

Heinz Neidhart, BSc

Entwicklung einer Ultraschallstimulationsplattform für die Validierung von ADAS-Funktionen

Masterarbeit

zur Erlangung des akademischen Grades
Diplom-Ingenieur

eingereicht an der
Technischen Universität Graz

Betreuer

Assoc.Prof. Dipl.-Ing. Dr.techn. Hannes Wegleiter

Mitbetreuer

Dipl.-Ing. Dr.techn. Bernhard Schweighofer

Institut für Elektrische Meßtechnik und Meßsignalverarbeitung

Graz, März 2019

This document is set in Palatino, compiled with pdfL^AT_EX₂ε and Biber.

The L^AT_EX template from Karl Voit is based on KOMA script and can be found online: <https://github.com/novoid/LaTeX-KOMA-template>

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

Datum

Unterschrift

Danksagung

Diese Arbeit entstand aufgrund meiner Tätigkeit als studentischer Mitarbeiter in der Abteilung für Global Research and Technology der Firma AVL List GmbH in Graz in Kooperation mit dem Institut für Elektrische Meßtechnik und Meßsignalverarbeitung.

Ich möchte hier folgenden Personen für ihre Unterstützung und Geduld bei der Erstellung dieser Arbeit danken:

Meinen Betreuern seitens des Instituts für Elektrische Meßtechnik und Meßsignalverarbeitung, Herrn Assoc.Prof. Hannes Wegleiter und Herrn Dr. Bernhard Schweighofer.

Meinem Betreuer seitens der AVL List GmbH, DI Steffen Metzner.

Dem Head of Research and Technology der AVL List GmbH, Dr. Michael Paulweber, für die Ermöglichung dieser Arbeit.

Meinem Vater Hermann und seiner Lebensgefährtin Brigitte für ihre Unterstützung, besonders während meines Studiums.

Meiner Freundin Ulrike für ihre Unterstützung in allen Lebenslagen.

Kurzfassung

In der vorliegenden Arbeit wird ein Testsystem für das Testen von Ultraschallbasierten Fahrerassistenzfunktionen in automotiven Anwendungen präsentiert. Automotiv Ultraschallsensoren messen die Distanz zu einem Objekt über das Puls-/Laufzeitprinzip. Bei diesem wird die Zeitdauer gemessen, die ein Ultraschall-Impuls für eine bestimmte Strecke und wieder zurück benötigt. Das Testsystem spielt einer automotiven Parkassistentz ein virtuelles Objekt in einem bestimmten Abstand ein, indem es das Ultraschall-Signal empfängt, verzögert und wieder zurücksendet. Dies nennt man Stimulation. Damit lassen sich Parkassistentzen unter Laborumgebungen prüfen. Die Arbeit beschreibt zunächst die Grundlagen von Fahrerassistenzfunktionen und Ultraschall im automotiven Umfeld. Danach wird eine Parkassistentz vermessen und darauf aufbauend das Testsystem implementiert. Das Testsystem wurde evaluiert. Das Testsystem wurde in eine vehicle-hardware-in-the-loop Umgebung integriert.

Abstract

This thesis presents a test system for testing Ultrasonic-based Driver Assistance Systems. Automotive Ultrasonic sensors measure the distance to an object by measuring the elapsed time between sending an Ultrasonic impulse and receiving the echo. The testing system injects a virtual object with a certain distance into the automotive parking assistance. To do this, the testing system receives an Ultrasonic impulse, delays it and sends it back to the parking assistance. This is called stimulation. With this method, parking assistance systems can be tested in the laboratory. This thesis gives an introduction to Driver Assistance Functions in general and describes the application of Ultrasonic in automotive environments. After that a parking assistance system is characterized and the testing system is implemented. The testing system was evaluated. The testing system was also integrated into a larger vehicle-hardware-in-the-loop environment.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Zielsetzung	2
2	Grundlagen von Fahrerassistenzfunktionen	3
2.1	Fahraufgabe und Fahrerassistenzfunktionen	3
2.2	Aufbau	4
2.3	Sensoren	5
2.3.1	Ultraschall	5
2.3.2	Radar	6
2.3.3	Lidar	6
2.3.4	Kamera	7
2.4	Beispiele für Fahrerassistenzfunktionen	7
2.4.1	Einparkassistent	7
2.4.2	Adaptive Cruise Control (ACC)	8
2.4.3	Frontkollisionsschutz	8
2.4.4	Querführungsassistent	9
2.5	Testen und Validieren von Fahrerassistenzfunktionen	9
3	Problemstellung der Ultraschall-Stimulation	11
3.1	Ultraschallsensoren für Einparkassistenten	12
3.1.1	Trilateration und Kreuzecho	13
3.1.2	Piezoelektrische Schallwandler	14
3.1.3	Ausbreitung und Reflexion von Ultraschall	15
3.2	Ultraschalldistanzmess-Systeme	18
3.2.1	Ansteuerung Sendesignal	18
3.2.2	Auswertung Empfangssignal	19
3.3	Analyse und Charakterisierung bestehender Systeme	20
3.3.1	Parkassistent - Valeo Beep&Park	20
3.3.2	Ultraschall-Transducer	23

3.4	Systemdesign	26
4	Implementierung der Ultraschall-Stimulation	29
4.1	Hardware-Implementierung	29
4.1.1	Mikrocontroller	29
4.1.2	Spannungsversorgung	30
4.1.3	Preconditioning und Signalverarbeitung	31
4.1.4	Echo-Erzeugung	35
4.1.5	Kommunikationsschnittstelle	35
4.1.6	Temperatur-Sensor	35
4.1.7	Platine und Gehäuse	36
4.2	Signal-Detektion und -Verzögerung	37
4.3	Kommunikation	40
4.3.1	Kommunikation zwischen den Knoten	40
4.3.2	CAN-Kommunikation	43
5	Evaluierung und Integration der Ultraschall-Stimulation	45
5.1	Räumliche Auflösung	45
5.2	Updaterate	46
5.3	Stimulationsdistanz	47
5.4	Stromverbrauch	47
5.5	Integration am AVL DrivingCube	48
5.5.1	Simulationsmodell	49
5.5.2	Virtueller Sensor	51
5.5.3	Einpark-Szenario	53
6	Diskussion	57
	Appendix	59
	Abbildungsverzeichnis	64
	Literatur	65

1 Einleitung

Fahrerassistenzfunktionen, auch *Advanced Driver Assistance Systems* (ADAS) genannt, sind automatisierte Funktionen in Kraftfahrzeugen, welche den Fahrer in seinen Fahrtätigkeiten unterstützen oder ihm diese komplett abnehmen. Solche Systeme können in Autos unterschiedlichste Fahrfunktionen übernehmen. Grundvoraussetzung für eine Fahrerassistenzfunktion ist die Aufnahme der relevanten Umgebungssituation mittels verschiedener Sensoren.

