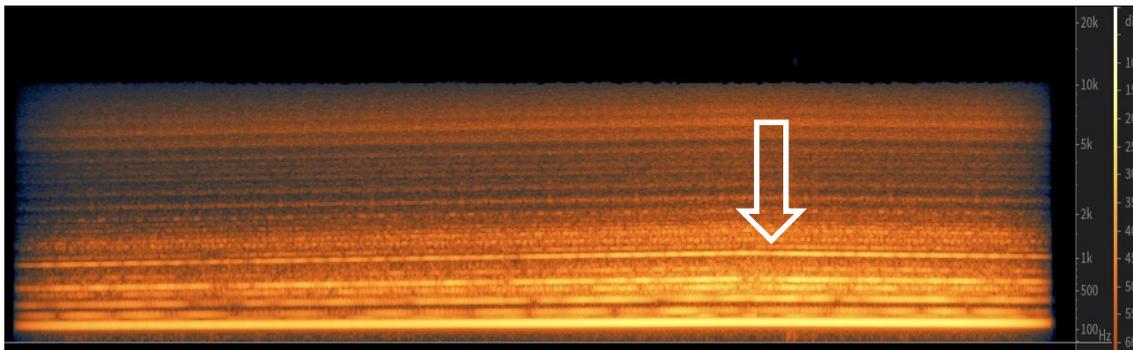


Abbildung 8 Spektrum des Fahrgeräuschs eines Renault Twizy

Beim zweiten Beispiel handelt es sich um den Renault Twizy. Er ist für diese Arbeit besonders interessant, da er unter die Kategorie der Leichtelektromobilmfahrzeuge (LEM), genauer in den Typ L6e bzw. L7e fällt und somit vom Typ her näher an einem Elektroroller ist als andere Modelle aus diesem Vergleich.¹⁸

In Abbildung 8 ist gut zu erkennen, dass es sich um ein harmonisches Signal handelt. Das Spektrum zeigt, dass sich die Linie der Grundfrequenz, die in diesem Fall breitbandig zwischen 100 und 200 Hz liegt, lediglich zu höheren Frequenzen hin parallel dupliziert. Durch die reine Betrachtung des Spektrums deutet dies bereits auf einen sehr tonalen Klang hin, welcher seine Tonhöhe bis zur in Abbildung 8 markierten Stelle linear erhöht. Im Anschluss an die Markierung sinkt der Pitch wieder ab.

Das deutet darauf hin, dass sich die Tonhöhe dynamisch mit der Geschwindigkeit des Fahrzeugs verändert. Je schneller das Fahrzeug fährt, desto höher ist auch die Tonlage des wiedergegebenen Geräuschs. Des Weiteren kann man am Farb- bzw. Helligkeitsverlauf des Spektrums auch erkennen, dass das Geräusch bis zur Markierung lauter wird. Dies ist ein weiterer Indikator dafür, dass das Geräusch dynamisch auf das Fahrverhalten reagiert. Je schneller das Fahrzeug fährt, desto lauter also auch das AVAS. Klanglich erinnert das Signal an einen Synthesizer, welcher sich in Tonhöhe und Lautstärke der Fahrgeschwindigkeit anpasst. Es ist auffällig, dass das Geräusch sehr weich klingt und sich vor allem im tiefen und mittleren Frequenzbereich abspielt. Die breitbandige, tiefe Grundfrequenz des Signals erinnert ein wenig an einen klassischen Verbrennungsmotor, welcher besonders beim Anfahren sehr tieffrequent ist.

¹⁸ <https://www.homeandsmart.de/elektroauto-renault-twizy-preis-reichweite-vergleich>

Zusammengefasst ist es aber ein eher unauffälliges, neuartig klingendes Geräusch, das im entferntesten Sinne an einen Verbrennungsmotor angelehnt ist.

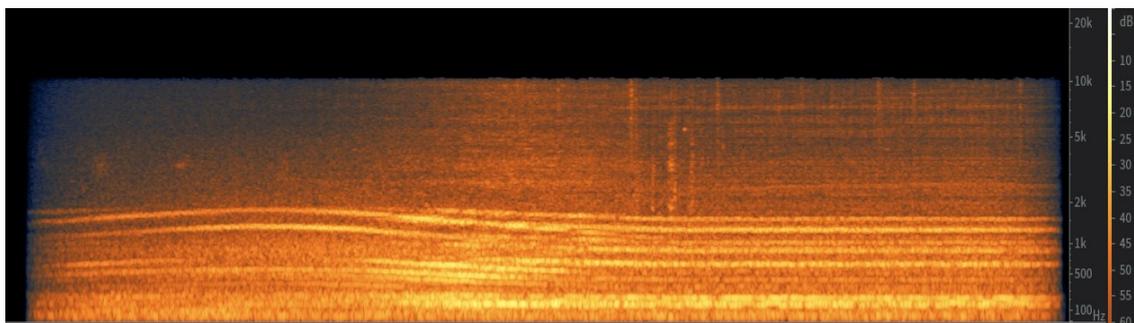


Abbildung 9 Spektrum des Fahrgeräuschs des BMW i8

Beim letzten Modell des Vergleichs handelt es sich um den Elektroswagen i8 von BMW. Auch hier ist aufgrund der parallelen Verteilung der Frequenzen im Spektrum in Abbildung 9 schnell zu sehen, dass es sich um ein harmonisch klingendes Signal handelt. Zu Beginn des Signals kann man erkennen, dass sich das Geräusch in der Tonhöhe nach oben bewegt. Dies geschieht nicht wie beim Renault linear, sondern eher kurvenförmig und etwas weicher. Danach ist im zeitlichen Verlauf zu erkennen, dass die Tonhöhe wieder absinkt, die Lautstärke des Signals aber dafür zunimmt. Hier hängt es damit zusammen, dass sich das Fahrzeug bei diesem Signal dem Mikrofon annähert und dadurch der Pegel ansteigt. Daher gilt es die Lautstärkeveränderung in diesem Fall für die Analyse des Geräuschs zu vernachlässigen.

Es ist zu hören, dass das Fahrzeug anfährt, sich anschließend nähert und anhält. Das Geräusch ist ähnlich aufgebaut wie das des Renault. Je schneller das Fahrzeug ist desto höher wird das wiedergegebene Signal. Auch hier basiert der Kern des Geräuschs auf einer Art synthetisiertem, harmonischen Geräusch.

An diesen drei Beispielen ist gut zu erkennen, dass sich die Hersteller zunächst an den Vorgaben der EU (s. Kap. 2) orientieren. Besonders der Klang der Geräusche vom Renault Twizy und dem BMW i8 ähneln sich sehr und das, obwohl es sich um zwei grundlegend verschiedene Fahrzeugtypen handelt. Der Citroen hat sich klanglich in eine andere Richtung entwickelt wie die Konkurrenz und befindet sich trotzdem innerhalb der gesetzlichen Vorgaben. Dennoch ist bereits zu hören, dass sich auch im Rahmen der EU-Vorgaben, durchaus verschiedene Fahrgeräusche produzieren lassen.

3.4 Umfrage zu AVAS

Als Grundlage für dieses Kapitel dienen die Ergebnisse der Umfrage aus Anhang A.2. Sie soll die Erwartungshaltung von Verkehrsteilnehmern an ein AVAS, grob einordnen.

Es wurden knapp 100 Leute befragt, gut dreiviertel davon sind männlich. Beruflich ist etwa ein Drittel im Ingenieursbereich tätig und ein Viertel sind Studierende. Das ist vor allem dem Umfeld, in dem diese wissenschaftliche Arbeit angefertigt wurde, zuzuschreiben.

Die Altersstruktur ist zwischen 20 und ca. 60 Jahren gut verteilt. Es fällt lediglich auf, dass zwischen Mitte 20 und Anfang 30 etwas mehr Leute an der Umfrage teilgenommen haben.

Aus diesen Gründen ist zu berücksichtigen, dass sich ein relativ hoher Anteil der Befragten aus einem eher jungen und fachnahen Publikum zusammensetzt.

Knapp drei Viertel der Befragten fährt ein Auto, innerhalb dieser Menge fährt ungefähr jeder Dritte außerdem auch ein Motorrad oder einen Roller. Insgesamt fahren knapp 40% der Befragten regelmäßig selbst mit Motorrädern oder Rollern. 20% der Befragten fahren weder Roller oder Motorrad, noch Auto.

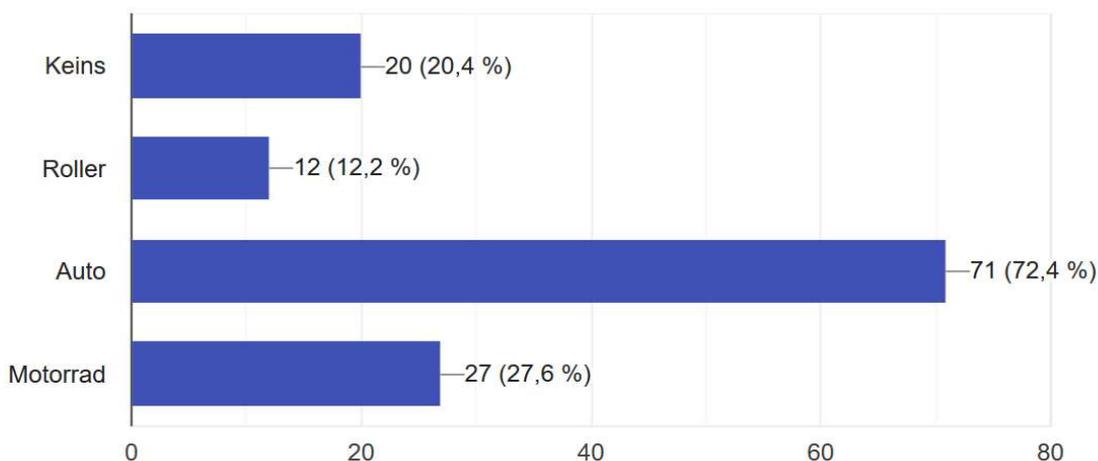


Tabelle 2 Auto-, Roller-, Motorradfahrer/innen

Über 80% der Befragten wünschen sich ein Fahrgeräusch als nicht-fahrender Verkehrsteilnehmer. Fast alle Befragten, die sich kein Geräusch wünschen, haben trotzdem später in der Umfrage angekreuzt, dass das Geräusch unauffällig sein soll. Lediglich drei

davon haben explizit kommentiert, dass sie sich gar kein Geräusch wünschen, um die Lärmbelastigung in den Städten zu reduzieren.

Wenn es hingegen um die Geräuschwiedergabe des Fahrzeugs als Fahrer geht, tendiert etwas mehr als die Hälfte dazu, dass es kein Geräusch geben soll. Wenn das Geräusch als Passant nicht erwünscht ist, war es auch in fast allen Fällen als Fahrer unerwünscht. Umgekehrt ist es zwar auch recht häufig der Fall, die Tendenz geht aber dahingehend, dass das Geräusch vor allem für die Passanten wiedergegeben werden soll.

Daraus lässt sich teilweise schon ableiten, dass gut die Hälfte der Befragten keinen großen Wert auf die Emotionalisierung durch die Fahrgeräusche legt, da es ihnen selbst nicht wichtig ist das Geräusch wahrzunehmen. Gleichzeitig lässt sich daraus auch schließen, dass dieser Gruppe von Leuten der Sicherheitsaspekt des Fahrgeräusches besonders wichtig ist. Es gilt allerdings zu beachten, dass etwa ein Viertel dieser Teilmenge von Befragten selbst gar kein Fahrzeug fährt.

Die Antworten auf Frage 3 sind sehr deutlich ausgefallen. Knapp 85% der Befragten stellen sich elektrisch betriebene Roller klanglich anders vor als klassische Roller mit Verbrennungsmotor. Dabei ist es relativ unabhängig davon ob es Motorrad-, Roller oder Autofahrer sind. Daraus lässt sich schließen, dass nicht nur in der Mobilität an sich ein Umdenken erwünscht ist, sondern auch im Erscheinungs- bzw. Klangbild neuartiger Antriebsarten.

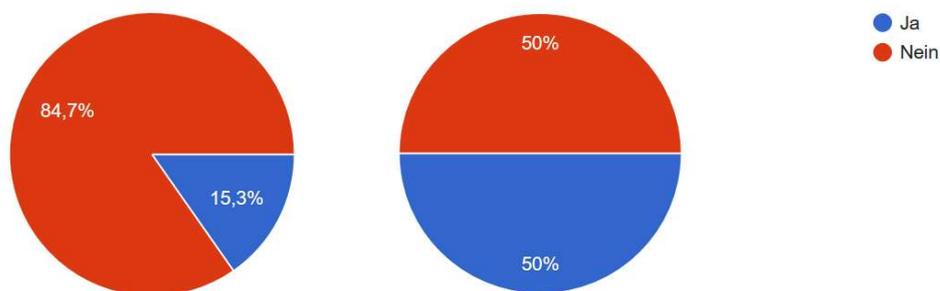


Tabelle 3

(links) Soll das Elektrofahrzeug nach einem klassischen Verbrennungsmotor klingen?

(rechts) Soll das Elektrofahrzeug nach künstlich generierten Geräuschen klingen?

Die Intention von Frage 4 war zunächst, das Gegenteil von Frage 3 abzufragen. Jedoch spricht das Ergebnis dafür, dass dies nicht so aufgefasst wurde. Exakt die Hälfte der Befragten wünscht sich, dass der Klang nach künstlich generierten Geräuschen klingt. Die Ableitung daraus ist, dass sich die andere Hälfte ein „natürlich“ klingendes, neuartiges Geräusch vorstellt, ohne dass es offensichtlich nach einem synthetisierten Geräusch klingt. Also einen Klang, der klassischerweise nicht von Fahrzeugen mit Verbrennungsmotor bekannt ist, aber trotzdem damit assoziiert wird.

Diese Erwartungshaltung deckt sich gut mit den in Kapitel 2 zusammengefassten, rechtlichen Anforderungen an das AVAS. Es ist ebenfalls vorstellbar, dass durch aktuellere Modelle, die bereits mit einem AVAS ausgestattet sind, die Vorstellungen der Befragten etwas voreingenommen sind. Gerade die Modelle, die in Kapitel 3.3 genannt wurden, sind teilweise schon auf den Straßen unterwegs und orientieren sich auch an den Vorgaben der UN. So bildet sich bereits ein allgemeines Verständnis bzw. eine Erwartungshaltung an die Klänge von elektrisch betriebenen Fahrzeugen.

Die Charakterisierung des Klanges durch die Adjektive hat auch eine recht eindeutige Richtung. Die vier meist gewählten Adjektive waren gleitend, elektrisch, harmonisch und unauffällig. Gleichzeitig sind diese vier Adjektive mit unter auch die häufigste Kombination. Erweitert man sie noch um „weich“, dann setzen sich über die Hälfte der ausgewählten Kombinationen aus diesen fünf Adjektiven zusammen. Diese Zahlen bestärken die Annahme, dass ein neuartiger Klang gewünscht ist. Dieser darf jedoch unter keinen Umständen als aufdringlich oder störend empfunden werden, sondern soll sich nahtlos in die Klangkulisse des Straßenverkehrs einfügen.

In Tabelle 4 sind die Begriffe nach dem Stichwortverzeichnis, das der Umfrage zugrunde liegt, in „unauffällige“ und „auffällige“ Begriffe zusammengefasst worden. So lässt sich erkennen, dass sich nur knapp ein Drittel der Befragten ein auffälligeres Fahrgeräusch vorstellen kann.

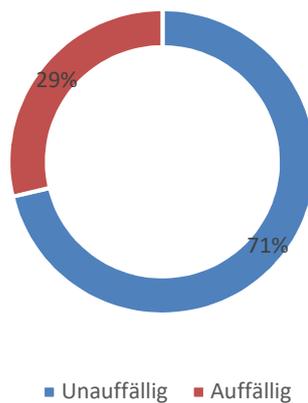


Tabelle 4

Vergleich der Adjektive die einen auffälligen bzw. unauffälligen Klang beschreiben

Neben den genannten Begriffen wurden auch jeweils von etwa einem Viertel die Adjektive „kraftvoll“ und „futuristisch“ gewählt. Ersteres ist selten in Kombination mit den häufigen genannten Begriffen ausgewählt worden, jedoch sind fast alle Antworten, die „herausstechend“ beinhalten, in Kombination mit „kraftvoll“ gewählt worden. Das spricht dafür, dass Leute mit einer eher sportlichen Veranlagung zu fahren, dies auch gerne andere Verkehrsteilnehmer hören lassen wollen. An dieser Stelle spielt die Emotionalität beim Fahren eine besondere Rolle, wie es z.B. in Sportwagen unter anderem über den kräftigen Motorenklang erreicht wird.

Die Antworten, die „futuristisch“ beinhaltet haben, wurden in verschiedensten Kombinationen ausgewählt. Daraus lässt sich schließen, dass zwar ein neuartiges Geräusch erwünscht ist, allerdings die Vorstellungen darüber weit auseinander gehen. Auffällig ist außerdem, dass viele Befragte, die „futuristisch“ angegeben haben, auch einen Kommentar mit einem eigenen Vorschlag verfasst haben. Beispielsweise „Transformers“ oder „Star Wars, Star Trek“. Die Vorstellungen von „futuristischen“ Klängen zur Fortbewegung sind durch Science-Fiction und Pop-Kultur maßgeblich geprägt worden. Tatsächlich sind diese Geräusche teilweise nicht weit von der Realität der Klänge von aktuellen Elektrofahrzeugen (EVs) entfernt und bieten einen validen Anhaltspunkt für die Erzeugung eines interessanten Fahrgeräusches.

In der letzten Frage nach den Zusatzfunktionen, sind sie zumeist jeweils auch recht eindeutig in ihrer Wichtigkeit bewertet worden. Die „freundliche Hupe“ wurde von knapp einem Drittel als sehr wichtig empfunden, ein weiteres Drittel findet sie wichtig. Auch hier schwingt mit, dass es den Befragten vor allem um Sicherheit im Straßenverkehr geht. Die „freundliche Hupe“ bietet die Möglichkeit aktiv auf sich aufmerksam zu machen, ohne dabei unnötigen Lärm zu verursachen und womöglich unbeteiligte Verkehrsteilnehmer zu erschrecken.

Die Möglichkeit die Fahrgeräusche zu personalisieren wurde tendenziell als unnötig bewertet oder hat zumindest nur eine neutrale Bewertung bekommen. Obwohl diese Zusatzfunktion ein Feature wäre um sein Fahrzeug zu personalisieren wird lieber darauf verzichtet. So wird verhindert, dass im Straßenverkehr der Zukunft ein Durcheinander von verschiedenen Fahrgeräuschen entsteht. Dieses Problem existiert mit Verbrennungsmotoren nicht. Der natürliche Schall, den Verbrennungsmotoren erzeugen, ähnelt sich bei allen Fahrzeugen in seinen Grundzügen und unterscheidet sich besonders bei Klein- und Mittelklasse Wagen nur marginal. Das AVAS bietet bereits ohne Personalisierung eine immense Bandbreite an verschiedenen möglichen Klängen. Wenn lediglich die Hersteller den Klang bestimmen, wird dieses Problem unter Umständen gar nicht erst entstehen, da die Klänge wesentlich kontrollierter erstellt werden. Trotzdem kann dieser Punkt in Zukunft noch zu einer unangenehm klingenden Verkehrssituation führen und sollte mit der steigenden Anzahl an EVs, im Hinterkopf behalten werden.

Der Wunsch nach Statussignalen ist hingegen ziemlich groß. Über zwei Drittel der Befragten halten diese Funktion für sehr wichtig bis wichtig. Hierbei gilt jedoch zu bedenken wie sinnvoll es ist das Signal über das AVAS wiederzugeben, da es in erster Linie darum geht den Fahrer und nicht die umliegenden Verkehrsteilnehmer auf Statusänderungen hinzuweisen. Eine einfachere Lösung wäre es, wenn es möglich ist einen Lautsprecher bzw. ein Piezoelement anzusteuern, welches das Warnsignal in Richtung des Fahrers wiedergibt.

Bei der letzten Zusatzfunktion ist die Intention gewesen, indirekt abzufragen, was die Befragten von „Gameification“ im Straßenverkehr in Bezug auf effiziente Fahrweise halten. Mit „Gameification“ ist in diesem Fall gemeint, wie sehr sich der Fahrer bemüht effizient zu fahren, wenn er dafür mit einem angenehmeren Fahrgeräusch belohnt

wird. Im Umkehrschluss, wie verhalten sich Fahrer, wenn sie ineffizient fahren und dafür einem etwas unangenehmeren Geräusch ausgesetzt werden.

Hier sind die Stimmen überwiegend neutral, in etwa gleich viele halten sie für wichtig bzw. unwichtig. Jedoch sind die Stimmen für „unnötig“ deutlich höher als für „sehr wichtig“, so ergibt sich auch eher die Tendenz, dass die Funktion nicht unbedingt erwünscht ist. Es ist anzunehmen, dass es einigen Fahrern zu unangenehm wäre, wenn das unmittelbare Verkehrsumfeld mitbekommt, dass dieser gerade ineffizient fährt und damit ein unangenehmes Geräusch abgibt. Womöglich kann dies zu frustrierten Verkehrsteilnehmern führen.

Zusammengefasst wird das AVAS als sehr sinnvolle Ergänzung der E-Mobilität wahrgenommen. Klanglich gehen die Vorstellungen teilweise auseinander, jedoch ist die klare Tendenz, dass es ein Geräusch sein soll, das sich natürlich in den Straßenverkehr einbettet und auf eine eher subtile Art andere Verkehrsteilnehmer warnt. Der Vorteil von Elektrofahrzeugen, dass sie nahezu lautlos betrieben werden und so den Straßenverkehr in Zukunft leiser gestalten zu können, soll trotz des AVAS erhalten bleiben.

Es ist eher die Ausnahme, dass die Befragten durch den Klang emotionalisiert werden möchten und ein „Sportwagen“ ähnliches Erlebnis haben möchten. Ebenso selten ist die Meinung vertreten, dass das AVAS versuchen soll den Klang eines Verbrennungsmotors zu imitieren. Die Befragten sind im Schnitt aufgeschlossen für eine neue Geräuschkulisse im Straßenverkehr, was insbesondere unter Rücksichtnahme auf das befragte Publikum betrachtet werden muss.

4 Annäherung an die Problemstellung

4.1 Motivation

Nachdem der Markt grob analysiert wurde und das Anforderungsprofil eingegrenzt werden konnte, gilt es ein eigenes AVA-System unter Berücksichtigung der Ergebnisse zu erstellen und zu definieren.

Das AVAS soll die Sicherheit von schwächeren Verkehrsteilnehmern und besonders von Passanten auf auditiver Ebene gewährleisten. Dabei soll das Fahrgeräusch idealerweise auf allgemeine Akzeptanz im Straßenverkehr stoßen und auch nach längerer Expositionsdauer nicht als anstrengend und unangenehm empfunden werden.

Gleichzeitig soll der Fahrer subtil hörbares Feedback für den niedrigen Geschwindigkeitsbereich bekommen. Auf diese Weise ist beispielsweise schon beim Start des Fahrzeugs sofort ersichtlich, dass das Fahrzeug in Betrieb ist, was bei Elektrofahrzeugen nicht immer automatisch der Fall ist.

Neben den Sicherheitsaspekten soll auch eine gewisse Emotionalität beim Fahrer und unter Umständen auch beim Passanten entstehen. Da Elektromobilität eine kohlenstoffdioxidfreie und innovative Fortbewegung ermöglicht, sollen die Fahrzeuge auch entsprechend erstrebenswert klingen. Es soll ein Wertigkeitsgefühl vermittelt werden, sodass die klimafreundliche Mobilität im Straßenverkehr als etwas Positives wahrgenommen wird.

Gleichzeitig sollen Hersteller davon profitieren können ihre AVA-Systeme zu individualisieren, so können sie ihre markenspezifischen Wünsche in den Klang einarbeiten und die Markenkommunikation auf die klangliche Ebene erweitern.

4.2 Use Cases

Da das AVAS einen in das Fahrzeug verbauten Lautsprecher mit sich bringt, öffnen sich etliche Möglichkeiten, das System auch für andere Zwecke zu verwenden. So entstand beispielsweise auch die Überlegung eine „freundliche Hupe“ zu integrieren. Sie hat zum Zweck Passanten auf eine subtile Art und Weise warnen zu können, ohne die herkömmliche, sehr laute und harsch klingende Hupe zu verwenden. Die Möglichkeit da-

für würde außerdem auch die in Kapitel 4.1 erwähnte allgemeine Akzeptanz im Straßenverkehr erhöhen, da die Hupe idealerweise nicht mehr als besonders störend empfunden wird. Gerade auch im Hinblick auf „letzte-Meile-Logistik“ ist dies ein interessanter Use Case, da diese häufig auf Gehwegen oder in Fußgängerzonen unterwegs sind um Pakete zuzustellen. Mit der „freundlichen Hupe“ könnten Paketzusteller in Zukunft auf sich aufmerksam machen, ohne dabei zu stören oder einen unnötig hohen Lärmpegel zu erzeugen. Gleichzeitig könnte hier auch wieder die Markenkommunikation ansetzen und die „freundliche Hupe“ sogar als eine Art „Soundlogo“ nutzen.

Eine weitere Funktion könnte sein, das Fahrgeräusch personalisierbar zu machen. So könnten die Nutzer ihr persönliches Fahrerlebnis verbessern. Für die Hersteller könnte es interessant sein, da sie die Möglichkeit hätten Soundsets zu verkaufen um dadurch noch nach dem Verkauf des Fahrzeugs an sich Geld zu verdienen.

Allerdings gilt hier, wie bereits in Kapitel 3.4 dargestellt, zu beachten, dass dann jedes Fahrgeräusch homologiert werden müsste, was mit zusätzlichen Kosten auf der Herstellerseite verbunden ist. Außerdem besteht bei der Personalisierbarkeit das Potenzial, dass die Geräuschvielfalt im Straßenverkehr so groß ist, dass sie in Summe zu einem unangenehmen Klangbild auf den Straßen führen könnten.

Weiterhin besteht die Möglichkeit Statussignale über das AVAS wiederzugeben. Hierbei muss man sich allerdings die Frage stellen, wie relevant bestimmte Statussignale für andere Verkehrsteilnehmer sind. Lediglich Warnsignale bei Ausfall einer Kernfunktion des Fahrzeugs könnten hierbei eine Rolle spielen. Der Verkehr wäre sofort informiert darüber, wenn beispielsweise die Batterie ausfällt oder eine Bremse defekt ist.

Zuletzt gibt es die Idee, das Fahrgeräusch je nach Fahrweise anzupassen. Dabei kann sich die Änderung auf die restliche Reichweite beziehen oder beispielsweise auch auf einen Effizienzfaktor. Dieser würde dann über einen gewissen Zeitraum den aktuellen Verbrauch errechnen und dementsprechend das Geräusch anpassen. Diese Art von Feedback ist in erster Linie allerdings für den Fahrer gedacht und weniger für andere Verkehrsteilnehmer.

Basierend auf den Ergebnissen aus Kapitel 3.4, wurde unter den Use Cases lediglich die „freundliche Hupe“ mit in Betracht gezogen und als sinnvolle Ergänzung zum AVAS gesehen.

5 Technische Umsetzung des AVAS

5.1 Controller Area Network

Zunächst wird in diesem Kapitel das Controller Area Network (CAN) erklärt, da es für die Umsetzung des AVAS unabdingbar ist.

Das Controller Area Network ist eine serielle Schnittstelle zur Übertragung von Informationen zwischen Komponenten innerhalb eines technischen Systems. Über dieses Bussystem lassen sich verschiedene Komponenten leicht verbinden, bei gleichzeitiger Einsparung von extra Verkabelungen und Implementierung exklusiver Schnittstellen. Es kommt vor allem in Kraftfahrzeugen zum Einsatz, sodass die elektronische Steuereinheit dazu in der Lage ist, mit einem Kabelpaar verschiedenste Komponenten im Fahrzeug anzusteuern.¹⁹

Die Komponenten werden über Knoten miteinander verbunden und funktionieren bidirektional, wie in Abbildung 10 zu sehen ist.

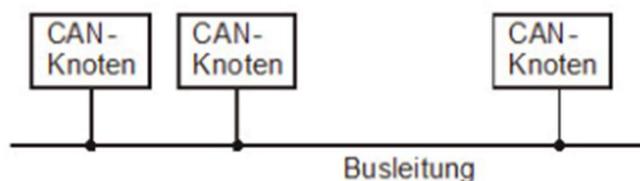


Abbildung 10 Controller Area Network²⁰

Das Signal wird differenzial übertragen, das führt zu einer elektrischen Störsicherheit. Da auf beiden Leitungen mögliche Einstreuungen auftreten können, heben sich die Störungen durch die gegensinnigen Potenziale der beiden Leitungen auf. Das Signal ist durch diese invertierte Übertragung vollständig redundant.

Deshalb ist für die Übertragung eines CAN-Signals mindestens ein 3-poliges Kabel notwendig.

¹⁹ <https://www.allaboutcircuits.com/technical-articles/introduction-to-can-controller-area-network/>

²⁰ <https://www.me-systeme.de/de/technik-zuerst/elektronik/can-bus-grundlagen>

In der Regel wird allerdings ein sogenannter 9-poliger Sub-D Steckverbinder genutzt.

CAN Pinbelegung auf DSUB9		
Pin	Signal	Beschreibung
1	Reserved	Upgrade Path
2	CAN_L	Dominant Low
3	CAN_GND	GND
4	Reserved	Upgrade Path
5	CAN_SHLD	Shield, Optional
6	GND	Ground, Optional
7	CAN_H	Dominant High
8	Reserved	Upgrade Path
9	CAN_V+	Power, Optional

Abbildung 11 CAN Pinbelegung auf D-SUB 9²¹

In Abbildung 11 ist die Belegung der Pins zu sehen, dabei sind abgesehen von den Pins 2, 3 und 7 alle Pins optional und bieten zusätzliche Übertragungsstabilität oder andere Vorteile wie eine Versorgungsspannung.

Die Datenrate der Übertragung des Signals ist vor allem abhängig von der maximalen Kabellänge, die das System verwendet. Je nach Länge und verarbeitetem Material sind bei 25 m Kabellänge 1 Mbit/s möglich und bei knapp 7 km nur noch 10 KBit/s.²²

Der CAN-Bus arbeitet objektorientiert, das bedeutet, dass die Nachrichten keinen konkreten Empfänger haben, stattdessen hat jede Nachricht einen Object-Identifier. Dieser ermöglicht es, dass jeder Knoten im CAN-System selbstständig entscheiden kann, ob die jeweilige Nachricht eine Relevanz für den Knoten hat oder nicht. Des Weiteren können mithilfe des Object-Identifiers Nachrichten in ihrer Priorität variieren. Je niedriger die Object-ID, desto höher wird die Nachricht priorisiert.²³

So ist beispielsweise am Fahrzeug die Object-ID der Geschwindigkeit oder der Beschleunigung recht niedrig, da dies wichtige Kernparameter eines Fahrzeugs sind.

²¹ https://www.opendcc.de/info/can/can_connector.html

²² <https://www.me-systeme.de/de/technik-zuerst/elektronik/can-bus-grundlagen>

²³ <http://www.elektro-archiv.de/archiv/c/can/>

Die CAN-Nachrichten selbst sind in Frames aufgeteilt, welche sich wiederum in acht Kennfelder unterteilen lassen.



Abbildung 12 Unterteilung eines CAN-Frames²⁴

Bei dem in Abbildung 12 gezeigten Frame handelt es sich um ein sogenanntes Extended-Frame, dabei ist die CAN-ID 29 Bit groß. Bei einem Basis Frame hätte die CAN-ID nur 11 Bit. Eine wichtige Unterscheidung bei den Frames ist außerdem, ob es ein Daten-Frame ist oder eine sogenannte Remote Transmission Request (RTR). Die RTR wird angefordert und der Empfänger, der die Nachricht angefordert hat, liefert dann das korrespondierende Daten-Frame.

Das CAN-Netzwerk ist eigenständig dazu in der Lage Fehler zu erkennen und zu signalisieren. Der Cyclic Redundancy Check (CRC) berechnet aus den Datenbits zusätzliche Prüfbits, welche der Nachricht hinzugefügt werden. So kann der Empfänger diese überprüfen und bei Nicht-Übereinstimmen der Prüf- und Datenbits einen Fehler melden.

Des Weiteren existiert noch ein Frame-Check, er überprüft die Struktur des Rahmens und meldet bei Unstimmigkeiten einen Formatfehler.

Zuletzt gibt es den Acknowledgement-Fehler, welcher die korrekt empfangene Nachricht bestätigt. Sobald am Sender kein Acknowledgement erkannt wird, entsteht ein ACK-Fehler. Dieser deutet auf einen Übertragungsfehler hin, der nur vom Empfänger erkannt wird.²⁵

So bietet der CAN-Bus eine sehr sichere, universell einsetzbare Datenübertragung im Fahrzeug, das AVAS wird ebenfalls an das CAN-System im Fahrzeug angebunden.

²⁴ <https://altronics.fr/en/guide/can-bus-and-its-benefits/>

²⁵ <https://www.me-systeme.de/de/technik-zuerst/elektronik/can-bus-grundlagen>

5.2 Konzeption der Hard- und Softwarekomponenten

Zu Beginn der Umsetzung stand eine umfangreiche Recherche für Hard- und Softwarekomponenten, die sich zur Umsetzung des AVAS eignen. Im Rahmen dessen entstand eine Art „proof of concept sheet“, welches sich im Anhang B befindet. Hier war es das Ziel, drei mögliche Umsetzungen gegenüberzustellen um so Vor- und Nachteile besser ausarbeiten zu können. Die Unterschiede der drei Variationen lagen vor allem im Funktionsumfang und im Preis. Im Folgenden werden diese drei Varianten kurz beschrieben und verglichen.

Der günstigste Lösungsansatz besteht darin, ein vorhandenes System von einem Zulieferer zu adaptieren und zu versuchen auf die Soundgenerierung Einfluss zu nehmen. Hier sind diese Systeme bereits mit einem Lautsprecher und einem Verstärker ausgestattet, sodass die Hardware über dieses Komplettsystem abgedeckt ist. Da sich die klanglichen Gestaltungsmöglichkeiten nach Rücksprache mit dem Hersteller des Systems als sehr schwierig herausstellen und in dieser Arbeit ein höherer Eigenanteil an der Entwicklung liegen soll, fiel dieser Ansatz früh aus der Auswahl.

Der mittlere Lösungsansatz bestand aus der Programmierung eines Mikrocontrollers. Hier ist weitere Hardware in Form eines Verstärkers und eines Lautsprechers notwendig. Softwareseitig gibt es dafür zwei Lösungsmöglichkeiten. Zum einen über einen programmierten Synthesizer, welcher über die Steuersignale des Fahrzeugs bestimmte Parameter des Synthesizers steuert. Zum anderen durch Samples, die von einem Speicher abgespielt werden und ebenfalls in beispielsweise ihrer Abspielgeschwindigkeit verändert werden. Der Vorteil dieses Lösungsansatzes besteht darin, dass das System von vornherein eingebettet funktioniert. Es ist allerdings unklar, wie gut die Programmierung eines Synthesizers auf einem Mikrocontroller klanglich möglich ist. Gleichzeitig sind beispielsweise Pitch Operationen nur komplex im Code umzusetzen und erfordern deshalb einen hohen Zeitaufwand und Knowhow, besonders auch hinsichtlich der resultierenden Qualität der Tonhöhenverschiebung.