Die Sicherstellung der funktionalen Sicherheit ist von großer Bedeutung in der Fahrzeugentwicklung und damit auch in der Entwicklung von ADAS-Funktionen. Hierzu müssen die Systeme im Entwicklungsprozess getestet und validiert werden. Das Testen und Validieren von ADAS-Funktionen auf realen Straßen und Teststrecken ist aufgrund der Fülle der möglichen Testszenarien eine langwierige und kostspielige Aufgabe. Daher wird bestrebt, diese Tests in einer Labor- oder Prüfstands Umgebung durchzuführen. Dies soll einerseits zu Zeit- und Kosteneinsparungen führen, andererseits soll damit auch eine nachvollziehbare und wiederholbare Testmethodik geschaffen werden.

Um solche Tests in Laborumgebungen durchführbar zu machen, muss das Fahrerassistenzsystem auch im Labor die relevanten Umgebungssituation erfassen können. In dieser Arbeit wird mit Hilfe des Ansatzes der Sensor-Stimulation gezeigt, wie am konkreten Beispiel eines Ultraschall-Sensors die Umgebungssituation simuliert werden kann. Ein solcher Ultraschall-Stimualtor kann anschließend in eine *vehicle hardware-in-the-loop* (VEHIL)-Umgebung, wie z.B. dem *DrivingCube* der Firma AVL List GmbH integriert werden.

Zunächst wird in Kapitel 2 eine Einführung in Fahrerassistenzsysteme und deren Bedeutung gegeben. In Kapitel 3 wird die Problemstellung

der Ultraschall-Stimulation beschrieben. Hierzu werden zunächst grundlegende Eigenschaften von automotiven Ultraschall-Sensoren erforscht und anschließend mit Hilfe einer konkreten Parkassistentz ein Anforderungsprofil und ein Systemdesign an einen Ultraschall-Stimulator abgeleitet. Kapitel 4 leitet das Systemdesign in eine konkrete Implementierung auf Hard- und Softwareebene über. Kapitel 5 zeigt Messergebnisse des entwickelten Ultraschall-Stimulators und beschreibt, wie dieser in die *vehicle hardware-in-the-loop*-Umgebung des *AVL DrivingCube* integriert werden kann. Abschließend wird in Kapitel 6 die Arbeit nochmals zusammengefasst, die Lösung diskutiert und ein Ausblick auf offene Fragestellungen gegeben.

1.1 Zielsetzung

Das Ziel dieser Arbeit ist es, die Anforderungen an einen Ultraschall-Stimulator anhand einer konkreten Einparkhilfe abzuleiten, einen solchen zu implementieren und anschließend diesen in eine *vehicle hardware-in-the-loop*-Umgebung zu integrieren.

2 Grundlagen von Fahrerassistenzfunktionen

Dieses Kapitel bietet eine Einführung in Fahrerassistenzfunktionen und die besondere Bedeutung der Sensorik für diese. Ebenso wird die Bedeutung der Validierung für Fahrerassistenzsysteme geklärt.

2.1 Fahraufgabe und Fahrerassistenzfunktionen

Fahrerassistenzfunktionen versuchen den Fahrer eines Kraftfahrzeuges entweder durch unterstützende Maßnahmen zu entlasten oder ihm die Fahraufgabe komplett abzunehmen. Das Führen eines Kraftfahrzeuges erfordert eine ständige Überwachung und Regelung des Systems *Fahrzeug*. Um die Fahraufgabe zu beschreiben, gibt Winner u. a., 2015 (S. 18) eine Definition der Fahraufgabe an:

Es gilt, das Fahrzeug mit seinen Passagieren oder seinem Transportgut unter Nutzung der verfügbaren sensorischen Informationen mit Hilfe motorischer Eingriffe über die Betätigungseinrichtungen des Fahrzeuges von einem Ausgangsort zu einem Zielort zu bringen.

Anhand dieser Definition lässt sich bereits grundlegend ableiten, wie eine Fahrerassistenzfunktion die Fahraufgabe durch technische Einrichtungen unterstützt oder abnimmt: Die Funktion muss die sensorischen Informationen des aktuellen Fahrumfelds aufnehmen und anschließend diese Informationen, mit Hilfe von Stellgliedern, in eine Bewegung umsetzen.

Fahrerassistenzfunktionen lassen sich hinsichtlich ihres Automatisierungsgrades klassifizieren: von keiner Automatisierung bis hin zu vollautomatisierter Fahrzeugführung (vgl. Bundesanstalt für Straßenwesen, 2012, S.9).

2.2 Aufbau

Der grundlegende Aufbau von Fahrerassistenzfunktionen lässt sich nach Watzenig u. a., 2017 (S. 6) in drei aufeinander aufbauenden *Building Blocks* gliedern. Diese Ebenen sind in Abbildung 2.1 dargestellt.

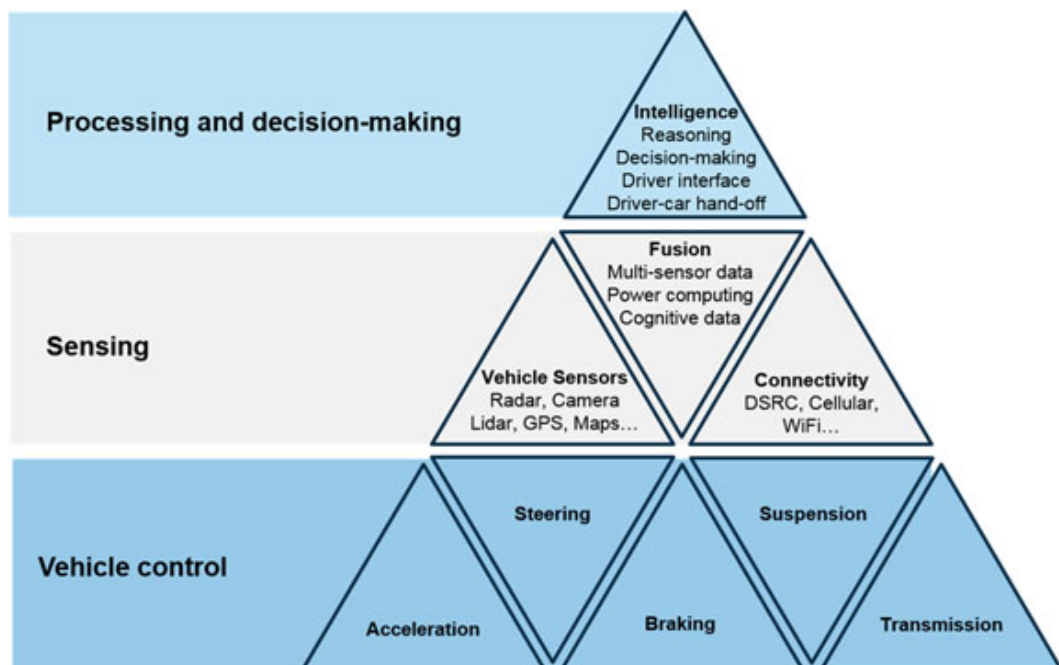


Abbildung 2.1: Building Blocks von Fahrerassistenzfunktion aus Watzenig u. a., 2017, S. 6