Bei der speicherbasierenden Lösung mit Samples muss aufgrund von geringen Speicherkapazitäten außerdem das Audiofile mitunter stark komprimiert werden. Dadurch kann die Tonqualität äußerst verlustbehaftet sein, man muss den Klang also stark an

das System anpassen, was gleichzeitig die Freiheiten im Sounddesign etwas einschränkt.

Im preislich oberen Segment der Lösungsansätze steht ein PC-Stick. Auch hier ist weitere Hardware in Form eines Lautsprechers und eines Verstärkermoduls nötig. Softwareseitig ist es die komfortabelste Lösung, da es hier nahezu grenzenlose Möglichkeiten gibt. Daher werden zwei Softwarelösungen in Erwägung gezogen, die bekannt dafür sind, Sound im interaktiven Kontext wiederzugeben und zu verarbeiten. Eine Möglichkeit bietet Max/MSP, mit dem es möglich ist, mithilfe von Codeblöcken Synthesizer zu erstellen und diese durch externe Steuersignale in ihren Eigenschaften und Parametern zu verändern. Eine Alternative bietet Unity3D in Kombination mit dem Plug-In FMOD. Unity3D ist als Game-Engine ein äußerst geeignetes Tool, um interaktive Medien erstellen zu können. FMOD ist eine Erweiterung, mit der es auf eine komfortable Weise möglich ist, komplexe, interaktive Automationskurven zu erstellen und tiefe Bearbeitungsmöglichkeiten für Audiosamples zu haben. Auf letzteren Ansatz wird in diesem Kapitel später genauer eingegangen.

Bei allen Lösungsansätzen ist es unumgänglich eine Schnittstelle zwischen den Fahrzeugdaten und dem Controller bzw. der Rechereinheit herzustellen. Dies kann bei der Controller-Lösung beispielsweise eine CAN-Erweiterungsplatine sein. Bei der PC-Lösung gibt es hingegen CAN-Dongle, die es ermöglichen, das Fahrzeugsignal per USB-Verbindung mit entsprechenden Treibern auszulesen.

Zusammengefasst ist die Lösung mit dem höchsten finanziellen Aufwand gleichzeitig die flexibelste für einen möglichst variablen Prototypen. Mit den genannten Softwarelösungen ist eine nahezu unendliche Bandbreite an möglichen Klängen verfügbar, die sich außerdem auch gut ansteuern lassen.

Neben der Rechereinheit bzw. dem Controller gilt es noch einen passenden Lautsprecher auszuwählen. Wichtig ist es hierbei einen möglichst breitbandigen Lautsprecher zu finden, der auch dazu in der Lage ist, einen angemessenen Schalldruck wiederzugeben. Auf diese Weise ist es möglich, ein breites Spektrum abzudecken und somit auch Geräusche in einem tieffrequenteren Spektrum abspielen zu können. Gleichzeitig sollte er robust und wetterfest sein, um auch bei schlechten Witterungsverhältnissen nicht in seiner Funktionalität eingeschränkt zu sein. Zudem wird noch eine Verstärkereinheit

benötigt, damit das Signal auf einen entsprechenden Lautstärkepegel gebracht werden kann.

Letztlich fiel die Wahl auf eine reduziertere Form vom dritten Lösungsansatz. Statt eines Computersticks wurde ein RaspberryPi 3 verwendet. Die Software, welche die Fahrzeugdaten einliest und verarbeitet, soll mit Unity3D in Kombination mit FMOD geschrieben werden. Da Unity3D in der Lage dazu ist eine ausführbare apk-Datei für Android-Systeme zu kompilieren, wurde der RaspberryPi mit einem frei verfügbaren Android Betriebssystem ausgestattet.

5.3 Herausforderungen und Probleme

Die auftretenden Probleme lassen sich grob in Hard- und Software aufteilen. Folgend wird zunächst der Entwicklungsprozess der Software erläutert und im Anschluss der der Hardware.

Da die Software mithilfe von Unity3D geschrieben wurde, liegt der Entwicklung die Programmiersprache C# zugrunde. Zunächst ist es wichtig die spätere Situation zu simulieren. Dafür wurden Daten von einem Arduino über eine serielle USB Schnittstelle an Unity3D geschickt. Auf diese Weise war es möglich eingehende Daten zu simulieren und diese mit Werten zu versehen, die dem späteren Signal ähnlich sind. So konnten die ersten Samples in FMOD angelegt und über ein Potentiometer, welches am Arduino angeschlossen ist, interaktiv verändert werden, indem der verknüpfte Parameter die Lautstärke reguliert.

Anschließend ging es darum, den Arduino durch die CAN-Schnittstelle zu ersetzen und das Programm dementsprechend anzupassen. Es musste zunächst die Bibliothek implementiert werden, welche es Unity ermöglicht auf den CAN-Dongle zuzugreifen, um so die Fahrzeugdaten auslesen zu können. Weiterhin musste definiert werden, welche konkreten CAN-Botschaften eingelesen und verarbeitet werden sollen. Diese Werte mussten im nächsten Schritt an FMOD weitergegeben werden, sodass die Parameter in FMOD angesteuert werden können.

In diesem Stadium läuft das Programm noch auf einem Notebook. Unity3D ist dazu in der Lage die programmierte Software aus der Engine heraus zu starten, ohne dafür eine ausführbare Programmdatei erstellen zu müssen. Dadurch ist es möglich schnell

Softwareänderungen zu testen. Gleichzeitig können bei der späteren Ausgabe der ausführbaren Datei unter Umständen Probleme entstehen, die zunächst durch das sichere ausführen innerhalb der Engine nicht auffallen.

Nachdem die Software über die Engine und das Notebook funktionierte, war es das Ziel das Notebook durch den RaspberryPi zu ersetzen. Dazu war es notwendig das Programm als „Android application package“ (APK) - Datei auszugeben, da das RaspberryPi eine frei verfügbare Android-Version als Betriebssystem nutzt.

Es war notwendig das Problem über ein Android-Betriebssystem zu lösen, da es die einzige Plattform ist, für die Unity3D eine ausführbare Datei ausgeben kann, welche auch mit dem RaspberryPi kompatibel ist. Nachdem die Software als APK ausgegeben wurde, stellte sich heraus, dass die Applikation auf dem RaspberryPi große Performance Probleme hervorruft. Gleichzeitig konnte die CAN-Bibliothek, welche im folgenden Kapitel näher erläutert wird, nicht erkannt werden. Der Hersteller des CAN-Dongles und der korrespondierenden Softwarebibliothek bietet keine Lösungen für Android-Betriebssysteme an. Daher war es nicht möglich, das AVAS basierend auf dem RaspberryPi umzusetzen. Im anschließenden Kapitel wird die finale Herangehensweise beschrieben und näher erläutert.

Bei der Audioverbindung von der Recheneinheit zum Lautsprecher traten mehrfach Probleme auf. Da das Signal per Klinkenkabel vom PC teilweise an der Fahrzeugbatterie vorbeigeführt wurde, kam es zu starken Einstreuungen, welche das Signal komplett unkenntlich machten. Auch durch unterschiedliche Positionierung des Rechners und des Lautsprechers, um den Kabelweg zu verkürzen, kam es zu Einstreuungen. Diese waren zwar nicht mehr so stark, jedoch stark genug um das Nutzsignal mitunter schwer zu beeinträchtigen. Es konnte durch direktes Anschließen des Lautsprechers an beispielsweise den Computer ausgeschlossen werden, dass die Probleme bereits innerhalb des Verstärkers oder des Lautsprechers lagen.

Daher musste dieses Problem in der finalen Lösung ebenfalls adressiert werden und wird gegen Ende des folgenden Kapitels erläutert.

5.4 Aufbau des Prototyps

5.4.1 Hardware

Letztlich fiel die Entscheidung der Recheneinheit auf einen Einplatinencomputer, der auf Windows 10 läuft. Auf diese Weise ist es möglich, das AVAS auf dieselbe Plattform zu bringen wie die Testumgebung. Dadurch entstehen wenig bis gar keine Portierungsprobleme der Software. Der Einplatinencomputer nennt sich LattePanda und wird von dem gleichnamigen Unternehmen hergestellt. Technisch ist er mit einem Quad Core Prozessor mit 1,8GHz und 4GB Arbeitsspeicher ausgestattet. Gleichzeitig hat er mehrere USB3-Ports und bietet Verbindungsmöglichkeiten via Bluetooth und WLAN.²⁶ Damit übertrifft er die minimalen Anforderungen des Systems zwar weit, ist dadurch aber auch die optimale Plattform für einen Prototypen. Es bieten sich viele Möglichkeiten das System auszureizen und beispielsweise Use Cases zu testen, die eine höhere Performance erfordern.

So musste aus Unity3D eine unter Windows10 ausführbare Datei (.exe) ausgegeben werden. Auch die CAN-Bibliothek für die Verbindung zum Fahrzeug per USB-CAN-Dongle ist für Windows verfügbar. So entstand eine performante, platzsparende Lösung für die Recheneinheit des AVAS. Der PC führt beim Start automatisch die Software für das AVAS aus und verbindet sich über den CAN-Dongle mit dem Fahrzeug.

Zum besseren Verständnis folgt eine Blockabbildung des Signalflusses des AVAS.

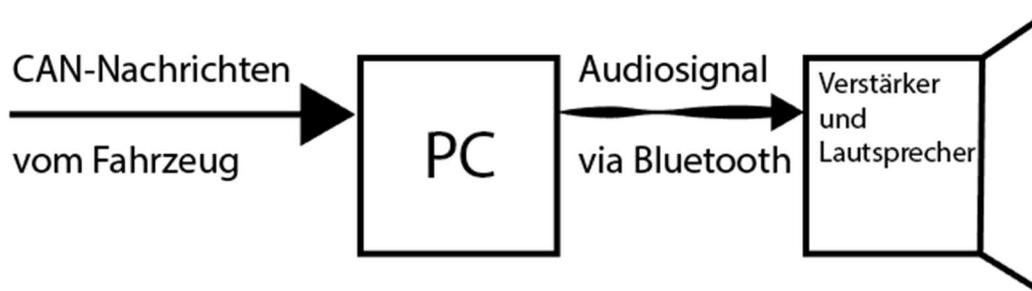


Abbildung 13 Blockschaltbild des Signalflusses im AVAS

²⁶ <https://www.lattepanda.com/products/lattepanda-4g-64g-with-enterprise-license.html>

Im nächsten Schritt war es wichtig einen Verstärker zu finden, der Bluetooth fähig ist um die Problematik der Einstreuung umgehen zu können. Dabei kommt eine günstige Verstärkerplatine mit Bluetooth Eingang zum Einsatz.²⁷ Zusätzlich besteht sogar die Möglichkeit einen weiteren Lautsprecher anzuschließen, um so beispielsweise zusätzlich einen nach hinten abstrahlenden Lautsprecher zu haben. Allerdings ist dies bei einem AVAS, das auf Rollern zum Einsatz kommt, nicht relevant, da diese üblicherweise nicht rückwärtsfahren. Abgesehen von den Anschlussmöglichkeiten des Verstärkers bietet er eine Leistung von 25W pro Kanal, was für den Zweck des AVAS ausreichend ist.

Als Lautsprecher wurde ein Einbaulautsprecher der Firma „Visaton“ verwendet. Dieser kann die vollen 25W des Verstärkerkanals ausschöpfen und ist außerdem wetterfest. Das Datenblatt des Lautsprechers verspricht eine hohe Frequenzbandbreite zwischen 90 Hz und 19 kHz.²⁸ Das bietet in diesem prototypischen Aufbau die notwendige Bandbreite, um Klänge in unterschiedlichen Frequenzbereichen erstellen und beurteilen zu können.

Die Verstärkerplatine und der Lautsprecher wurden in ein Gehäuse verschraubt, so dass man den Lautsprecher besser an das Fahrzeug montieren kann, ohne dass der Verstärker immer wieder neu verkabelt und positioniert werden muss. In Anhang A.3 befinden sich Fotos der Lautsprecherbox. Hier gilt es zu erwähnen, dass die Größe und Beschaffenheit der Kunststoffbox Auswirkungen auf die klangliche Eigenschaft des Lautsprechers haben. Genauso hat auch der Verstärker einen klanglichen Einfluss auf das System. Der Frequenzverlauf des Lautsprechers ist also nur noch ein grober Anhaltspunkt für die Eigenschaften des AVAS.

Dies wird allerdings in Kauf genommen, da der Lautsprecher sonst, ohne den Roller selbst zu verändern, nicht hätte angebracht werden können.

Die Stromversorgung des Verstärkers und des Rechners mussten aufgrund von unterschiedlichen Anforderungen auf zwei Wegen gelöst werden. Der Verstärker wurde direkt an das 12V Bordnetz des Rollers angeschlossen und startete dadurch mit der Zündung des Fahrzeugs. Da eine 12V Spannung für den LattePanda zu hoch ist, kam dafür

²⁷ <https://www.amazon.de/gp/product/B07DK5JHX3/>

²⁸ <https://at.rs-online.com/web/p/einbau-lautsprecher/3643240/>

eine handelsübliche Powerbank zum Einsatz. So ist zwar die Dauer der Nutzung des AVAS begrenzt, jedoch treten keine Einstreuungen durch einen Transformator auf und das System bleibt mobil. Die Betriebsdauer ist außerdem ausreichend für mehrere Stunden und damit auch für die Nutzertests und Probefahrten. In Anhang A.3 befinden sich Bilder des Aufbaus am Fahrzeug.

Da sich der CAN-Anschluss des Fahrzeugs unter dem Sitz befindet, wurde auch der CAN-Dongle dementsprechend unter dem Sitz angeschlossen, sodass lediglich ein USB-Kabel von außen unter den Sitz geführt werden muss. Dadurch wird der Schließmechanismus des Sitzes nicht beeinträchtigt und eine sichere Fahrt bleibt gewährleistet.

Der Rechner und die Powerbank sind mithilfe von Klebeband hinter dem Sitz an einer Metallgriffstange befestigt und lassen sich so auch wieder schnell entfernen.

Dies war aufgrund von vielen unterschiedlichen Tests, denen die Fahrzeuge auf täglicher Basis unterzogen werden, notwendig. So konnte ein schneller Auf- und Abbau des AVAS gewährleistet werden, sodass andere Abteilungen nicht in ihren Tests eingeschränkt sind.

5.4.2 Software

Bei der Umsetzung der Software wird nicht auf den vollständigen Quellcode eingegangen, da ein Großteil davon auf einer Bibliothek beruht, welche es ermöglicht, die CAN-Daten des Fahrzeugs mithilfe eines CAN-Dongles auszulesen. Der Dongle und die „pcan“- Bibliothek stammen von der Firma PeakCAN.²⁹

Aus diesem Grund wird folgend nur der Teil des Codes beschrieben, welcher die CAN-Daten verarbeitet und in Befehle für Unity bzw. FMOD umwandelt.

Zu Beginn werden alle für FMOD notwendigen Events und Parameter erstellt, auf die zugegriffen werden soll. Event-Instanzen entsprechen in FMOD einem Audioereignis.

```
public class CANtoUnity : MonoBehaviour {  
    FMOD.Studio.EventInstance Drive_3;  
    FMOD.Studio.EventInstance Hupe_3;  
  
    FMOD.Studio.ParameterInstance Velocity;  
    FMOD.Studio.ParameterInstance HupeAN;  
    FMOD.Studio.ParameterInstance HupeMode;  
    FMOD.Studio.ParameterInstance Gaspedal;  
    FMOD.Studio.ParameterInstance Fahrmodus;
```

Abbildung 14 Anlegen der Event- und Parameterinstanzen

Bei diesem AVAS gibt es zwei verschiedene Ereignisse, die für die Klangerzeugung relevant sind – Fahren, welche der Drive_3 Event-Instanz entspricht und Hupen, welche der Hupe_3 Event-Instanz entspricht.

Jedes Event kann innerhalb von FMOD auf Parameterinstanzen zugreifen, welche auf klangformende Parameter Einfluss nehmen können. Daher gibt es für jede Variable, welche das Klangfeedback beim Fahren beeinflussen soll, eine eigens angelegte FMOD-Parameterinstanz.

Dabei greift die Instanz „Velocity“ auf die Geschwindigkeit zu, „HupeAN“ liest aus, ob die Hupe aktiv ist, „HupeMode“ erkennt den Fahrmodus, „Gaspedal“ reagiert auf die Gaspedalstellung und „Fahrmodus“ erkennt ebenfalls den Fahrmodus.

²⁹ <https://www.peak-system.com/PCAN-USB.199.0.html>

Der Fahrmodus ist doppelt belegt, da Parameterinstanzen exklusiv sind und jede Parameterinstanz nur einer Eventinstanz zugewiesen werden kann.

Dies wird anhand der folgenden Abbildung verdeutlicht.

```
// Use this for initialization
private void Awake()
{
    Drive_3 = FMODUnity.RuntimeManager.CreateInstance("event:/SoundSet_3/Drive_3");
    Drive_3.getParameter("Velocity", out Velocity);
    Drive_3.getParameter("Gaspedal", out Gaspedal);
    Drive_3.getParameter("Fahrmodus", out Fahrmodus);

    Hupe_3 = FMODUnity.RuntimeManager.CreateInstance("event:/SoundSet_3/Hupe_3");
    Hupe_3.getParameter("HupeAN", out HupeAN);
    Hupe_3.getParameter("HupeMode", out HupeMode);
}
```

Abbildung 15 Aufrufen der Event- und Parameterinstanzen

In der Awake Methode werden die Instanzen der Events und der Parameter aufgerufen und gestartet. Gleichzeitig werden innerhalb der „getParameter“ Klammern die Werte in Variablen gespeichert, die exakt nach den entsprechenden Parameterinstanzen benannt sind. Der CreateInstance-Befehl greift auf das Event innerhalb von FMOD zu und benötigt dazu den in der Klammer gegebenen Pfad zum jeweiligen Event.

Wie zuvor beschrieben verbirgt sich hinter „Fahrmodus“ und „HupeMode“ dieselbe CAN-Botschaft. Die Instanz muss jedoch einmal für das Drive_3-Event aufgerufen werden und einmal für das Hupe_3-Event, weil die Parameterinstanzen ausschließlich exklusiv für ein Event verwendet werden können.

```
void Start() {
    btnInit_Click();
    print("Start");

    FMODUnity.RuntimeManager.AttachInstanceToGameObject(Drive_3, GetComponent<Transform>(), GetComponent<Rigidbody>());
    Drive_3.start();

    FMODUnity.RuntimeManager.AttachInstanceToGameObject(Hupe_3, GetComponent<Transform>(), GetComponent<Rigidbody>());
    Hupe_3.start();
}
```

Abbildung 16 Fehlermeldung und Events an Gameobject anfügen

In Abbildung 16 wird die „Start“-Methode aufgerufen. Hier wird das Programm gestartet, zu Debugging Zwecken wird ein String mit „Start“ ausgegeben.

Noch davor wird die „btnInit_Click“ Methode aufgerufen. Sie ist dafür zuständig den pcan-Handler zu initialisieren und damit das Auslesen der CAN-Daten durch die

PeakCAN-Bibliothek zu ermöglichen. Gleichzeitig beinhaltet diese Methode auch eine Fehlerausgabe für das Debugging.

Des Weiteren müssen die Eventinstanzen noch an ein sogenanntes Gameobject gehängt werden, damit sie ausgeführt werden. Das muss ausschließlich aus Syntax Gründen in Unity und FMOD geschehen.

Nachdem das Programm gestartet wurde, wird in Abbildung 17 die Update Methode aufgerufen. Sie führt die „ReadMessage“ - Methode in einer Schleife aus, bis das Programm beendet wird. Falls es beendet wird oder unerwartet abstürzen sollte, wird die „OnDestroy“- Methode aufgerufen. Sie beendet zum einen das Auslesen der CAN-Daten mithilfe der „btnRelease_Click“ Methode aus der PeakCAN Bibliothek. Des Weiteren wird ein String ausgegeben, der auf der Debugging Konsole das erfolgreiche Beenden der Software wiedergibt.

```
// Update is called once per frame
void Update() {
    ReadMessage();
    // print("Update");
}

private void OnDestroy()
{
    btnRelease_Click();
    print("destroy");
}
```

Abbildung 17 Update und „OnDestroy“ Methoden

Die folgenden Abbildungen des Codes befinden sich alle innerhalb der „ReadMessage“ Methode, welche dafür zuständig ist, die CAN-Daten permanent auszulesen und zu interpretieren.

Zu Beginn der Methode werden zunächst alle notwendigen Variablen definiert und initialisiert (Abb.18). In „speed“, „direction“ und „app“ werden die Werte für die Geschwindigkeit, Fahrtrichtung und Gaspedalstellung gespeichert. Außerdem wird der Fahrmodus über „driveMode“ in einer Integer-Variablen gespeichert. Jede dieser Variablen bekommt außerdem noch einen Rückgabewert, auf diesen wird später im Kapitel noch genauer eingegangen. Des Weiteren werden die Tasten der Fernbedienung am Lenker des Rollers definiert.

```
public TPCANStatus ReadMessage()
{
    TPCANMsg CANMsg;
    TPCANTimestamp CANTimeStamp;
    TPCANStatus stsResult;
    double speed = 0;
    int direction = 0;
    int app = 0;
    float return_value_speed = 0;
    float return_value_hupe = 0;
    float return_value_gaspedal = 0;
    float return_value_driveMode = 0;

    int driveMode;

    int RemoteUp;
    int RemoteConfirm;
    int RemoteDown;
```

Abbildung 18 Definieren der notwendigen Variablen

In Abbildung 19 wird zunächst unter der Kommentarzeile SPEED die Geschwindigkeit ausgelesen. Die if-Bedingung bezieht sich auf das Vorhandensein der CANMsg.ID == 0x15. Die CAN-Botschaften des Fahrzeugs haben alle eine ID, die durch einen hexadezimalen Code definiert ist. Innerhalb der Botschaften befinden sich mehrere Informationen über den Zustand, in dem sich das Fahrzeug aktuell befindet. Um die Geschwindigkeit auslesen zu können, muss das richtige Bit der CAN Botschaft gelesen werden. Dazu dient die bitweise Verschiebung, mit der die Variable „speed“ gleichgesetzt wird. Die Geschwindigkeit wird in 2 Bits der CAN-Botschaft gespeichert und übertragen. Im Anschluss wird die Variable „speed“ in die entsprechende Variable „Velocity“ von FMOD übertragen. Dies geschieht mithilfe des „set.Value“ Befehls. Innerhalb der Klammern des Befehls kann noch der Variablentyp bestimmt werden, mit dem FMOD weiterarbeiten soll. Mithilfe des „get.Value“ Befehls wird der Wert der Geschwindigkeit außerdem noch in die Rückgabewariable geschrieben. Diese wird dann im letzten Schritt an die Debug-Konsole von Unity weitergegeben, sodass etwaige Fehler in den Werten schnell erkannt werden können.

```
//SPEED
if (CANMsg.ID == 0x15)
{
    speed = 0.5 * (2 * CANMsg.DATA[2] + (CANMsg.DATA[1] >> 7));
    Velocity.setValue((float)speed);
    Velocity.getValue(out return_value_speed);
    // print(return_value_speed);
    UnityEngine.Debug.Log("Speed" + return_value_speed);

// FAHRMODI

    // Wertebeschreibung 0=Go 1=Cruise 2=Boost 3=nicht verwendet
    driveMode = ((CANMsg.DATA[1] & 0x30) >> 4);
    Fahrmodus.setValue((float)driveMode);
    Fahrmodus.getValue(out return_value_driveMode);
    print(return_value_driveMode);
    UnityEngine.Debug.Log("Speed" + return_value_driveMode);
}
```

Abbildung 19 Geschwindigkeit und Fahr-Modi

Identisch aufgebaut wie das Auslesen der Geschwindigkeit in Abbildung 19 ist auch das Auslesen des Fahrmodus. Der Wertebereich liegt hier zwischen 0 und 3, da die Fahr-Modi des Fahrzeugs innerhalb der CAN-Botschaft mit diesen Werten arbeiten. Dabei ist

3 in diesem Fall nicht belegt, da es lediglich drei relevante Fahr-Modi gibt.

Das Auslesen des Fahrmodus befindet sich in der gleichen if-Schleife wie die Geschwindigkeit, weil beide Informationen in derselben CAN-Botschaft gesendet werden, lediglich in verschiedenen Bytes innerhalb der Botschaft.

Der „print“ Befehl dient hierbei ausschließlich zum Debugging auf Ebene von der Entwicklungsumgebung Visual Studio und hat daher keine Funktion im Zusammenhang mit der Klangerzeugung.

```
// DIRECTION 0=neutral 1=forward 2=reverse 3=parking
if (CANMsg.ID == 0x64)
{
    direction = ((CANMsg.DATA[3] & 0x30) >> 4);

    // BUTTONS

    // UP
    RemoteUp = ((CANMsg.DATA[3] & 0x04) >> 2);

    // CONFIRM
    RemoteConfirm = ((CANMsg.DATA[3] & 0x02) >> 1);
    HupeAN.setValue((float)RemoteConfirm);
    HupeAN.getValue(out return_value_hupe);
    // print(return_value_hupe);

    // DOWN
    RemoteDown = (CANMsg.DATA[3] & 0x01);
}
```

Abbildung 20 Fahrtrichtung und Fernsteuerung

In Abbildung 20 wird eine neue if-Schleife angelegt, um eine weitere CAN-Botschaft auszulesen. In dieser befinden sich die Informationen zur Fahrtrichtung und zur Steuerung der Fahr-Modi am Lenker. Die Fahrtrichtung wird zu Testzwecken ausgelesen und triggert keinerlei Ereignisse in FMOD und hat demnach keinen Einfluss auf die Generierung des Fahrgeräuschs. Hier wäre es beispielsweise aber möglich, ein dediziertes Fahrgeräusch im Rückwärtsgang abzuspielen, was besonders bei größeren Fahrzeugklassen zum Tragen kommen wird, da diese rechtlich dazu verpflichtet sind.

Die Steuerung des AVAS wurde an die Tasten des Lenkers gekoppelt, welche für die

Steuerung der Fahr-Modi bzw. des Human-Machine-Interfaces (HMI) zuständig sind. Auch hier ist wieder zu sehen, dass zunächst bitweise Verschiebungen durchgeführt werden müssen, um die korrekten Bytes der CAN-Botschaft auszulesen.

Die Knöpfe „UP“ und „DOWN“ haben in diesem Fall keine direkte Auswirkung auf den Klang, obwohl sie den Fahrmodus wechseln. Da der Fahrmodus allerdings, wie in Abbildung 19 zu sehen ist, eigenständig ausgelesen wird, ist die Änderung des Klanges vom Fahrmodus abhängig und nicht vom Tastendruck auf der Steuerung.

Lediglich die „CONFIRM“ Taste wurde dafür verwendet, die „freundliche Hupe“ betätigen zu können. Dementsprechend wird hier für die „HupeAN“ Variable von FMOD der Wert „RemoteConfirm“ auf 1 gesetzt, da die Tasten der Steuerung binär funktionieren. Auch hier wird im Anschluss zur Überprüfung der Wert aus FMOD zurückgegeben.

```
// FAHRPEDALSTELLUNG 0-100
if (CANMsg.ID == 0x83)
{
    app = (CANMsg.DATA[5] >> 2)*2;
    Gapedal.SetValue((float)app);
    Gapedal.GetValue(out return_value_gapedal);
    // print(return_value_gapedal);
    UnityEngine.Debug.Log("Speed" + return_value_gapedal);
}
```

Abbildung 21 Fahrpedalstellung

In der letzten if-Schleife in Abbildung 21 befindet sich die CAN-Botschaft, in der die Fahrpedalstellung übertragen wird. Sie entspricht der Stellung des Gashebels, welche prozentual zwischen 0 und 100 übergeben wird. Wie bei den vorangegangenen Parametern wird der Wert in der FMOD-Variablen gespeichert, um Anschluss zurückgegeben und in den Unity Debug Log geschrieben.

Zusammengefasst werden lediglich die Parameter Geschwindigkeit, Fahrmodus, Gaspedalstellung und die „Confirm“ Taste des Lenkers. Sie werden an FMOD weitergegeben und sind damit relevant für die Klangerzeugung.

6 Erstellung der Soundpatterns

6.1 Grundlagen der elektronischen Klangerzeugung

6.1.1 Grundelemente der Klangerzeugung

Elektronische Klangerzeuger lassen sich im Wesentlichen auf wenige Grundelemente herunterbrechen. Zu Beginn wird mithilfe eines Oszillators ein Signal erzeugt. Die Grundfrequenz des erzeugten Signals legt dabei die Tonhöhe fest. Je nach Art der Erzeugung spricht man von unterschiedlichen Methoden der elektronischen Klangerzeugung. Dabei unterscheidet sich beispielsweise die Obertonstruktur und damit der wahrgenommene Klang.

Der nächste Baustein eines Klangerzeugers sind Filter. Wichtige Kernparameter von Filtern sind die obere und untere Grenzfrequenz, sie schränken das Signal in seinem Frequenzspektrum ein. Zumeist ist auch eine Resonanzfrequenz einstellbar, diese verstärkt das Signal zusätzlich bei einer bestimmten Frequenz bzw. dessen Obertöne.

Eine Verstärkung des Signals ist allerdings auch mit einem nachfolgend geschalteten Verstärker möglich. Mit ihm lässt sich die Gesamtlautstärke des Signals verändern. Der Verstärker lässt sich in der Regel auch abhängig von der Zeit einstellen, sodass aus einem gleichbleibenden Rauschsignal ein perkussives Element wird.³⁰

Über die sogenannten Hüllkurven kann man ein ähnliches Ergebnis erreichen. Die Hüllkurve besteht aus Attack, Decay, Sustain und Release, daher auch häufig als ADSR-Kurve bezeichnet. Der Attack gibt dabei an, wann das Signal den Spitzenpegel erreicht, der Decay, wann das Signal auf den Pegel des eingestellten Sustain-Wertes sinkt und der Release die Ausklingzeit. So lässt sich das Signal im zeitlichen Verlauf recht umfangreich bearbeiten.³¹

Der letzte Baustein der Klangerzeugung sind die sogenannten Low-Frequency-Oscillator, LFOs. Dabei handelt es sich um einen niederfrequenten Oszillator, welcher beispielsweise Parameter eines Filters innerhalb eines Intervalls kontinuierlich verändern

³⁰ Vgl. Hannes Raffaseder, 214-216

³¹ <https://www.bonedo.de/artikel/einzelansicht/synthesizer-und-sounddesign-3-huellkurven.html>

kann. Die Frequenz des LFOs legt fest, wie häufig die Veränderung innerhalb des Zeitintervalls ausgeführt wird. Die Amplitude legt die Höhe der Veränderung des eingestellten Parameters fest.³²

6.1.2 Wichtige Verfahren der elektronischen Klangerzeugung

Additive Klangerzeugung

Der additiven Klangerzeugung liegt vor allem die Fourier-Analyse zugrunde. Durch additive Überlagerungen verschiedener harmonischer Schwingungen werden Klänge erzeugt. Hierbei kann man unmittelbar Einfluss auf jede klangformende Komponente nehmen, was Vorteile in der Benutzbarkeit schafft. Außerdem lassen sich dadurch auch Obertonstrukturen von bekannten Instrumenten grob nachempfinden. So kann man beispielsweise die Klänge eines Pianos bis zu einem bestimmten Grad imitieren. Ein Nachteil ist allerdings, dass man bei komplexer generierten Klängen auch schnell den Überblick über die klangformenden Komponenten verliert. Gleichzeitig steigt auch der Rechenaufwand deutlich je mehr Schwingungen erzeugt werden. Besonders in der Anfangsphase der elektronischen Klangerzeugung konnten hiermit allerdings, sehr schnell viele neue Klangwelten entdeckt bzw. bereits Bekanntes ein Stück weit elektronisch nachgebildet werden. Heute wird die additive Klangerzeugung allerdings eher selten verwendet, da andere Verfahren effizienter und intuitiver in der Handhabung sind und häufig auch qualitativ hochwertiger klingen.³³

Subtraktive Klangsintese

Die subtraktive Klangsintese arbeitet mit Oszillatoren, welche eine vorgegebene Wellenform besitzen. Klassische Wellenformen zur Klangerzeugung sind beispielsweise Sinus, Dreieck, Rechteck und Sägezahn. Außerdem ist es meistens auch möglich ein Rauschen zu erzeugen.

Der Klang wird also aus dem vorhandenen Oszillator über z.B. Filter geformt. Die subtraktive Klangsintese ist dadurch sehr nachvollziehbar und übersichtlich. Sie kommt mit wenigen Parametern aus, welche in einfache Blöcke unterteilt werden können. Die Wahl der Grundwellen spielt hier aber eine größere Rolle, da sie das komplette

³² <https://www.delamar.de/faq/was-ist-ein-oszillator-29487/>

³³ Vgl. Hannes Raffaseder, S.220-221

Klangmaterial des finalen Geräuschs bereits beinhalten müssen.

Die subtraktive Klangsynthese hat in der Geschichte der Klangerzeugung einen hohen Stellenwert, da die ersten analogen Synthesizer mit dieser Form der Klangerzeugung ausgestattet waren. Doch auch bis heute sind analoge Klangerzeuger zumeist mit einer subtraktiven Klangerzeugung ausgestattet.³⁴

Wavetable Synthese

Die Wavetable Synthese erweitert die Möglichkeiten der subtraktiven Erzeugung, indem sie weitere Wellenformen als Ausgangsmaterial anbietet. Viele dieser Wellenformen werden meist in 32 bzw. 64er Blöcke – Wavetable, zusammengefasst. Dabei kann das Signal von einem Oszillator zum nächsten springen, auf diese Weise bietet sich eine sehr experimentelle Herangehensweise an. Dabei ist der Klang abhängig davon, wie sehr sich in der Wavetable nebeneinanderstehende Wellenformen voneinander unterscheiden bzw. wie sehr sie sich ähneln. Je ähnlicher, desto weicher der Klang und je unterschiedlicher, desto härter und sprunghafter. Durch die Wavetable-Synthese lassen sich sehr schnell experimentelle und interessante Klänge erzeugen.³⁵

Sampling

Das Sampling basiert auf einem digital aufgezeichneten, akustischen Ereignis. Dieses Signal kann per Tastendruck auf beispielsweise einem MIDI-Keyboard wiedergegeben werden und somit in seiner Tonhöhe verändert oder mehrstimmig gespielt werden. Das initiale Ziel von Sampling, war es den Klang eines Instrumentes aufzunehmen und es zu „samplen“, sodass es über den Synthesizer nachgebildet werden kann. Dies funktionierte in der Anfangszeit nur bedingt, es wurde einfach die Abspielgeschwindigkeit des Ausgangsmaterials verändert, um eine Tonhöhenverschiebung zu erreichen. Dadurch wurden allerdings auch die klangformenden Eigenschaften und das gesamte Spektrum verändert. Das führte dazu, dass der Klang nicht mehr nach dem ursprünglich aufgenommenen Instrument klang, sondern nach einer äußerst künstlich klingenden Nachbildung.³⁶

³⁴ <https://www.delamar.de/faq/syntheseformen-klangsynthese-34050/#subtraktiv>

³⁵ <https://www.amazona.de/was-genau-ist-ein-synthesizer-synthesen-im-ueberblick/>

³⁶ Vgl. Raffaseder, S. 224

6.2 Fahrgeräusch

Die Fahrgeräusche wurden wegen der hohen Flexibilität vor allem mithilfe von subtraktiver Klangsynthese erstellt. Eingangs bestand noch die Idee auch samplebasierende Geräusche zu erstellen um auf diese Weise einen klassischen Verbrennungsmotor zu imitieren. Allerdings hat sich schnell herausgestellt, dass sich die Arbeit mit Samples einerseits im Workflow mit FMOD nicht besonders gut eignen, da sie eine besonders hohe Vor- bzw. Nachbereitung erfordern um loopbar zu sein. Auf der anderen Seite hat die Umfrage aus Kapitel 3.4 ergeben hat, dass ein konventionelles Motorengeräusch nicht erstrebenswert ist. So war es das Ziel ein Geräuschpattern zu erstellen, welches bei sehr guter Ortung gleichzeitig eine hohe allgemeine Akzeptanz findet.