In der untersten Ebene muss die Funktion Zugriff auf die laterale und longitudinale Steuerung des Fahrzeuges besitzen, um überhaupt eine Wirkung auf die Fahrzeugführung ausüben zu können. Je nach Funktion und Automatisierungsstufe kann hierbei der Zugriff auf die Steuerung der Beschleunigung, der Lenkung, des Bremsens, der Federung und der Schaltung

erfolgen. Diese Zugriffe erfolgen über die jeweiligen Steuergeräte im Fahrzeug, welche getrennt von der Fahrerassistenzfunktion sind.

Die zweite Ebene beschreibt die Erfassung der aktuellen Umgebungssituation des Fahrzeuges. Hierzu wird mit geeigneten Fahrzeugsensoren (Radar, Kamera, Lidar...) die aktuelle Umgebung des Fahrzeuges erfasst, diese eventuell mit Informationen der Straßeninfrastruktur oder anderen Fahrzeugen (*Connectivity*) ergänzt und in einem Fusionsschritt zum Umgebungsmodell zusammengeführt.

Dieses Umgebungsmodell dient als Eingang für den Schritt *processing and decision-making*. In diesem Schritt wird eine optimale Reaktion hinsichtlich der gegebenen Rahmenbedingungen (selektierte Geschwindigkeit, Abstand zum Vorderfahrzeug, Verlauf der Fahrspur...) berechnet. Diese Reaktion wird anschließend wieder der untersten Ebene zugeführt, welche diese dann fahrdynamisch ausführt.

2.3 Sensoren

Um das Umgebungsmodell zu erzeugen, werden verschiedene Sensoren eingesetzt. Hier soll kurz ein Überblick über die wichtigsten Sensoren nach Winner u. a., 2015 gegeben werden.

2.3.1 Ultraschall

Schallwellen, die eine Frequenz oberhalb des vom Menschen wahrnehmbaren Frequenzbereiches besitzen (ab ca. 16 kHz), werden als Ultraschall bezeichnet. Ultraschall wird zur Messung von Abständen zu Objekten im Nahbereich bis maximal 4m des Fahrzeuges eingesetzt.

Ultraschallsensoren sind kostengünstig und witterungsunabhängig (vgl. Winner u. a., 2015, S. 256). Ebenso bietet Ultraschall ein gutes Reflexionsverhalten an allen im Straßenverkehr üblicherweise auftretenden Materialien. Durch die kostengünstige Herstellung werden oftmals vier oder mehr Sensoren verbaut, um so einen möglichst großen Bereich abdecken zu können.

Derzeit werden für automobiler Sensoren Ultraschallfrequenzen im Bereich von 40 kHz bis 50 kHz verwendet. Dieser Frequenzbereich bietet einen Kompromiss aus Robustheit gegen Fremdschallgeräuschen (aus tieferen Frequenzen von der Fahrzeugumgebung) und zu geringen Echoamplituden (durch Verwendung höherer Frequenzen) (vgl. Winner u. a., 2015, S. 247).

2.3.2 Radar

Radio Detection and Ranging, kurz Radar, basiert auf der Abstrahlung elektromagnetischer Wellen und wird für die Messung der Distanz zu einem Objekt über größere Entfernungen (bis 200 m) bei gleichzeitiger Messung der Relativgeschwindigkeit und des Relativwinkels verwendet. Im Automobilbereich können hierbei die Frequenzbänder 24,0-24,25 GHz, 76-77 GHz, 77-81 GHz und 21,65-26,65 GHz verwendet werden. Die größte technische Bedeutung besitzt dabei der Bereich um 76,5 GHz.

Um Distanz und Relativgeschwindigkeit zu ermitteln, wird das sogenannte FMCW (*Frequency Modulated Continuous Wave*)-Verfahren (auch: Dauerstrichradar) angewendet (vgl. Reif, 2016, S. 52).

2.3.3 Lidar

Lidar bezeichnet ein Messprinzip, welches wie Radar auf elektromagnetischen Wellen basiert. Da Lidar jedoch im optischen Bereich (Ultraviolett, Infrarot und sichtbares Licht) mit Laserimpulsen arbeitet, wird es daher *Light Detection and Ranging* genannt. Lidar wird ebenso wie Radar zur Messung der Distanz und Relativgeschwindigkeit verwendet.

Um die räumliche Ausdehnung des Lidarsensors auf einen 360°-Messbereich zu erweitern, wird der Laserstrahl durch eine geeignete Vorrichtung mechanisch geschwenkt (z.B. über Ablenkung durch einen rotierenden Spiegel). Der Laser ist dabei impuls gesteuert und teilt den Erfassungsbereich so in eine Reihe von diskreten Messpunkten, die mit Sampleraten im Kilohertzbereich abgetastet werden (vgl. Hälker u. a., 2018).

2.3.4 Kamera

Digitale Kamerasysteme im Fahrzeug dienen der Erfassung von Verkehrszeichen, Fahrbahnmarkierungen und anderen Objekten (z.B. Fahrzeugen und Personen). Die Kamera soll dabei eine möglichst vollständige Erfassung des Umfelds ermöglichen (vgl. Winner u. a., 2015, S. 349). Das Bild der Kamera kann dem Fahrer auch direkt auf einem Bildschirm im Innenraum angezeigt werden (z.B. Rückfahrkamera).

Eine besondere Anordnung von zwei Kameras zur präzisen Messung von Tiefeninformationen nennt sich Stereokamera. Dabei werden zwei Kameras nebeneinander platziert und synchronisiert ausgelöst. Aus den aufgenommenen Halbbildern kann anschließend eine Tiefeninformation (z.B. Abstand zum Vorderfahrzeug) gewonnen werden.

2.4 Beispiele für Fahrerassistenzfunktionen

2.4.1 Einparkassistent

Einparkassistenten unterstützen den Fahrer beim Parken hinsichtlich der lateralen Führung des Fahrzeuges. Der Einparkassistent wirkt daher hauptsächlich auf das Lenksystem. Als Umfeldsensoren verwenden Parkassistenten Ultraschallsensoren und Kameras. Der Parkassistent erkennt zunächst über die Umfeldsensorik eine freie Parklücke und berechnet anschließend laufend eine Trajektorie, um das Fahrzeug in diese Parklücke zu führen. Nach Winner u. a., 2015 (S. 842) lassen sich vier verschiedene Arten von Parkassistenten unterscheiden:

- Informierend: Der Assistent zeigt nur die Umfeldinformationen
- Geführt: Der Assistent zeigt eine Lenkempfehlung, die vom Fahrer befolgt werden kann
- Semiautomatisch: Der Fahrer übernimmt die Längssteuerung, der Assistent führt die Quersteuerung automatisiert aus
- Vollautomatisch: Der Assistent übernimmt Quer- und Längssteuerung

2.4.2 Adaptive Cruise Control (ACC)

Eine Weiterentwicklung des klassischen Tempomaten, Adaptive Cruise Control, auch Abstandsregeltempomat genannt, dient zur automatisierten und verkehrsangepassten Fahrgeschwindigkeitsregelung. Hierzu wird dem System eine Sollgeschwindigkeit und eine Zeitlücke, genannt *time gap* τ , vorgegeben. Das ACC regelt nun die Geschwindigkeit des eigenen Fahrzeuges so, dass dieses entweder der Sollgeschwindigkeit oder einem etwaigen Vorderfahrzeug in Abhängigkeit der Zeitlücke folgt. Üblicherweise übt ein aktiviertes ACC volle Kontrolle über die longitudinale Steuerung des Fahrzeuges aus, d.h. sowohl Gas als auch Bremse werden durch das ACC gesteuert (vgl. Winner u. a., 2015, S. 852).

Der Umfeldsensorik des ACC kommt dabei besondere Bedeutung zu. Die Umfeldsensorik muss das relevante Vorderfahrzeug identifizieren, dessen Relativgeschwindigkeit und Abstand bestimmen und dieses Fahrzeug kontinuierlich tracken. Zum Einsatz kommen hierbei Radar- und Lidar-Sensoren.

2.4.3 Frontkollisionsschutz

Der Frontkollisionsschutz soll die Überlebenschancen eines Fahrers bei einer frontalen Kollision mit einem Gegenfahrzeug erhöhen bzw. diese Kollision im besten Fall verhindern.

Zum Frontkollisionsschutz gibt es nach Winner u. a., 2015 (S. 915) zahlreiche Ausprägungen, die sich in ihren Warnungen, aktiven Bremsengriffen und Eingriffzeitpunkten unterscheiden. In heutigen Fahrzeugen findet man von Fahrzeugherstellern vielfach als *Autonomous Emergency Brake (AEB)* bezeichnete Systeme. Diese Systeme nutzen einen aktiven Bremsengriff mit maximaler Verzögerung um eine Frontkollision zum spätest möglichen Zeitpunkt noch zu verhindern.

Die eingesetzte Sensorik unterscheidet sich je nach Hersteller. Oftmals werden die Sensoren aus anderen Systemen (z.B. Radarsensor vom ACC) auch

für den Kollisionsschutz verwendet. Als Sensorik für den Frontkollisionsschutz werden daher alle verfügbaren Sensoren verwendet (Radar, Kamera, Ultraschall und Lidar).

2.4.4 Querführungsassistentz

Eine Querführungsassistentz unterstützt den Fahrer bei der lateralen Führung des Fahrzeuges, auch bei hohen Geschwindigkeiten. Voraussetzung hierfür ist die korrekte Erkennung der aktuell benutzten Fahrspur und des weiteren Fahrstreifenverlaufes. Nach Winner u. a., 2015 (S. 940) kann die Querführungsassistentz entweder rein informierend ausgelegt sein oder mit aktiven Lenkeingriffen den Fahrer unterstützen.

Zur Erkennung des Fahrstreifens werden Kameras verwendet. Aus den aufgenommenen Kamerabildern extrahiert eine Bildverarbeitung anschließend die relevanten Eingangsgrößen wie z.B. Position der Fahrstreifen, Abstand des Fahrzeuges von den Fahrstreifen und den weiteren Verlauf.

2.5 Testen und Validieren von Fahrerassistenzfunktionen

Bei der Entwicklung von Fahrerassistenzfunktionen spielt das Testen und anschließende Validieren eine große Rolle. Bei der Validierung wird geprüft, ob die Funktion alle Entwicklungsziele, vor allem hinsichtlich funktionaler Sicherheit erfüllt (vgl. Winner u. a., 2015, S. 98).

In Watzenig u. a., 2017 (S. 444) wird dargestellt, wie die Toolchain zur Validierung einer Fahrerassistenzfunktion aussehen kann. Abbildung 2.2 (S. 10) zeigt den Aufbau einer solchen Toolchain. Kernstück der Toolchain ist ein V-Modell, in dem die Validierung von einer reinen virtuellen *Model-in-the-loop* (MiL) Umgebung über eine *Software-in-the-loop* (SiL) in eine *Hardware-in-the-loop* (HiL) überführt wird. Mittels *X-in-the-loop* (XiL), komplette Integration aller relevanten Komponenten, und *Proving ground in-the-loop* wird in der Umgebung schließlich das Gesamtfahrzeug validiert.

Für die Validierung müssen Validierungsszenarios erstellt und ausgewählt werden. Zunächst werden relevante und geeignete Szenarios durch verschiedene Quellen festgelegt. Die Szenarios werden anschließend über ihre Parameter (Straßenbedingungen, Wetter, Tag/Nacht...) variiert. Diese erzeugt eine große Fülle an möglichen Szenarios. Ein weiterer Schritt reduziert diese Szenarios auf jene, die für die Funktion und den aktuellen Schritt im V-Modell notwendig sind.

In jedem Schritt des V-Modells werden mehr und mehr reale Komponenten in die Testumgebung integriert. Dadurch steigen in jedem Schritt auch die Anforderungen an die Umgebung, ein möglichst realistisches virtuelles Weltmodell (*virtual world model*) zur Verfügung zu stellen.