Im Folgenden werden jene drei Geräusch Patterns näher beschrieben, die es in die Endauswahl für die anschließende Nutzerstudie geschafft haben. Die Auswahl erfolgte durch kleine Probefahrten auf der Teststrecke und eigene Präferenzen bzw. gute Übertragbarkeit auf den Straßenverkehr. Diese Übertragbarkeit ist gemessen an der subjektiven Differenzierbarkeit des Klanges in einer Verkehrssituation bzw. der vorhandenen Teststrecke.

In folgender Abbildung ist der Workflow für die einzelnen Soundpatterns als Blockdiagramm abgebildet.

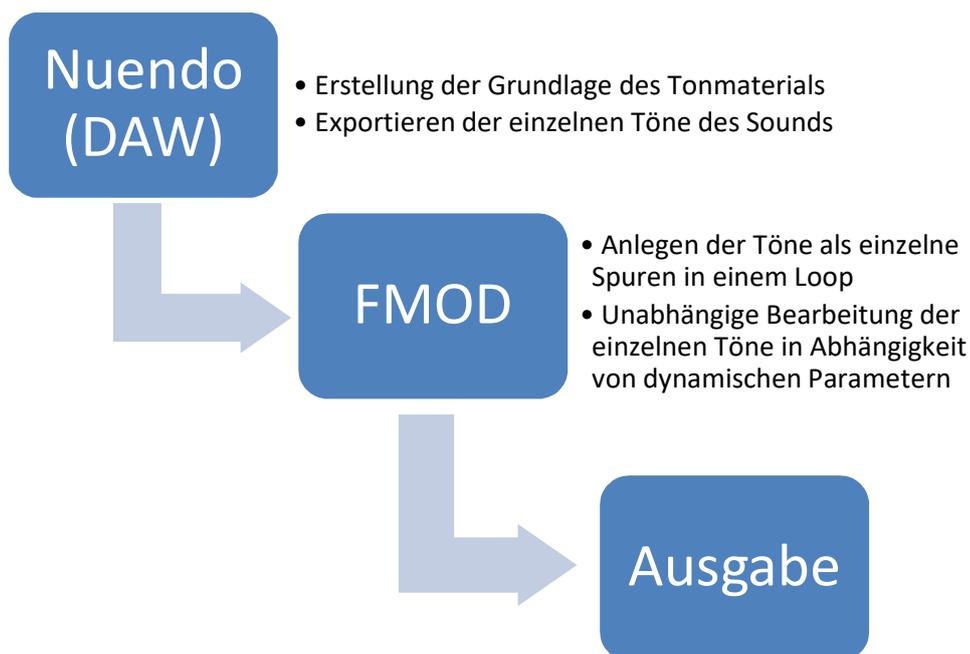


Abbildung 22 Blockdiagramm des Sounddesign Workflows

Soundpattern 1

Die Basis für das erste Soundpattern wurde mithilfe des Software Synthesizers Massive X erstellt. Wie eingangs erwähnt handelt es sich hierbei um eine subtraktive Klangersynthese.



Abbildung 23 Einstellungen des Software Synthesizers

In Abbildung 23 sind die Einstellungen von Massive X zu sehen. Im Grunde handelt es sich bei diesem Sound um eine einfache Kombination aus zwei Wellenformen. Zum einen einer Mischung aus einer Dreiecks- und Rechteckswelle, zum anderen aus einem einfachen Sinus. Diese sind auf Abbildung 23 an den großen Piktogrammen, welche die konkrete Form der Wellenformen optisch wiedergeben, gut zu erkennen. Beide Wellen werden zu fast gleichen Anteilen zueinander gemischt, zu sehen an den Schiebereglern, die mit einer roten Zahl „2“ und „4“ gekennzeichnet sind. Neben der Einstellung der Wellenformen wurde zusätzlich die Tonhöhe der ersten Welle um 12 Halbtöne abgesenkt. Ziel war es, einem Verbrennungsmotor durch eine tief klingende, rauere Welle, nachzuempfinden. Die Sinuswelle blieb in ihrer Tonhöhe unverändert und soll eine stabile, angenehme Komponente im Klang bilden.

Des Weiteren wurde die Rechteckswelle noch ein wenig gefiltert, wie am Filterregler unter der Wellenform zu sehen ist. Dadurch werden besonders hohe Frequenzen gefiltert, sodass der Klang nicht zu spitz und so auf Dauer nicht unangenehm wird. Bei der Sinuswelle ist der Filterregler vollständig aufgedreht, dies entspricht keiner Filterung.

Alle weiteren Parameter in Massive X blieben unberührt.

So wurde mit diesem Klang ein C-Dur Akkord gespielt, wobei der Grundton auf dem C der Klaviatur liegt und das c noch als Oktave dazu genommen wird. Anschließend wurden die Töne des Akkords einzeln aus der Digital Audio Workstation (DAW) herausgerechnet und in FMOD importiert.

In FMOD wurden die einzelnen Töne als Audiospuren angelegt, das G wurde dabei noch einmal dupliziert, um den Ton auf zwei unterschiedliche Arten bearbeiten zu können. Mit diesem Vorgehen können die einzelnen Töne unabhängig voneinander weiterbearbeitet werden.

Alle Töne, ausgenommen dem c, wurden mit einem Pitch Shift Effekt belegt. Dieser verschiebt die Tonhöhe in Abhängigkeit von der Geschwindigkeit. Je schneller das Fahrzeug fährt, desto höher sind jeweils die abgespielten Töne.

Auch die Lautstärke der einzelnen Töne verändert sich je nach Geschwindigkeit. So ist im Stand kaum etwas zu hören, beim Anfahren werden die tieferen Töne zunehmend lauter, mit steigender Geschwindigkeit werden schließlich die höheren Töne lauter, wohingegen die tieferen Töne langsam wieder leiser werden. Zwischen 30 km/h und 33 km/h wird das Geräusch vollständig ausgeblendet, weil das Abrollgeräusch des Rollers zur Wahrnehmung ausreicht.

Der dritte von vier Kernparametern ist ein Multiband Equalizer. Dieser funktioniert hier im Grunde als Tiefpassfilter für drei der Töne, welcher sich mit zunehmender Geschwindigkeit weiter öffnet und somit höhere Frequenzen hörbar werden. Das Öffnen des Filters suggeriert außerdem einen Beschleunigungsvorgang, da ein konventioneller Motor ebenfalls mit zunehmender Drehzahl höher frequente Anteile im Spektrum aufweist. Um noch mehr dem Gefühl eines Verbrennungsmotors nachzukommen, liegt - mit Ausnahme des hohen c's - ein Tremolo Effekt auf der Spur. Dieser Effekt sorgt dafür, dass mit einer bestimmten Frequenz die Lautstärke des Signals oszilliert. Auf diese Weise bekommt der Klang eine rhythmische, perkussive Komponente, welche wiederum an die Bewegung der Kolben eines Verbrennungsmotors erinnert. In Anhang C.1 sind die Automationskurven der Effekte und der Lautstärke, in Abhängigkeit von der Geschwindigkeit zwischen 0 km/h und 40 km/h abgebildet. „G_dup“ ist markiert, da der duplizierte Ton nicht von der Geschwindigkeit abhängig ist.

Er ist ausschließlich von der Gaspedalstellung abhängig. Dadurch entsteht bei stärkerem Gas geben ein zusätzliches Geräusch. Die Gaspedalstellung wird prozentual zwischen 0 und 100 angegeben, bei etwa zwei Drittel erreicht der Pitch Shifter das Maximum und gibt somit die Oktave wieder. Genauso auch der Tremolo Effekt, welcher hier die Maximale Frequenz von 20Hz erreicht, was so viel bedeutet, dass das Signal innerhalb einer Sekunde 20-mal zwischen eingestellter Lautstärke und kompletter Stille oszilliert. Durch diesen Ton, der lediglich bei starker Beschleunigung auftritt, soll der Fahrer auch dementsprechendes Feedback auf der auditiven Ebene bekommen. Des Weiteren ist auch der tiefe Grundton des Klanges von der Beschleunigung abhängig. Er wird kontinuierlich lauter bzw. höher gestimmt je höher die Gaspedalstellung ist.

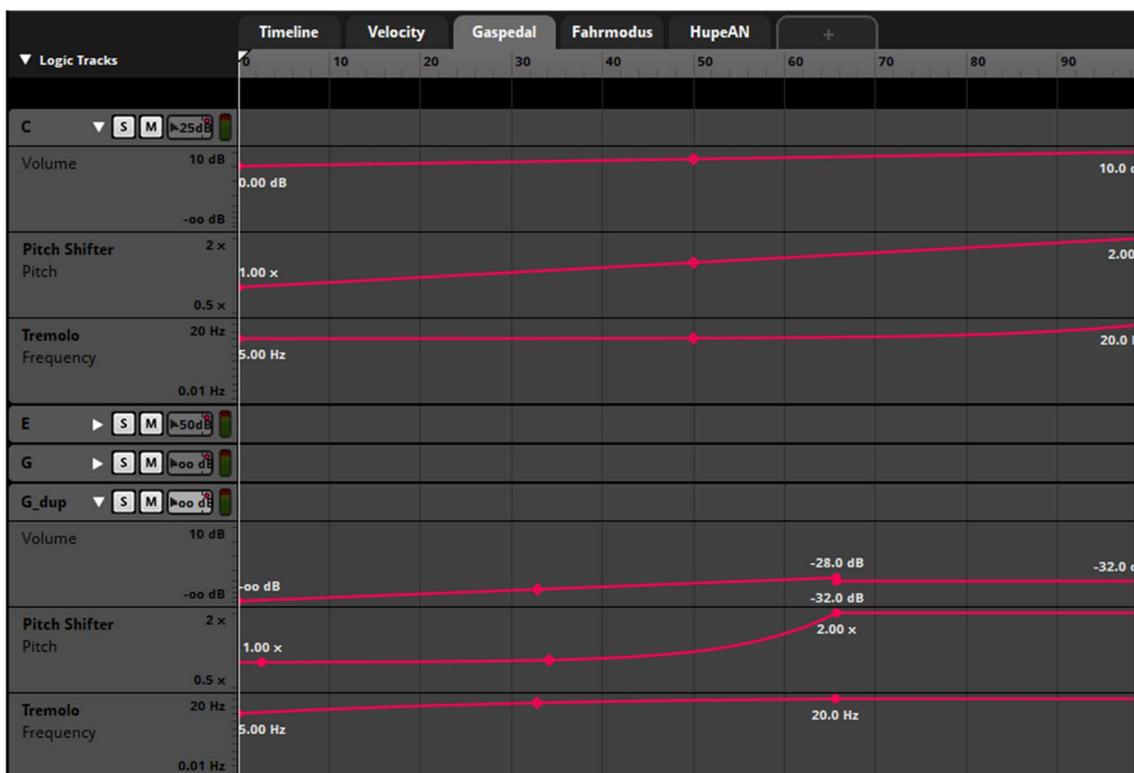


Abbildung 24 Automationskurven der Gaspedalstellung – Soundpattern 1

In Abbildung 24 sind die Automationskurven der Gaspedalstellung in FMOD zu sehen. Es ist zu erkennen, dass das duplizierte G lediglich maximal -28dB erreicht, es wird also eher subtil dazu gemischt und erst bei hoher Beschleunigung wahrnehmbar. Aus der Mischung dieser Parameter war es das Ziel mit dem ersten Soundpattern, einem Verbrennungsmotor auf synthetische Art und Weise nachzuempfinden.

Soundpattern 2

Das zweite Soundpattern soll im Gegensatz zum ersten etwas futuristischer und neuartiger klingen. Auch hier ist für die Erstellung des Grundklanges der Massive X Software-synthesizer zum Einsatz gekommen. In der folgenden Abbildung sind die Einstellungen von Massive X zu sehen.



Abbildung 25 Oszillator des zweiten Soundpatterns

Auf Abbildung 25 ist grob die Wellenform zu erkennen, welche zunächst nicht viel Aufschluss über den Klang gibt. Es handelt sich dabei um eine auswählbare Voreinstellung von Massive X, welche an ein synthetisiertes Telefonklingeln erinnert. Ansonsten wurden keine weiteren Einstellungen an Massive X vorgenommen, um in FMOD ein möglichst rohes Ausgangssignal zu erhalten.

Auch hier wurde dann ein C-Dur Vierklang gespielt, die einzelnen Töne exportiert und in FMOD auf einzelne Spuren übertragen. Ähnlich wie beim ersten Soundpattern wurden die einzelnen Töne mit Effekten in Abhängigkeit von Geschwindigkeit und Gaspedalstellung bearbeitet. Dabei kommen sogar die gleichen Effekte zum Einsatz. Lediglich die Automationskurven unterscheiden sich.

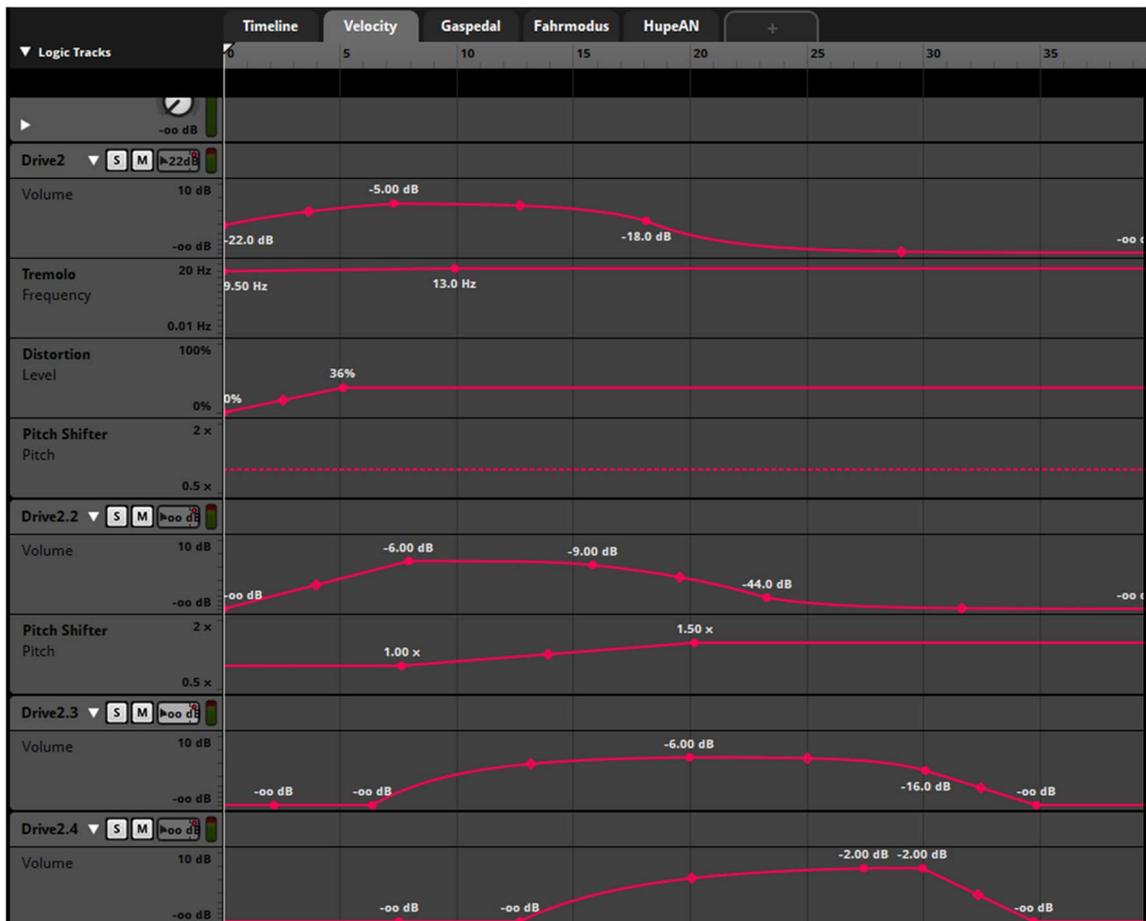


Abbildung 26 Automationskurven der Geschwindigkeit – Soundpattern 2

Auch beim zweiten Soundpattern ist die oberste Spur „Drive2“ der tiefste Ton und „Drive2.4“ der Höchste. An den Volume Kurven ist also auch hier zu erkennen, dass im niedrigen Geschwindigkeitsbereich eher die tieferen Töne aktiver sind und je schneller das Fahrzeug fährt desto mehr werden die höheren Töne eingeblendet. Ab 30 km/h werden die Geräusche vollständig ausgeblendet.