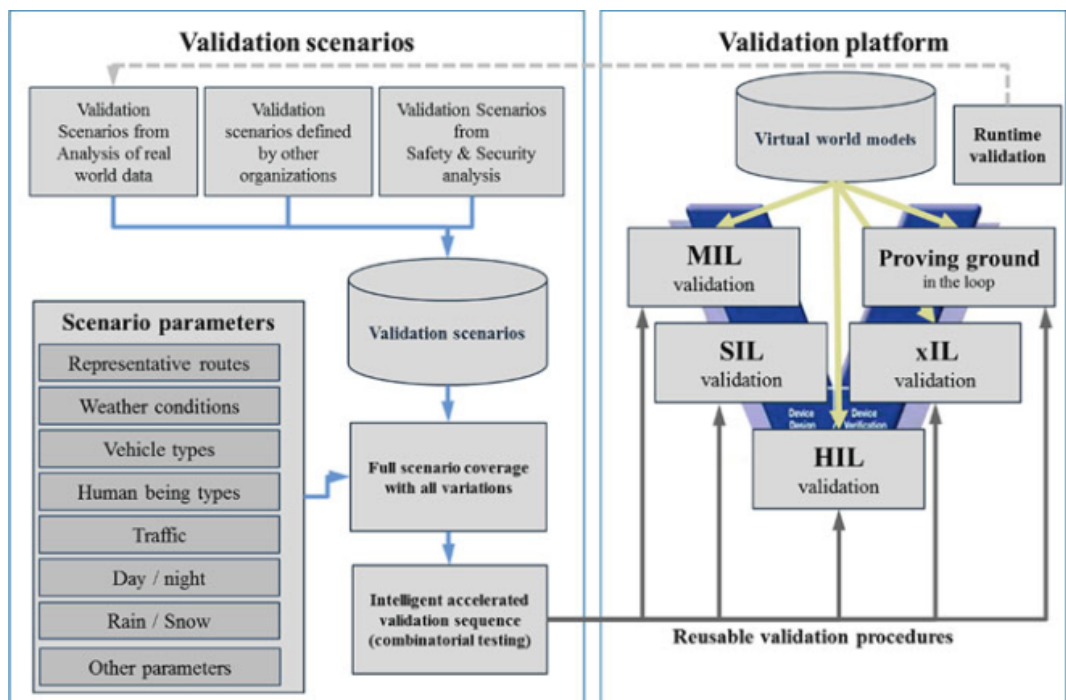


Abbildung 2.2: Validation Toolchain aus Watzenig u. a., 2017, S.444

Ein Ultraschall-Stimulator ist dabei Teil dieser Toolchain in den Schritten der HiL- und XiL-Validierung. Er dient dazu, einer Funktion, die auf Ultraschall-Sensoren basiert, im Rahmen des virtuellen Weltmodells virtuelle Objekte vorzuspielen.

3 Problemstellung der Ultraschall-Stimulation

Bei der Sensor-Stimulation wird ein realer Sensor mit einem physikalischen Signal, erzeugt von einem Stimulator, beaufschlagt. Ziel der Stimulation ist es, dass der reale Sensor die durch den Stimulator injizierten Signale nicht von Signalen, wie sie in der echten Einsatzumgebung des Sensors auftreten, unterscheiden kann. Dem realen Sensor wird dadurch vom Stimulator ein virtuelles Objekt vorgetäuscht. Mithilfe der Sensor-Stimulation können Sensoren unter Laborbedingungen getestet werden. Ein konkretes Anwendungsbeispiel wird im Rahmen einer *vehicle hardware-in-the-loop*-Umgebung in Kapitel 5 präsentiert.

Bei Sensoren, die Distanzen nach dem Laufzeitprinzip messen (z.B. Ultraschall und Radar), erfolgt dies durch Verzögerung und Rücksendung des ausgesendeten Messimpulses. Der Stimulator empfängt einen Messimpuls, d.h. Ultraschall- oder Radarwelle, verzögert den Impuls und sendet diesen wieder zurück an den Sensor (vgl. Gadringer u. a., 2018). Gleichzeitig muss der Stimulator das zurückgesendete Signal so aufbereiten, dass der Sensor dieses als gültiges Signal eines virtuellen Objektes erkennt.

Im weiteren Verlauf sollen die Rahmenbedingungen eines Stimulators für einen Ultraschall-basierten Parksensoren diskutiert werden und wie aus diesen Rahmenbedingungen ein Systemdesign für einen Stimulator abgeleitet werden kann.

3.1 Ultraschallsensoren für Einparkassistenten

Ultraschallsensoren werden in Parkassistenten zur Messung von Distanzen eingesetzt (siehe Kapitel 2). Die Messung der Distanz erfolgt auf Basis des Puls-/Laufzeitprinzips, welches in Abbildung 3.1 schematisch abgebildet ist.

Zunächst wird ein Ultraschall-Sendeimpuls ausgesendet. Die ausgesendete Ultraschallwelle durchläuft das Ausbreitungsmedium (in der Regel Luft) mit einer bestimmten Schallgeschwindigkeit und wird an einem möglichen Hindernis reflektiert. Durch Kenntnis der Schallgeschwindigkeit und der Laufzeit der Welle vom Sender zum Hindernis und wieder zurück lässt sich anschließend der Messabstand rechnerisch ermitteln.

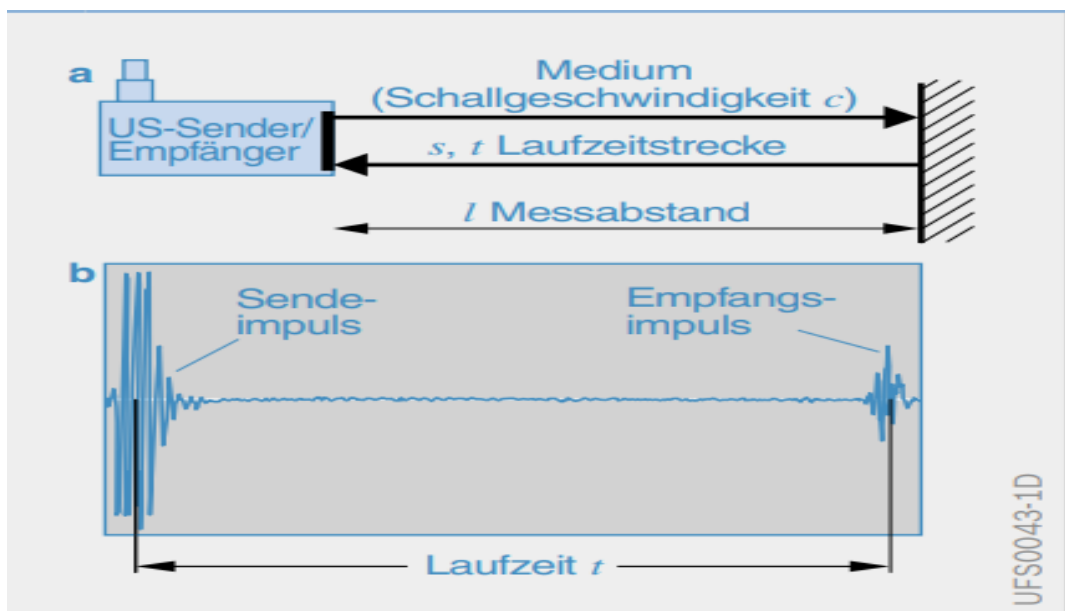


Abbildung 3.1: Schematische Darstellung des Puls-/Laufzeitprinzips (Reif, 2016, S. 50)

Der Messabstand lässt sich folgendermaßen berechnen:

$$l = \frac{t}{2c}$$

l ist dabei der Messabstand in Meter, t die Laufzeit vom Sender zum Hindernis und wieder zum Empfänger in Sekunden und c die Schallgeschwindigkeit im Ausbreitungsmedium in $\frac{m}{s}$.