Der tiefste Ton wird außerdem mit einem Tremolo Effekt versehen um wieder eine rhythmische Komponente hinzuzufügen. Genauso wird er ab 5 km/h etwas verzerrt, um ihn etwas präsenter zu machen bis er zwischen 15 km/h und 20 km/h ausgeblendet wird.

Ansonsten wird lediglich „Drive2.2“, also das e, in der Tonhöhe verändert, sodass der Übergang zum dritten Ton etwas weicher ist und das Gefühl eines musikalischen Slides entsteht.

Bei der Automation der Gaspedalstellung wird ausschließlich der tiefste Ton bearbeitet.

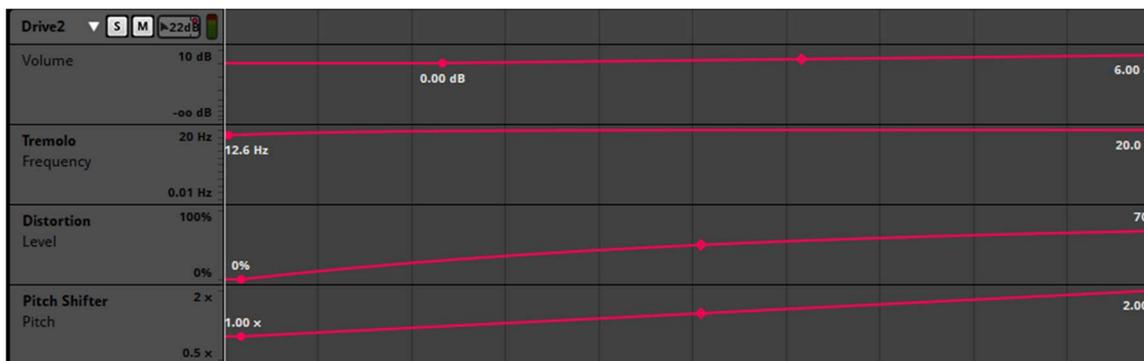


Abbildung 27 Automationskurven der Gaspedalstellung – Soundpattern 2

Auf Abbildung 27 ist zu erkennen wie sich die Lautstärke, das Tremolo, die Verzerrung und die Tonhöhe in Abhängigkeit von der Gaspedalstellung verändern. Bei hohen Geschwindigkeiten stehen die Automationskurven der Geschwindigkeit hierarchisch über den Kurven der Gaspedalstellung. Das bedeutet, wenn das Fahrzeug etwa 25 km/h fährt, dann verstummt der tiefste Ton beinahe und auch die Kurven der Gaspedalstellung haben demnach keine direkte Auswirkung mehr auf den Klang von „Drive2“. Auf diese Weise soll das Geräusch bei sehr schneller Beschleunigung etwas rauer werden und dem Fahrer so Feedback geben.

Der höchste Ton des Soundpatterns ist mit einem permanenten Tremolo Effekt versehen, welche unabhängig von den CAN-Daten ist. So entsteht zwischen 20 und 30 km/h ein etwas perkussiveres, höher frequentes Geräusch, da besonders in diesem Bereich die Ortung und Erkennung zu gewährleisten ist.

Soundpattern 3

Das dritte Soundpattern soll in dieser Reihe den experimentellsten Klang darstellen. Hier wurde gezielt ein Ansatz gewählt, welcher eher nach einem realen Instrument klingt.

Er basiert auf einem atmosphärischen Pad Sound, welcher ebenfalls in Massive X erzeugt wurde. Er erinnert etwas an den Klang einer Orgel.

Auf der folgenden Abbildung ist zu erkennen, wie die Wellenform näherungsweise aussieht, außerdem auch teilweise die Einstellungen zur erzeugten Wellenform.



Abbildung 28 Oszillator des Grundgeräuschs für das zweite Soundpattern

Über der Wellenform ist der Name jener, welche in Massive als Ausgangswellenform wählbar ist. Hierbei handelt es sich um den sogenannten „Chrome“, welcher entfernt an eine Sinuswelle erinnert. Der rote Ring hat in diesem Fall keine Bedeutung, er zeigt die Veränderung der Wellenform in Abhängigkeit von einem zugewiesenen Controllerwert an. Diese Funktion wurde ausschließlich zum Testen des Sounds genutzt und über einen Encoder gesteuert.

Der blaue Ring hingegen zeigt an wie sich die Wellenform in Abhängigkeit von einem voreingestellten Envelope Filter ändert. Dieser ist in der folgenden Abbildung dargestellt.

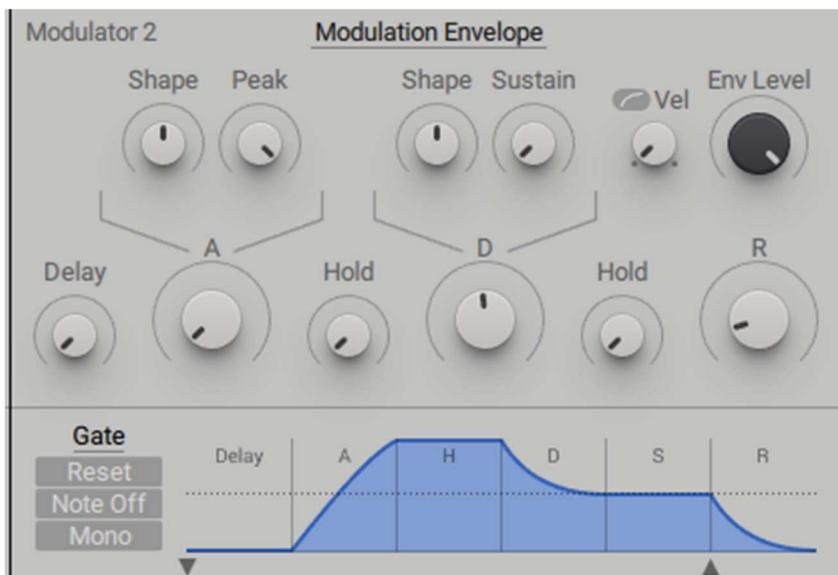


Abbildung 29 Envelope Filter E1

Er funktioniert im Gate Modus, das bedeutet, dass der Filter auf einen Trigger reagiert und im Anschluss der Filter angewendet wird. Das „Gate“ öffnet sich also beim Betätigen einer Taste des MIDI Controllers und somit beim Spielen eines Tons. Die Filterkurve spiegelt somit den Verlauf, der Veränderung, der Wellenform des Signals wider. Zur besseren Veranschaulichung ist in der folgenden Abbildung zu sehen, wie die Wellenform beim Maximum der Filterkurve aussieht.



Abbildung 30 Wellenform bei Envelope Maximum

Das Signal wird etwas rechteckiger und die Frequenz erhöht sich ebenfalls. Es wurde erneut ein C-Dur Dreiklang gespielt und jeder Ton einzeln in FMOD importiert.

Auf der folgenden Abbildung sind die Automationskurven in Abhängigkeit von der Geschwindigkeit zu sehen.



Abbildung 31 Automationskurven der Geschwindigkeit – Soundpattern 3

Wenn man die Volume-Kurven der drei Töne betrachtet, erkennt man im Unterschied zu den anderen beiden Sounds, dass die Töne erst ab 30km/h ausgeblendet werden. Mit zunehmender Geschwindigkeit entsteht also, angefangen mit dem tiefsten Ton, ein vollständiger C-Dur Akkord, da die Tonhöhe unabhängig von der Geschwindigkeit ist und sich nicht verändert. Lediglich der höchste Ton wurde mit weiteren geschwin-

digkeitsabhängigen Effekten bearbeitet. Zum einen ein Multiband EQ, welcher als Tiefpassfilter mit einer Resonanzfrequenzanhebung an der Flanke bei steigender Geschwindigkeit geöffnet wird. Des Weiteren wurde hier ein Tremolo-Effekt verwendet, um wieder bei steigender Geschwindigkeit eine pulsierende Komponente hinzuzufügen. Hier wird diesmal nicht nur die Tremolo-Frequenz, sondern auch die „Depth“, also die Dynamik des Tremolo-Effekts mithilfe der Automationskurve, erhöht.

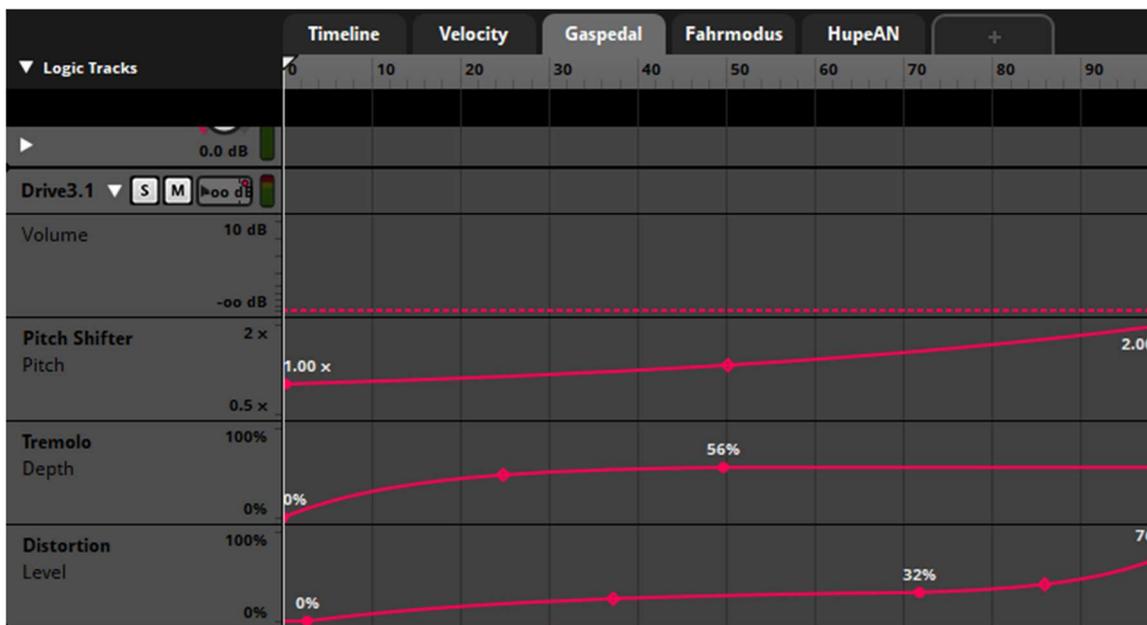


Abbildung 32 Automationskurven der Gaspedalstellung – Soundpattern 3

Bei diesem Sound wird lediglich der Grundton in Abhängigkeit von der Gaspedalstellung verändert. Die Tonhöhe wird zum einen parabelförmig, langsam angehoben. Zum anderen kommt hier ein leichter Tremolo-Effekt zum Einsatz.

Neben den parameterabhängigen Effekten liegt auf dem höchsten Ton des Akkords noch ein Hall, da dieser auch häufig in der Musik, insbesondere bei Orgel ähnlichen Klängen, Verwendung findet.

Da FMOD das Signal als Summe an den Ausgang schickt, wurde noch eine Summenbearbeitung, die für alle Soundpatterns gültig ist, durchgeführt. Dabei handelt es sich um einen Kompressor und einen Limiter. Mithilfe des Kompressors konnten die Signale etwas in der Dynamik eingegrenzt werden, sodass das Signal konsistenter in seiner Lautstärke ist. Außerdem wurde der Gesamtpegel um 7dB angehoben, da aus der DAW heraus mit etwas zu viel Headroom gearbeitet wurde.

Der Limiter war dafür zuständig, auftretende Spitzen abzufangen, um eine potenzielle Übersteuerung zu verhindern.

Auf Abbildung 30 sind die Einstellungen der Dynamikbearbeitungen zu sehen.



Abbildung 33 Summenbearbeitung in FMOD

„LINK“ bedeutet hier, dass das Signal in Stereo bearbeitet wird. Dies ist notwendig, da einige der Soundeffekte in Stereo anliegen, obwohl die Kanäle auf einen Lautsprecher geschickt werden und die zwei Kanäle dort gemischt werden.

6.3 „Friendly Horn“

Die Geräusche der freundlichen Hupe wechseln mit dem Fahrmodus, um so eine größere Bandbreite von verschiedenen Hupen testen zu können. Die Hupen sind klanglich nicht mit dem jeweiligen Fahrgeräusch abgestimmt worden und sind daher in der klanglichen Bewertung unabhängig von den Fahrgeräuschen zu beurteilen.

In FMOD triggern die Hupen immer, wenn die „Set“-Taste auf der Lenkersteuerung betätigt wird. Dies geschieht, indem der Parameter Wert auf 1 gesetzt wird und das Abspielen des Tonmaterials nur dann einmalig getriggert wird. Das Geräuschevent ist also direkt abhängig vom Parameter Wert.

Damit nicht alle Hupen der unterschiedlichen Fahr-Modi gleichzeitig getriggert werden, haben die Spuren, auf denen die Hup-Geräusche liegen, eine Lautstärkeautomation in Abhängigkeit vom Fahrmodus. Wenn der erste Fahrmodus ausgewählt ist, sind somit die Hupen der anderen Fahr-Modi stumm geschaltet und ertönen nicht, obwohl sie genau genommen auch getriggert werden.

Die zum ersten Fahrmodus korrespondierende Hupe wurde mithilfe von „Kontakt“, einem samplebasiert arbeitenden Softwareinstrument, erstellt. Innerhalb von „Kontakt“

stehen unterschiedlichste Instrumente in virtueller Form zur Verfügung. Für die erste „freundliche Hupe“ kam ein Kalimba zum Einsatz.

Auf der folgenden Abbildung ist die Softwareoberfläche des Softwareinstruments in „Kontakt“ zu sehen. Die Einstellungen wurden zum größten Teil belassen, da standardmäßig ein angenehm klingender Kalimba-Klang beim Laden des Instruments vorhanden ist.



Abbildung 34 Softwareoberfläche von Kontakt mit geladenem Kalimba

Es erklingt also ein sattes Kalimba mit etwas Hall. Eingespielt wurde lediglich ein C und dessen Oktave, da dieser Ton eine allgemein hohe Akzeptanz hat. Besonders in der westlichen Kultur wird dieser Ton als vertraut und freundlich wahrgenommen.

Die Oktave wurde dann zweimal in Folge eingespielt. Die erste Oktave ist etwas kürzer und staccato gespielt, die Zweite legato und klingt demnach aus.

Wie bei den Fahrgeräuschen wurde dann das Tonmaterial aus der DAW exportiert und in das FMOD-Projekt eingefügt.

Die zweite Hupe basiert auf einem reinen Sample aus einer Sound-Bibliothek.³⁷ In dem Sample werden Motoren- und Hupengeräusche vokal nachgestellt. So wurde eine Hupe aus dem Signal herausgeschnitten und um 3 Halbtöne nach oben transponiert um auch hier in etwa ein C als Grundfrequenz zu haben.

³⁷ <https://freesound.org/people/evan.schad/sounds/462677/>

Des Weiteren wurde noch ein leichter Equalizer auf das Signal angewendet.



Abbildung 35 Equalizer-Einstellungen der zweiten "freundlichen Hupe"

Auf Abbildung 35 ist zu erkennen, dass die Grundfrequenz bei 261Hz liegt, welche fast dem angestrebten C entspricht. Das Sample wurde anschließend so in FMOD importiert und entsprechend auf den zweiten Fahrmodus angelegt.

Die dritte Hupe wurde mithilfe eines polyphonen Synthesizers namens „Prism“ erstellt. Hierbei ist per Zufall ein Preset, welches sich gut eignete, zum Einsatz gekommen. Klanglich ähnelt es einem perkussiven glockenartigen Klang, allerdings etwas kleiner und trotz guter Annäherung, synthetisiert und künstlich.

Durch diese Mischung fiel die Entscheidung für die letzte Hupe auf diesen vorgefertigten Klang. Wie bei der ersten Hupe wurden zwei aufeinanderfolgende Töne gespielt, der erste kurz als staccato und der zweite ausklingend, legato. Für eine bessere klangliche Vergleichbarkeit der Hup-Geräusche wurde auch hier wieder ein C gespielt. Diesmal allerdings ohne eine zusätzliche Oktave, da der Klang an sich schon eine gute Durchsetzungsfähigkeit hat.

Auch die Sounds für die „freundliche Hupe“ sind von der in Kapitel 6.2 am Schluss erwähnten Summenbearbeitung betroffen.

Es fanden zu den Fahrgeräuschen keine Messungen statt, um die gesetzliche Konformität der EU-Vorgaben zu prüfen. Dies hatte vielerlei Gründe, zum einen stand keine adäquate Testumgebung zur Verfügung. Das bedeutet die Messergebnisse wären nur eine grobe Annäherung und müssten im Kontext der verfügbaren Teststrecke bewertet werden. Zum anderen ist dieses AVAS lediglich ein Prototyp und man müsste zumindest den Teil der aktiven Geräuschwiedergabe, also den Lautsprecher, fest in das Fahrzeug verbauen. Erst wenn sich die Audiohardware nicht mehr ändert und fest im Fahrzeug installiert ist, lohnt es sich eine Messung nach den Richtlinien der EU durchzuführen. Daher sind die erstellten Sounds für diese Arbeit zunächst als Annäherung zu verstehen.

Alle erstellten Fahrgeräusche mit der jeweiligen Hupe befinden sich in einem Screenshot als Videodatei in Anhang A.4. Des Weiteren sind auch die Rohdaten der Audiodateien, bevor sie in FMOD importiert wurden, in Anhang A.5 zu finden.

Aus betrieblichen Gründen war es leider nicht möglich das Fahrzeug mit ausgestattetem AVAS-Prototypen aufzuzeichnen.

6.4 Portierung auf ein eingebettetes System

Zum Abschluss dieses Kapitels folgt ein kleiner Ausblick auf die Portierung des AVAS auf ein eingebettetes System und die entsprechenden Anforderungen dafür. Dazu ist es zunächst wichtig zu erfahren, wie viel Leistung die Software in etwa braucht. FMOD bietet die Funktion ein Leistungsprofil zu erstellen. Dort werden verschiedenste Parameter während des Betriebs der Software aufgezeichnet. Dabei sind vor allem Größen wie die Berechnungsdauer des Signals oder Arbeitsspeicher ausschlaggebend, da diese Größen in der Regel am meisten Einfluss auf die Wirtschaftlichkeit des Systems haben. Die anschließende Analyse der Leistung des Systems basiert auf Anhang C.2, hier wurde in jedem Fahrmodus mindestens einmal auf etwa 30 km/h beschleunigt und anschließend ausgerollt.

In dem sogenannten FMOD Profiler gibt es eine Größe, die sich „CPU – Mixer“ nennt. Sie gibt in Prozent an, wie viel Zeit der Mixer-Thread benötigt. Sie repräsentiert also das Verhältnis der benötigten Zeit für einen Mix zur verfügbaren Zeit. Folglich bedeutet das, je niedriger die prozentuale Auslastung, desto stabiler und sicherer kann ein konstanter Audiostrom gewährleistet werden. Bei dem verwendeten AVAS liegt die CPU-Mixer Nutzung in ihren Spitzen bei nur etwa 9%. Sie ist davon abhängig, wie viele Spuren gleichzeitig aktiv sind. Wenn das Fahrzeug beginnt schneller zu fahren, steigt auch der Rechenaufwand etwas, übersteigt allerdings nie 9%. So lässt sich sicher sagen, dass der Prototyp überdimensioniert ist und die Berechnungen ohne Probleme durchführen kann. Im Umkehrschluss heißt das für die Portierung auf ein eingebettetes System, dass hinsichtlich dieses Parameters sogar etwa ein Zehntel der Prozessorleistung für dieses Programm ausreichend wären.

Neben der „CPU - Mixer“ Größe gibt es noch die „CPU - Update“ Größe im Profiler von FMOD. Sie gibt die prozentuale Zeit an, die benötigt wird, um den FMOD Update-Thread durchzuführen. Dieser wird üblicherweise alle 20ms durchgeführt. Die Angabe zeigt also auch hier an, wie viel Prozent der verfügbaren Zeit dafür genutzt werden. Dieser Wert bewegt sich konstant im Bereich von maximal etwa 4% und minimal 3,1%. Das heißt, auch hier könnte bei einer Portierung der Prozessor wesentlich kleiner dimensioniert werden.³⁸

Die nächste Kenngröße ist der Arbeitsspeicher. Er gibt an, wie viel Systempeicher während des Ausführens des Programms benötigt wird. Dieser Wert liegt bei dem Test konstant bei knapp 17,5 KB. Das ist angesichts der verfügbaren 4GB Arbeitsspeicher sehr wenig. Folglich könnte man an dieser Stelle wesentlich reduzieren und komplett auf einen zusätzlichen Arbeitsspeicher verzichten. Dann könnte beispielsweise ein einfacher Mikrocontroller die Steuerung der zu ladenden Audiodateien übernehmen.

Zuletzt spielt der reine Massenspeicher noch eine Rolle. Diesen kann man allerdings beinahe vernachlässigen, da FMOD die Audiodaten als Ogg Vorbis komprimiert und man sogar den Grad der Kompression einstellen kann. Ogg Vorbis ist grob mit den bekannteren Datenformaten AAC oder mp3 zu vergleichen. Es bietet also bei guter Kompressionsrate nach wie vor gute Audioqualität. Die Audiodateien sind demnach beim

³⁸ <https://www.fmod.com/resources/documentation-studio?version=2.0&page=profiling.html>

aktuellen AVAS zwischen maximal 200 kB und mindestens 30 kB groß. Das entspricht bei einem einzelnen Soundpattern nicht einmal einem Megabyte.

Zusammenfassend kann man also sagen, dass die Anforderungen an das System von der technischen Seite her nicht allzu hoch sind und sich sicherlich ein kostengünstiges, eingebettetes System zusammenstellen ließe – Idealerweise direkt in Verbindung mit einem Audioverstärker, der genug Leistung für die erforderlichen Schallpegel zur Verfügung stellt.

Softwareseitig ist es wichtig, dass die CAN-Bibliothek in das System eingebunden werden kann. Auch das Programm, das hier verwendet wird, müsste neu geschrieben werden, da es auf Windows-Systeme ausgerichtet ist. Daher müsste das Anpassen der Software von der verwendeten Hardware abhängig gemacht werden. Grundsätzlich lässt sich der reine Code gut auf andere Programmiersprachen übertragen. Jedoch müsste ein Workaround für die dynamische Gestaltung der Geräusche ohne FMOD oder vergleichbarer Werkzeuge, gefunden werden.

7 Nutzerstudie

Dieses Kapitel basiert vor allem auf einem persönlichen Gespräch mit den Kollegen von Bosch in Vaihingen, sie setzen sich mit den Themen Projektstrukturierung für User Research und User Experience auseinander. Da das Gespräch leider erst gegen Ende der Praxisphase dieser Arbeit stattfinden konnte, sind einige Konzepte der Methodik nicht mit in den Designprozess der praktischen Arbeit eingeflossen. Dennoch wird das entsprechende Vorgehen in diesem Kapitel theoretisch aufgegriffen und aufbereitet.

Zu Beginn des Designprozesses steht die sogenannte „Question 0“ (Q0), sie hat zum Ziel, dass das Produktziel mit einer einzigen Frage so gut wie möglich eingegrenzt wird. Sie hilft dabei, den Prozess zu strukturieren und den Umfang stets zu definieren. In der Regel werden sechs Unterfragen in die spätere Q0 einfließen, die hier beispielhaft anhand des AVAS dargelegt werden:

Was ist das Produkt?

Eine aktive Soundunterstützung für geräuscharme Fahrzeuge, mit dem Fokus auf E-Scooter.

Wofür ist das Produkt?

Für andere Verkehrsteilnehmer zur Verbesserung der Akzeptanz und der Wahrnehmung im Straßenverkehr. Der Fahrer profitiert so auch durch Sekundäreffekte. Flottenbetreiber und OEMs (Original Equipment Manufacturer) können außerdem durch Soundbranding eine verbesserte Erkennbarkeit erzeugen.

In welchem Kontext und wo?

Urbane Kurzstreckenmobilität durch private Nutzer und „Last-Mile-Logistik“, innerhalb der europäischen Verkehrskultur.

Für wen, wer ist der Nutzer?

Für potenzielle E-Scooter Fahrer.

Wie ist der Status Quo?

Derzeit gibt es noch keine AVAS für Zweiräder.

Welche Technologie wird verwendet?

Technische Plattform sind im Wesentlichen der LattePanda und der Lautsprecher.

Dabei müssen für die Q0 mindestens zwei dieser Fragen verarbeitet werden. Die Formulierung der Q0 hat einen deterministischen Einfluss auf den Lösungsfreiraum und daher auch auf das Innovationspotenzial des Projektes.

Zur Formulierung der Q0 kann es eine gute Hilfestellung sein, sich zu fragen was das „bestmögliche“ Ergebnis ist, welches das Produkt erzielen kann. Mithilfe der Q0 ist das Produktziel immer klar definiert und bietet trotzdem Raum für verschiedenste Lösungsansätze. Daher sollte man sich die Q0 im Laufe des Entwicklungsprozesses immer wieder stellen.

Bei der Entwicklung des AVAS bzw. auch eines Use Cases wie dem „friendly horn“ wäre beispielsweise ein Lösungsansatz wie bei einem Audiologo denkbar. Es ist sinnvoll sich zunächst die Frage zu stellen was die Werte der Marke sind und wie sie sich auf die Hörebene übertragen lassen. Dabei ist es wichtig, dass eine Unmissverständlichkeit entsteht, die Marke muss klar kommuniziert werden können.

7.1 Vorgehen und Methodik

Durch die dargelegte Projektstrukturierung lassen sich je nach Stadium verschiedene User Tests ableiten. Sie sind immer abhängig vom Projektfortschritt und können dynamisch angepasst werden, sodass man in der entsprechenden Situation das optimale Kosten-Nutzen Verhältnis des User Tests erhält.

Je konkreter das Produkt wird und je weiter der Designprozess fortgeschritten ist, desto hilfreicher sind quantitative Verprobungen. Das setzt ein nahezu auslieferungsfähiges Produkt voraus, was gleichzeitig auch bedeutet, dass das Designteam ihre Ideen ausgeschöpft hat und bereits mehrere Iterationszyklen des Produkts durchlaufen hat.

Die Tests beinhalten in der Regel konkrete Fragen nach dem Empfinden, unter Umständen sogar Multiple Choice Fragen mit vorgegebenen Antwortmöglichkeiten. Auf Grundlage dieser messbaren Werte aus der Befragung können fundierte Projektentscheidungen gegen Ende des Prozesses getroffen werden.

Die qualitative Befragung eignet sich hingegen auf dem Weg zum Ziel besser, wenn sich das Designteam in Entscheidungen selbst noch unsicher ist und lediglich ein Gefühl für die Richtung braucht, in die weitergearbeitet werden soll. Dabei spielt es eine untergeordnete Rolle, ob besonders viele Personen befragt werden. Es ist, je nach Zielgruppe des Produkts, wichtiger, dass die Befragten eine ausgeglichene Altersstruktur aufweisen. So würde man beispielsweise bei der Planung eines User Tests für das AVAS berücksichtigen, dass die allgemeine Altersverteilung bei Rollerfahrern bzw. e-Rollerfahrern ausgeglichen ist.

Bei der qualitativen Befragung wird näher auf die Personen eingegangen. Das Ergebnis der Umfrage basiert auf dem Gespräch mit den Probanden, dabei ist es wichtig, auch die Emotion und Stimmung in das Ergebnis mit aufzunehmen. Hierbei ist je Proband wesentlich mehr Zeit einzuplanen als bei einer quantitativen Verprobung. Gleichzeitig sind die Ergebnisse der einzelnen Probanden im frühen Projektstadium wesentlich aufschlussreicher und zielführender. Sie sind außerdem auch einfacher durchzuführen, da nicht viele Personen benötigt werden.

Häufig lässt sich in einem frühen Stadium des Produkts eine aufgemachte Testsituation schaffen, die nicht das fertige Produkt benötigt um durchgeführt werden zu können. So lassen sich Zeit und Kosten einsparen, ohne Einbußen auf die Repräsentierbarkeit des Ergebnisses in Kauf nehmen zu müssen. Wenn es beispielsweise nur darum ginge eine Klangprobe zu beurteilen und zu erfragen, was die Probanden dabei empfinden, würde es reichen ihnen Kopfhörer zu geben und ihnen das Signal vorzuspielen. Diese Information kann zunächst auch unabhängig vom Verkehr und sogar vom Fahrzeug erhoben werden.

Aufgrund des Projektstadiums des AVAS dieser Arbeit kommt zusammengefasst daher eine qualitative Verprobung mit einer möglichst breitgefächerten Altersstruktur in Frage. Dabei ist es wichtig, die Situation während des aktiven Fahrens zu beurteilen. Denn dabei kann bereits auf Merkmale wie Ortbarkeit, Richtung aber auch auf Aspekte wie den Doppler Effekt, Rücksicht genommen werden. Auf diese Weise können der

Klang und die Art der Fahrgeräusche effektiv eingegrenzt werden und man iteriert langsam in Richtung eines Ergebnisses, das eine gute allgemeine Akzeptanz erzeugt. Eine quantitative Befragung würde sich erst richtig lohnen, wenn das Produkt sich kurz vor dem Endstadium befindet. Dabei wäre es beispielsweise erforderlich, dass der Lautsprecher fest verbaut oder die Hardwareplattform bereits in das System des Fahrzeugs eingebettet ist, da sich so auch die klanglichen Eigenschaften verändern werden und man sich dementsprechend klanglich auch noch daran anpassen müsste. Falls sich der Klang aufgrund der Bauweise und der Hardware nicht mehr verändert und die Designer bereits zufrieden mit dem Klang sind, ist es erforderlich eine breite Masse zu befragen. Dadurch lässt sich gut feststellen, welches Geräusch die schlichte Mehrheit der User besser findet bzw. ob das Ergebnis von anderen Faktoren wie zum Beispiel der Altersgruppe abhängig ist.

Rückblickend wäre es für das Sounddesign sicherlich sinnvoller und effizienter gewesen, sich an einer Question 0 zu orientieren und den Projektverlauf danach auszurichten. Die Soundvorstellung hätte sich vermutlich früher konkretisiert und wäre von Anfang an etwas strukturierter gewesen als die angewandte „Trial & Error“ Methode. Nichtsdestotrotz konnten noch einige Personen zu einem Nutzertest des AVAS gefunden werden, sodass ein kleiner qualitativer Test möglich war. Dadurch konnten noch fundierte Änderungen am Klang des AVAS vorgenommen werden, welche zu einem subjektiv besseren Ergebnis geführt haben.

7.2 Ergebnisse

Letztlich wurden 15 Personen einzeln zu dem AVAS-Stand befragt, welcher sich in Anhang A.4 befindet. Konkret hat der Nutzertest mit einer Probefahrt begonnen. 12 der Probanden waren vor dem Versuch bereits mit E-Mobilität und dem Testträger vertraut, sodass sie sich auf die Geräusche fokussieren konnten.

Nachdem die Probanden selbst gefahren sind, wurden einige Fragen gestellt um so die ersten Eindrücke sammeln zu können und ein unmittelbares Gefühl dafür zu bekommen, wie die Klänge aus Sicht des Fahrers wirken. Im Anschluss wurden die Probanden

in die Situation des Passanten gebracht und er konnte das AVAS als Fußgänger beurteilen. Auch hier wurden direkt im Anschluss die ersten Eindrücke gesammelt. Anschließend wurde je nach Interesse der Probanden noch ein Gespräch über das AVAS geführt.

Zusammengefasst hatten die meisten Probanden nach den Testfahrten allgemein eine sehr positive Einstellung gegenüber dem AVAS und erachteten es als wichtig, dass Fahrzeuge damit ausgestattet werden. Klangliche Präferenzen konnten sich unter den drei verschiedenen Fahrgeräuschen nicht direkt ausmachen. Am ehesten ist Soundpattern 1 aus dem Raster gefallen, es wurde häufig als zu „aggressiv“ und zu tieffrequent empfunden. Dies ist interessant, da sich Soundpattern 1 am ehesten an einem Verbrennungsmotor orientiert hat. Daher deckt sich das etwas schlechtere Abschneiden von Soundpattern 1 auch mit der Umfrage aus Kapitel 3.4, aus der auch deutlich hervorgegangen ist, dass das Imitieren eines Verbrennungsmotors nicht erstrebenswert ist. Jedoch gab es unter den wenigen Probanden auch etwa zwei Personen, die Soundpattern 1 den anderen beiden ausdrücklich vorgezogen hätten. Im weiteren Gespräch stellte sich heraus, dass diese Personen viel Roller- oder Motorrad fahren und den distinktiven Klang des Verbrennungsmotors für ein vollständiges Fahrerlebnis voraussetzen.

Zwischen Soundpattern 2 und 3 war der Unterschied in der Präferenz nicht sehr groß, trotzdem hatte Soundpattern 2 etwas mehr Für-Stimmen bzw. wurde klanglich für etwas besser als Soundpattern 3 befunden. Es war deutlich zu erkennen, dass ein Großteil der Probanden es gut fand, dass die Geräusche sich vom gegenwärtigen Klangbild des Straßenverkehrs stark unterscheiden und dadurch auch gut ortbar waren.

Unter den Probanden befanden sich 4 Musiker, welche alle Soundpattern 3 als am angenehmsten empfanden. Dadurch bestätigte sich auch die Intention dieses Soundpatterns, etwas musikalischer zu sein. Auf der anderen Seite wurde dies gleichzeitig von anderen Probanden als Kritikpunkt geäußert. Sie fanden, dass es zu sehr nach einem Musikinstrument bzw. durch die Anlehnung an eine Orgel zu kirchlich klang. Daher hat das zweite Soundpattern in der Summe die größte Akzeptanz erreicht und wurde von den allerwenigsten Probanden als störend empfunden.

Dies bestärkt ebenfalls die eingangs durchgeführte Umfrage, in der ein neuartiger, freundlicher und nicht zu auffälliger Klang erwünscht war.

In einem nächsten Schritt könnte man nun ausgehend von dem zweiten Soundpattern wieder eine weitere Iteration im Design durchführen und sich so Stück für Stück an einen Sound heranarbeiten, der einem breiteren Testpublikum vorgeführt werden könnte. Dies ist in dieser Arbeit nicht mehr vorgesehen, da es hier in erster Linie darum ging, ein erstes Gefühl zu bekommen und einen prototypischen Testträger aufzubauen. Trotzdem konnten durch die Nutzerstudie in Kombination mit der Umfrage wichtige Informationen zur Erwartungshaltung an ein AVAS gesammelt werden.

Die Idee einer „freundlichen Hupe“ stieß durchweg auf große Akzeptanz. Sehr häufig kam hier der Wunsch der Personalisierbarkeit, ähnlich wie bei einem Handyklingelton. Allgemein wurde es für den Straßenverkehr als sehr sinnvoll erachtet, Personen auf eine zurückhaltende Art und Weise warnen zu können. Die vorhandenen Geräusche fanden bereits Anklang, vor allem das Sample, das aus dem zweiten Soundpattern stammt, wurde gut aufgenommen, da es ein eher lustiger Klang ist und man dadurch im Straßenverkehr auf mehr Akzeptanz stößt.