In typischen Anwendungsfällen sind Ultraschall-Sensoren für Kraftfahrzeuge als gekapselte Sensorbaugruppen mit digitaler oder analoger Schnittstelle ausgeführt, welche direkt den gemessenen Messabstand ausgeben. Der erfassbare Messabstand beträgt üblicherweise nur wenige Meter (vgl. Reif, 2016, S. 50). Der Einbau erfolgt in die hintere und/oder vordere Stoßstange (vgl. Winner u. a., 2015, S. 256).

Durch diesen Aufbau ergibt sich die Forderung, dass der Stimulator vor einen eingebauten Sensor platziert und montiert werden können muss.

3.1.1 Trilateration und Kreuzecho

Eine Erweiterung des Puls-/Laufzeitprinzips ist die Trilateration. Mit dieser Technik lässt sich die Distanz zu einem Objekt noch genauer auflösen und zusätzlich eine Positionsbestimmung durchführen (vgl. Winner u. a., 2015, S. 253).

In automotiven Anwendungen ist typischerweise der kürzeste Abstand vom Fahrzeug zum Objekt von besonderem Interesse. Nach dem Puls-/Laufzeitprinzip detektiert der Sensor jegliche Reflexion im Detektionsfeld. Wenn sich das Objekt jedoch nicht auf der Normalen des Sensors befindet, so ergibt sich ein Fehler zum tatsächlichen minimalen Abstand zum Fahrzeug. Werden die Ultraschall-Sensoren wie in Abbildung 3.4 (S. 17) dargestellt überlappend eingebaut, so gibt ein Objekt, welches sich in einem der Überlappungsbereiche befindet, eine Reflexion für beide Sensoren. Wird nun mit beiden Sensoren synchron gemessen, kann aus dem gedachten Schnittpunkt der Laufzeiten der beiden Reflexionen und dem Abstand zwischen den Sensoren ein Dreieck gebildet werden. Die Höhe dieses Dreiecks ist der kürzeste Abstand vom Fahrzeug zum Objekt.

Durch zusätzliche Auswertung der Kreuzechos, das heißt die Detektion der gestreuten Reflexion eines Sensors durch einen anderen Sensor, lässt sich noch zusätzlich die Objektposition bestimmen.

3.1.2 Piezoelektrische Schallwandler

Im Automobilbereich werden fast ausschließlich piezoelektrische Schallwandler (*Transducer*) eingesetzt (vgl. Winner u. a., 2015, S. 244). Diese basieren auf dem piezoelektrischen Effekt, bei dem bei mechanischer Deformation eines Kristalls, eine dazu proportionale elektrische Spannung an diesem abgegriffen werden kann. Der piezoelektrische Effekt ist auch umkehrbar. Dies bedeutet, dass durch Anlegen einer elektrischen Spannung auch eine mechanische Deformation im Kristall hervorgerufen werden kann. Ein piezoelektrischer Schallwandler kann dadurch sowohl als Sender als auch als Empfänger eingesetzt werden.

Um den Zusammenhang zwischen mechanischer und elektrischer Energie zu beschreiben, ist die Proportionalitätskonstante d definiert (vgl. Müller u. a., 2017, S. 3). Der erste Teil beschreibt die mechanische Verformung S des Schallwandlers bei Anlegen eines äußeren elektrischen Feldes E . Der zweite Teil der Gleichung beschreibt die entstehende elektrische Verschiebung D wenn eine äußere mechanische Spannung T auf den Schallwandler wirkt.

$$d = \frac{S}{E} = \frac{D}{T}$$

Zur Modellierung der elektrischen Eigenschaften des piezoelektrischen Schallwandlers nahe seiner Resonanzfrequenz kann das in Abbildung 3.2 gezeigte Ersatzschaltbild verwendet werden.

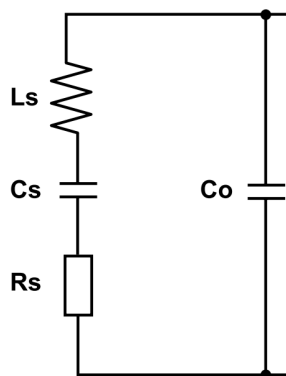


Abbildung 3.2: Elektrisches Ersatzschaltbild piezoelektrischer Schallwandler (Winner u. a., 2015, S. 248)

Die elektrischen Größen können dabei mechanischen Größen zugeordnet werden: Die Kapazität C_s und die Induktivität L_s entsprechen der Federsteifigkeit der Membran und der schwingenden Masse. Der Widerstand R_s beschreibt die Verluste durch Reibung, ferroelektrische Hysterese und Schallabstrahlung (vgl. Winner u. a., 2015, S. 247).

Die Sensitivität eines Schallwandlers beschreibt, wieviel Spannung ein Schallwandler pro Pascal Schalldruck erzeugt und wird in dB angegeben. Die Sensitivität ist abhängig von der Frequenz des auftreffenden Ultraschalls und am höchsten an der Resonanzfrequenz.

Der Schalldruckpegel beschreibt den erzeugten Schalldruck für eine bestimmte angelegte Spannung in einer bestimmten Entfernung und wird in dB angegeben. Wie die Sensitivität hängt auch der Schalldruckpegel von der Frequenz ab.

Der wichtigste Werkstoff zur Herstellung piezoelektrische Keramiken ist zur Zeit Bleizirkonattitanat (PZT), eine Mischung von Oxiden der Metalle Blei, Titan und Zirkonium, welche anschließend kalziniert wird (vgl. Winner u. a., 2015, S. 244). Das piezokeramische Blättchen wird auf eine fixierte metallische Membran geklebt und mit ihrer Resonanzfrequenz betrieben. Dadurch entsteht eine Biegeschwingung. Beim Auftreffen von Schallwellen auf die Membran entsteht ebenfalls eine Biegeschwingung, die zu einer Deformation des piezokeramischen Blättchens führt und damit zu einer Spannungsänderung.

3.1.3 Ausbreitung und Reflexion von Ultraschall

Um das Schallfeld und damit das Ausbreitungsverhalten des vom Wandler erzeugten Ultraschalls zu beschreiben, lässt sich die Schalldruckverteilung angeben. Aus der Schalldruckverteilung ergibt sich auch die Richtcharakteristik. Das Schallfeld lässt sich durch Simulation oder Messung ermitteln (vgl. Winner u. a., 2015, S. 250).