Aus betrieblichen Gründen war es nicht möglich die Nutzertests aufzuzeichnen.

8 Fazit

Diese Abschlussarbeit bietet einen umfassenden Einblick in die Erwartungshaltung und an die technischen Voraussetzungen von AVA-Systemen. Weiterhin sind sogar schon konkrete Geräusch-Vorstellungen zu entnehmen. Interessant war es hierbei besonders, dass seitens der Umfrageteilnehmer eine große Offenheit gegenüber einer bevorstehenden Neuerung besteht. Denn durch die EU-Vorgaben und die zunehmenden Investitionen in die E-Mobilität wird sich der Straßenverkehr, zumindest langfristig, vom Verbrennungsmotor lösen.

Technisch gesehen sind die meisten Fahrzeughersteller bereits gut gerüstet und staten ihre Fahrzeuge mit AVAS aus. Die Klangwelt dieser Fahrzeuge ist jedoch noch nicht ausgeschöpft und vor allem auch noch nicht optimiert. Im Hinblick auf kleinere Fahrzeuge, wie sie in dieser Arbeit im Mittelpunkt stehen, gibt es beispielsweise noch keinerlei Lösungsansätze. Gleichzeitig wird hier auch von der Politik noch kein Handlungsbedarf gesehen. An dieser Stelle zeigt die Arbeit, die Übertragbarkeit von Geräuschen für größere Fahrzeugklassen auf kleinere Fahrzeugklassen auf. In Bezug auf akustische Warnsysteme für kleinere Fahrzeugklassen im Straßenverkehr dient sie als Grundlage für mögliche Gesetzesanpassungen.

Mithilfe von drei unterschiedlichen Soundpattern konnten in einem Nutzertest Geräusche ausgeschlossen bzw. an ein neutrales, im Straßenverkehr akzeptiertes Geräusch, angenähert werden. Dadurch entsteht ein guter Anhaltspunkt für die zukünftige Entwicklung von AVAS für kleinere Fahrzeugklassen, besonders im Hinblick auf den asiatischen Markt, auf dem vermehrt elektrisch betriebene Roller maßgeblich die Fortbewegung bestimmen. Aber auch „letzte-Meile“-Transporte in westeuropäischen Ländern arbeiten vermehrt mit elektrisch betriebenen Kleinfahrzeugen.

Zuletzt sind die technischen Anforderungen für ein AVAS aufgeschlüsselt worden. Des Weiteren bietet die Arbeit einen detaillierteren Einblick in die Erstellung von Fahrgeräuschen. Dabei sind Kernparameter für die Klanggestaltung herausgestellt worden, welche sich auch auf andere Systeme gut übertragen lassen.

Das Endergebnis ist ein mit einem prototypischen AVAS ausgestatteter Versuchsträger, mit dem es möglich ist ein AVAS zu präsentieren und gleichzeitig die Klangentwicklung fortzuführen.

Anhang A: Datenträger

Aus Gründen der besseren Umweltverträglichkeit sind die Umfrageergebnisse aus A.2 und die Fotos aus A.3 nicht ausgedruckt, sondern befinden sich auf beiliegendem Datenträger.

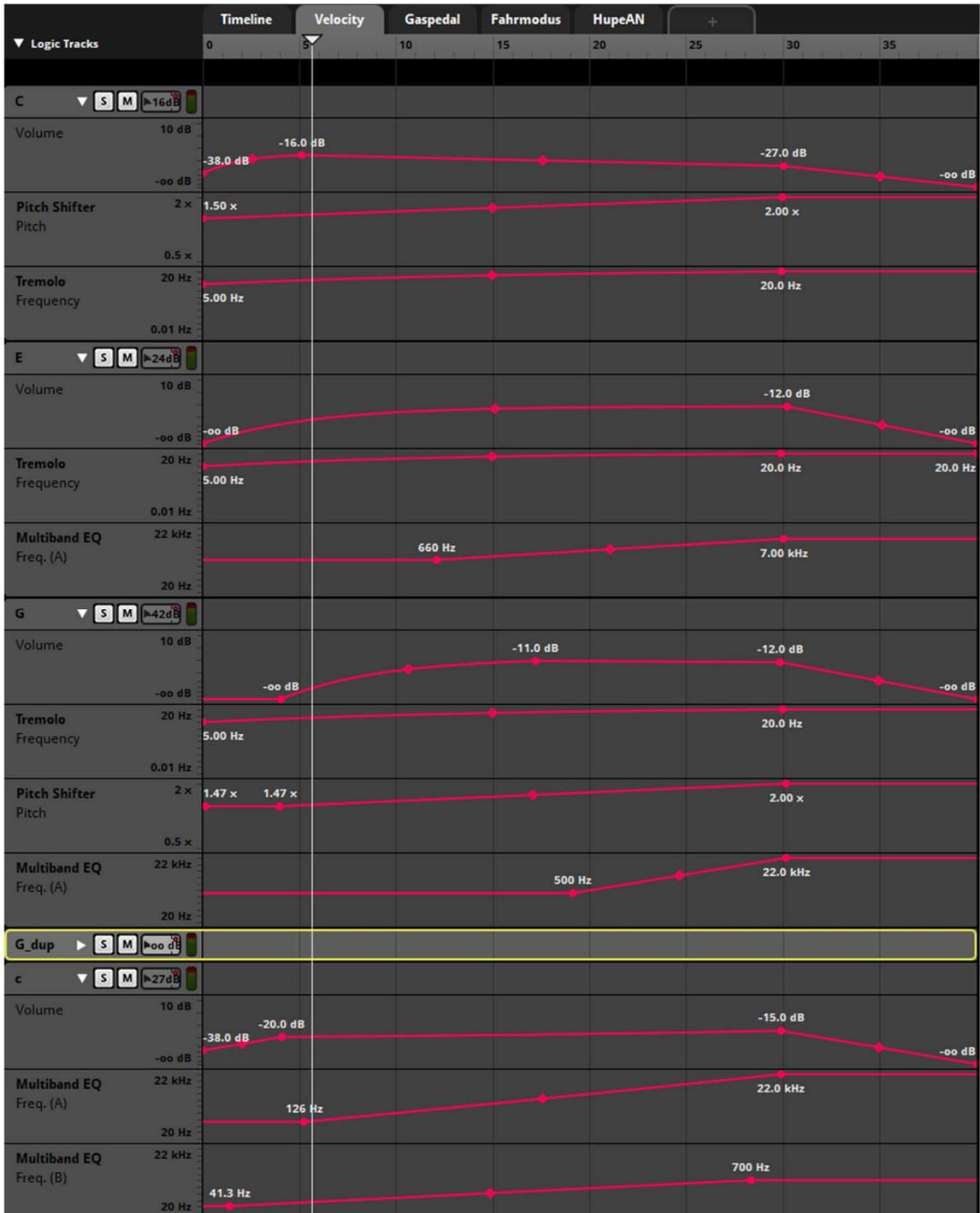
Des Weiteren befindet sich die gesamte Abschlussarbeit als PDF, inklusive FMOD Projektdaten auf dem Datenträger. Das Unity Projekt wurde nicht kopiert, da es ohne die CAN-Hardware nicht starten würde. Jedoch ist der Quellcode ebenfalls angefügt.

[DATENTRÄGER]

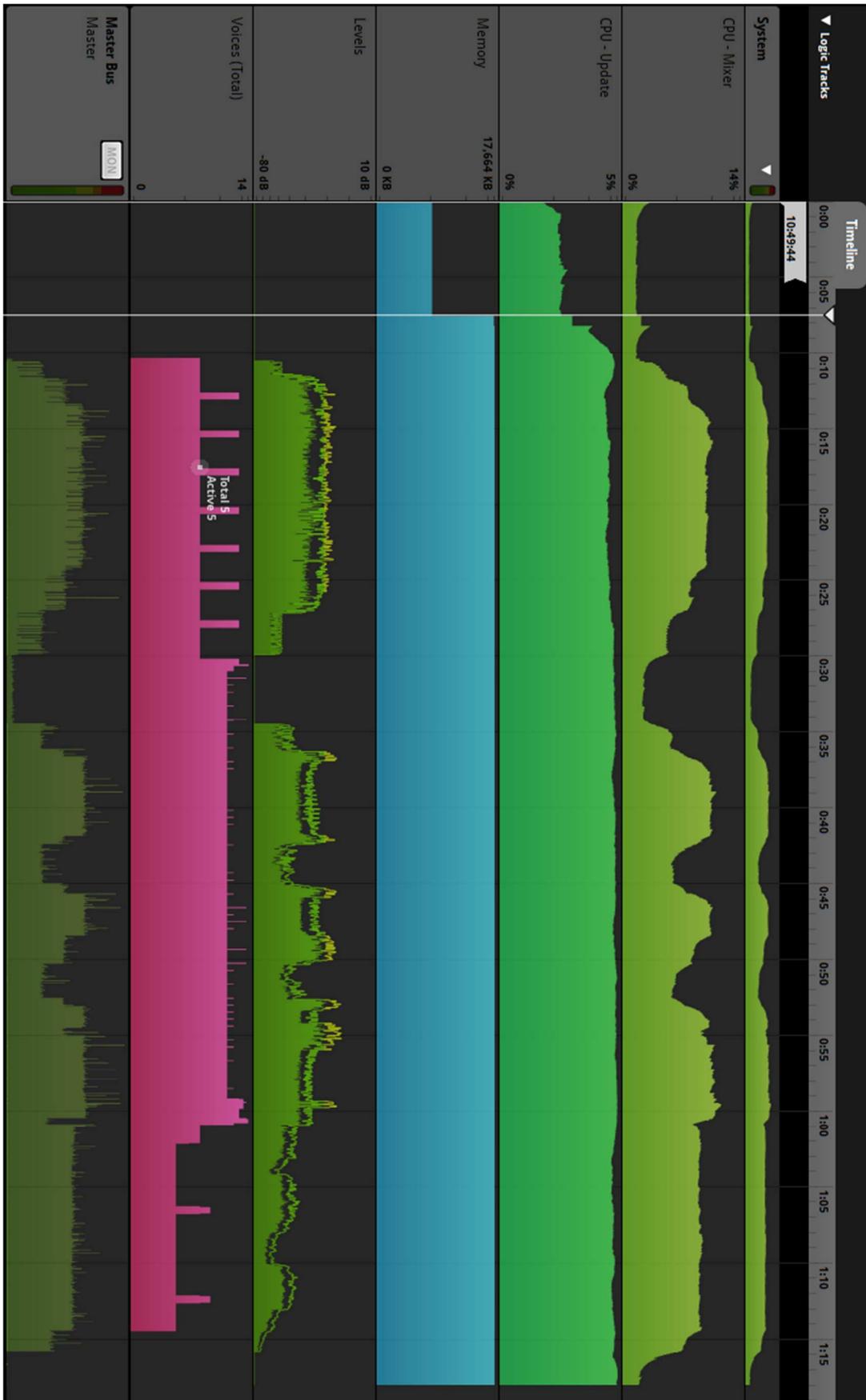
Anhang B: „Proof of Concept – Sheet“

	Low	Medium	High
Hardware	<ul style="list-style-type: none"> - AVAS von Grewus -> Lautsprecher und Verstärkerkombination 	<ul style="list-style-type: none"> - Teensy mit Verstärker (Adapter Board Teensy), SD-Karte und Lautsprecher → Mit Arduino IDE programmierbar → Stromversorgen USB oder 9V 	<ul style="list-style-type: none"> - PC-Stick → Windows 10 → SD-Speicher (bis zu 128GB) → Stromversorgung per microUSB
Software	<ul style="list-style-type: none"> - Bereits vorhandenes Programm optimieren 	<ul style="list-style-type: none"> - Code gibt Sample von der SD-Karte wieder - CAN Messages werden gelesen um Pitch und Lautstärke des wiedergegebenen Samples zu steuern 	<ul style="list-style-type: none"> - Beliebig da normales Win10 Betriebssystem → Unity/FMOD oder Max/MSP
Vorteile	<ul style="list-style-type: none"> - Günstig - Geringer Zeitaufwand 	<ul style="list-style-type: none"> - Vergleichsweise günstig - Kein "Rechner" nötig - Relativ kleiner Formfaktor - Samplebasiert 	<ul style="list-style-type: none"> - Bearbeitungsmöglichkeiten des Klanges - Stereo auch möglich - Leichter Zugriff auf andere Quellen (Hupe, HMI...) - Für verschiedene Plattformen auslieferbar
Nachteile	<ul style="list-style-type: none"> - Wenig Klangliche Modifizierungsmöglichkeiten → Klang nicht optimiert 	<ul style="list-style-type: none"> - Code vermutlich recht kompliziert, da Pitch Operationen nicht trivial - Auch wenig Möglichkeiten den Klang dynamisch zu verändern - Audioqualität verlustbehaftet 	<ul style="list-style-type: none"> - Mit Abstand teuerste Lösung - Leicht „hackbar“ - Evtl. nur manuell bootbar?
Preis	<ul style="list-style-type: none"> - Keine Preise für Grewus Teile verfügbar 	<ul style="list-style-type: none"> - Arduino, je nach Ausführung zw. 20,- – 50,- - Verstärker 4,- – 5,- - SD-Karte 5,- – 10,- - Breitbandlautsprecher 10 – 30,- → 70,- 	<ul style="list-style-type: none"> - Intel PC-Stick 130,- - SD-Karte 5 – 10,- - Adapter 10,- - Outdoor Lautsprecher 6.5" 70,- → 220,-

Anhang C.1: Automationskurven der Geschwindigkeit - Soundpattern 1



Anhang C.2: FMOD Profiler



Quellenverzeichnis

Literaturquellen

Dickreiter, M. Handbuch der Tonstudioteknik, Berlin: De Gruyter, 2014

Raffaseder, H. Audiodesign, München: Carl Hanser Verlag, 2010

Weinzierl, S. Handbuch der Audiotechnik, Berlin: Springer, 2008

Internetquellen (Alle zuletzt am 17. November 2019 abgerufen)

All About Circuits (2019)

<https://www.allaboutcircuits.com/technical-articles/introduction-to-can-controller-area-network/>

Altronics (2019)

<https://altronics.fr/en/guide/can-bus-and-its-benefits/>

Amazon (2019)

<https://www.amazon.de/gp/product/B07DK5JHX3/>

Amazona.de, Lorenz, M. (2018)

<https://www.amazona.de/was-genau-ist-ein-synthesizer-synthesen-im-ueberblick/>

Amtsblatt der Europäischen Union (19. September 2017)

<https://eur-lex.europa.eu/legal-content/DE/TXT/PDF/?uri=CELEX%3A32017R1576&from=DE&fbclid=IwAR3REqOR4HkL77Dmang8-cahdg2UvdSZ-2ufYPIZ9V6YnAznyUY5dtKetAw>

Amtsblatt der Europäischen Union (13. Januar 2017)

<https://eur-lex.europa.eu/legal-content/DE/TXT/PDF/?uri=CELEX:42017X0071&from>

Auto-Motor-Und-Sport, Sommer, M. (24. September 2019)

<https://www.auto-motor-und-sport.de/tech-zukunft/sound-fuer-elektroautos/>

Bonedo Das Musikportal, Eilers, L. (2015)

<https://www.bonedo.de/artikel/einzelsicht/synthesizer-und-sounddesign-3-huellkurven.html>

Delamar – Fachmagazin für Musiker, Baarß, F. (2019)

<https://www.delamar.de/faq/was-ist-ein-oszillator-29487/>

<https://www.delamar.de/faq/syntheseformen-klangsynthese-34050/#subtraktiv>

Elektro-Archiv.de (2018)

<http://www.elektro-archiv.de/archiv/c/can/>

Firelight Technologies Pty Ltd, FMOD (2019)

<https://www.fmod.com/resources/documentation-studio?version=2.0&page=profiling.html>

Freesound.org (2019)

<https://freesound.org/people/evan.schad/sounds/462677/>

Grewus GmbH (2019)

<https://www.grewus.de/produkte/avas-soundmodul/>

Home & Smart, Kammerer, S. (2018)

<https://www.homeandsmart.de/elektroauto-renault-twizy-preis-reichweite-vergleich>

LattePanda (2019)

<https://www.lattepanda.com/products/lattepanda-4g-64g-with-enterprise-license.html>

ME-Meßsysteme GmbH (2019)

<https://www.me-systeme.de/de/technik-zuerst/elektronik/can-bus-grundlagen>

National Highway Traffic Safety Administration (2016)

https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/documents/quietcar_finalrule_11142016.pdf

N-TV (2018)

<https://www.n-tv.de/wirtschaft/Markt-fuer-E-Roller-waechst-weiter-article20492467.html>

OpenDCC (2019)

https://www.opendcc.de/info/can/can_connector.html

PEAK-System Technik GmbH (2019)

<https://www.peak-system.com/PCAN-USB.199.0.html>

RS Components Handelsgesellschaft m.b.H. (2019)

<https://at.rs-online.com/web/p/einbau-lautsprecher/3643240/>

Smartplanet, Dignan, L. (2010)

<https://web.archive.org/web/20101219073150/http://www.smartplanet.com/business/blog/smart-takes/hybrid-electric-vehicles-to-become-louder-for-pedestrian-safety/13077/>

Spektrum.de (2019)

<https://www.spektrum.de/lexikon/physik/kurven-gleicher-lautstaerke/8648>

Wikipedia (2019)

<https://de.wikipedia.org/wiki/H%C3%B6rfl%C3%A4che#/media/Datei:Hoerflaeche.svg>