Abbildung 3.3 (S. 16) zeigt eine solche Schalldruckverteilung für einen Ultraschallwandler. Die Linien verbinden Punkte gleichen Schalldrucks. Helle Farbtöne stellen hohen Schalldruck dar, dunkle Farbtöne niedrigen

Schalldruck. Die gezeigte Verteilung ist abhängig von der Sendespannung U_S und deren Frequenz.

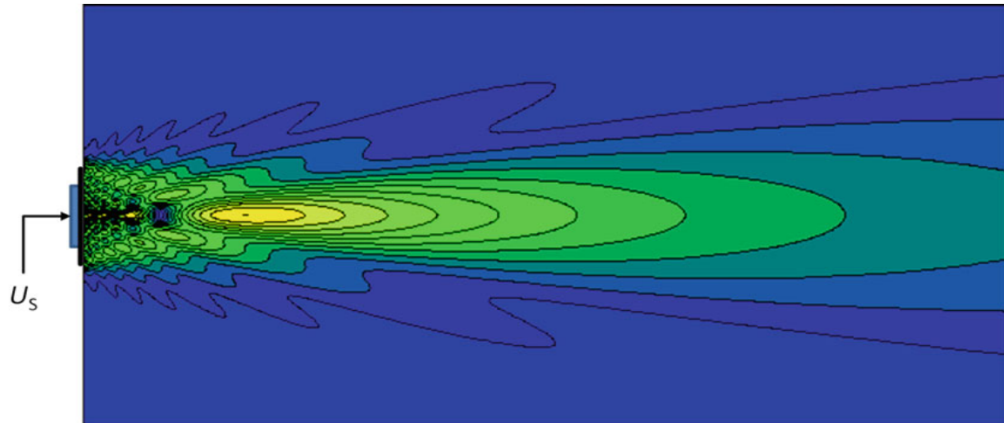


Abbildung 3.3: Beispielhafte Schalldruckverteilung für einen Ultraschallwandler (Müller u. a., 2017, S. 13)

Die Geschwindigkeit der Schallausbreitung wird als Schallgeschwindigkeit bezeichnet. Mit dieser Geschwindigkeit breiten sich Schallwellen im Medium aus. Die Schallgeschwindigkeit in Gasen wie Luft ist stark temperaturabhängig, hängt jedoch auch von anderen Parametern wie Luftfeuchtigkeit, Luftdruck und Dichte ab (vgl. Niedrig u. a., 2014, S. 231). In praktischer Literatur wird daher oft eine Approximation für die Schallgeschwindigkeit in Abhängigkeit der Lufttemperatur angegeben, wie z.B. in Analog Devices, 2015 (S. 2):

$$c = 20,0457 * \sqrt{273,15 + T}$$

Wobei c die Schallgeschwindigkeit in $\frac{m}{s}$ ausdrückt und T die Temperatur in $^{\circ}C$.

Für das Reflexionsverhalten von Ultraschall spielt das Material des Hindernisses eine große Rolle. Es lassen sich schallharte Materialien (z.B. Metalle und Kunststoffe) und schallabsorbierende Materialien (z.B. Schaumstoff) unterscheiden (vgl. Winner u. a., 2015, S. 256). Schallharte Stoffe reflektieren Ultraschall mit starken Reflektionssignalen, während schallabsorbierende

Stoffe fast den gesamten Ultraschall schlucken und somit kein Echo mehr detektiert werden kann.

In automotiven Anwendungen ist das Detektionsfeld bzw. Field-of-View mehrerer Sensoren, die z.B. an der Stoßstange eingebaut sind, von großer Bedeutung. Aus Kenntnis der Richtcharakteristik und der Schalldruckverteilung lässt sich ein Field-of-View für eine bestimmte Anordnung simulieren. Das gewünschte Field-of-View bestimmt dabei maßgeblich die Auslegung der Sensorkomponenten hinsichtlich ihrer Ansteuerung und Auswertung. Das Field-of-View lässt sich auch nachträglich messen. Abbildung 3.4 zeigt beispielhaft eine *Field-of-View*-Messung für ein Gesamtsystem aus vier Sensoren. Mit dieser Anordnung kann die Einparkhilfe im Fahrzeug den gesamten vorderen Bereich des Fahrzeuges überwachen.

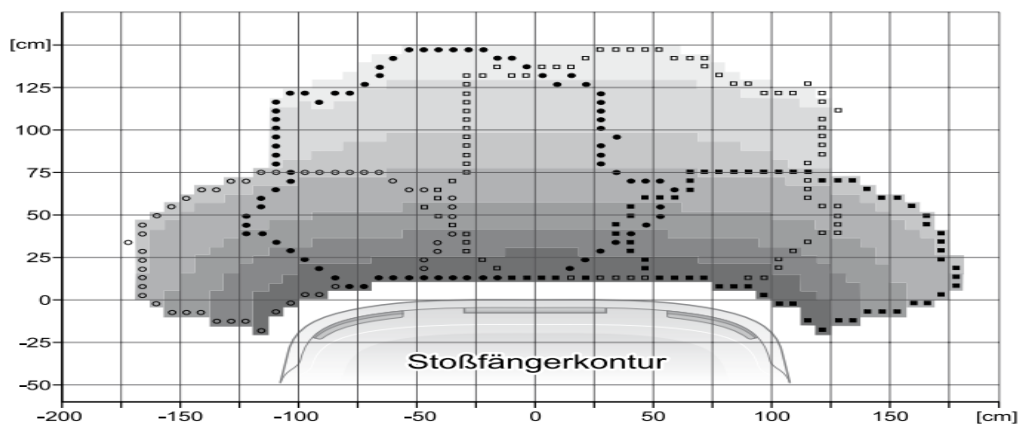


Abbildung 3.4: Beispielhafte Field-of-View-Messung für ein Ultraschallsystem mit vier Sensoren (Winner u. a., 2015, S. 256)

Ein relevanter Störeinfluss für automotive Anwendungen sind hochfrequente Störgeräusche des Fahrzeuges, wie sie z.B. an Druckluftbremsen auftreten. Diese Störgeräusche können durch Überlagerung das Signal-/Rauschverhältnis so beeinflussen, dass eine Messung nicht mehr möglich ist (vgl. Winner u. a., 2015, S. 256).

3.2 Ultraschalldistanzmess-Systeme

In weiterer Folge wird ein Überblick über Referenzdesigns von Ultraschalldistanzmess-Systemen gegeben. Dazu wurden vier Application Notes von verschiedenen Herstellern ausgewertet.

Application Note	Reichweite	Transducer
CN-0343 (vgl. Analog Devices, 2015)	0,5 m - 10 m	2
TIDUAI8A (vgl. Texas Instruments, 2015)	0,15 m - 2,5 m	2
AN76530 (vgl. Cypress, 2017)	0,2 m - 10 m	1
AN4841 (vgl. NXP, 2014)	0,2 m - 4 m	1

Tabelle 3.1: Übersicht ausgewählter Ultraschalldistanzmess-Systeme

Die Anzahl der Transducer hat großen Einfluss auf die realisierbare untere Grenze der Messdistanz (vgl. Winner u. a., 2015, S. 248). Jede Membranschwingung hat eine ausgeprägte Auf- und Abklingdauer. Je kürzer diese beiden Zeitdauern sind, desto nähere Objekte kann der Sensor messen. Wenn der reflektierte Ultraschall noch während der Abklingphase des Transducers eintrifft, so lässt sich dieser nicht eindeutig einer Distanz zuordnen.

Die vier Designs wurden hinsichtlich zwei Designentscheidungen analysiert:

- *Ansteuerung Sendesignal*
- *Auswertung Empfangssignal*

Die Designs nutzen einen zentralen Mikrocontroller zur Steuerung der restlichen Schaltung.

3.2.1 Ansteuerung Sendesignal

Alle vier Designs nutzen als Messfrequenz 40 kHz mit Schallwandler, dessen Resonanzfrequenz bei ebendiesen 40 kHz liegt. Das 40 kHz Signal wird bei allen Designs zunächst mittels PWM in den Mikrocontrollern erzeugt und anschließend verstärkt. Die Verstärkerschaltungen erzeugen jeweils

Peak-to-Peak-Spannung V_{pp} von 20 V, 36 V und 100 V, jeweils bezogen auf 0V. Die beiden Designs mit den höchsten Reichweiten (10 m) erzeugen eine V_{pp} von 36 V und 100 V. Die beiden Designs mit den geringeren Reichweiten erzeugen eine V_{pp} 20 V. Drei unterschiedliche Verstärkerschaltungen wurden gefunden:

- *Transformator*: Das 40 kHz Signal wird über einen Transformator auf eine höhere Spannung transformiert.
- *Schwingkreis*: Ein LC-Schwingkreis wird über einen Transistor mit dem PWM-Signal periodisch angesteuert.
- *Pegelumsetzer (Level Shifter)*: Über einen IC wird das PWM-Signal logisch auf ein höheres Spannungsniveau gebracht.

Direkt nach Aussenden des Signals wird in allen Designs ein Timer zur Zeitmessung gestartet.

3.2.2 Auswertung Empfangssignal

Bei der Auswertung des Empfangssignal wird das stark abgeschwächte Signal zunächst in allen Designs verstärkt.

Das NXP-Design nutzt eine zeitabhängige Verstärkung über einen *Programmable gain amplifier*. Das Design mit der höchsten Reichweite nutzt einen zweistufigen Verstärker mit einer Verstärkung von 132. Das Design von Texas Instruments nutzt einen invertierenden Verstärker mit einer Verstärkung von 333. Das Design von Cypress verzichtet auf einen Operationsverstärker und verwendet einen NPN-Transistor in Emitterschaltung.

Drei Designs filtern das Signal mit einem 40 kHz Bandpass-Filter, realisiert über Operationsverstärker.

Die weitere Erkennung des Empfangssignals erfolgt in allen Designs über eine Komparator-Schaltung, entweder durch einen externen Komparator der einen Interrupt auslöst oder durch Verwendung der Mikrocontroller-Peripherie z.B. über eine Capture/Compare-Unit oder den AD-Wandler. Wenn das vorverstärkte und gefilterte Signal einen bestimmten Schwellenwert überschreitet, wird die Zeitmessung gestoppt und die Distanz

berechnet. Bei keinem der Designs wird die Amplitude des Signals weiter ausgewertet.

Falls innerhalb eines bestimmten Timeouts kein Signal oberhalb des Schwellenwerts registriert wird, zeigen die Systeme eine *Out of Range*-Meldung an.

3.3 Analyse und Charakterisierung bestehender Systeme

Zur Erstellung eines Anforderungsprofils für einen konkreten Stimulator wurde im weiteren Verlauf die Signalerzeugung und -auswertung einer bestehenden Parkassistentz charakterisiert und ein geeigneter Ultraschall-Transducer für den Stimulator in Verbindung mit der Parkassistentz analysiert.

3.3.1 Parkassistentz - Valeo Beep&Park

Als Parkassistentz wurde das Produkt *Valeo Beep&Park* (vgl. Valeo, 2018) verwendet. Diese informierende Parkassistentz besteht aus vier Ultraschall-Sensoren und einer LCD-Anzeige, auf der der jeweils geringste Abstand der vier Sensoren, sowie der Sensor, der das Objekt detektiert hat, angezeigt wird. Diese Parkassistentz dient zum nachträglichen Einbau in Fahrzeuge ohne Einparkhilfe.

Das System besteht aus vier Ultraschallsensoren zum Einbau an der rückwärtigen Stoßstange, einer Kontrolleinheit, die in einem Plastikgehäuse verbaut ist, und einem LCD-Display.

Zur Inbetriebnahme benötigt die Kontrolleinheit eine elektrische Versorgung mit einer Betriebsspannung von 12 V. Die Parkassistentz ist jedoch nur aktiv, wenn der Rückwärtsgang eingelegt wird. Hierzu wird im echten Fahrzeug die Versorgungsleitung der Kontrolleinheit mit der Versorgungsleitung des Rückfahrlichts verbunden. Zur Inbetriebnahme im Labor wurde diese Leitung dauerhaft mit 12 V versorgt. Die Spannungen wurden über ein

Labornetzteil erzeugt. Zur maximalen Stromaufnahme lagen vorab keine Informationen vor. Die Parkassistent zeigt Distanzen im Bereich von 0.3 m bis 1.7 m an, bei einer Auflösung von 0.1 m. Unterhalb von 0.3 m gibt das System eine Empfehlung zum Parken an, oberhalb von 1.7 m zeigt das System einen Bindestrich an.

In einem ersten Schritt wurde direkt an einem Sensor des Parkassistenten gemessen. Hierzu wurde ein Stück Kabel von der Steuerelektronik zu einem der Sensoren freigelegt und an diesem mit einem Oszilloskop gemessen.

Zunächst wurde die zeitliche Abfolge der Messimpulse gemessen. In Abbildung 3.5 konnte der Messzyklus identifiziert werden. Das System sendet in einer Abfolge drei Impulse aus. Der zweite Impuls folgt 180 ms nach dem ersten Impuls. Der dritte Impuls folgt 100 ms nach dem zweiten Impuls.

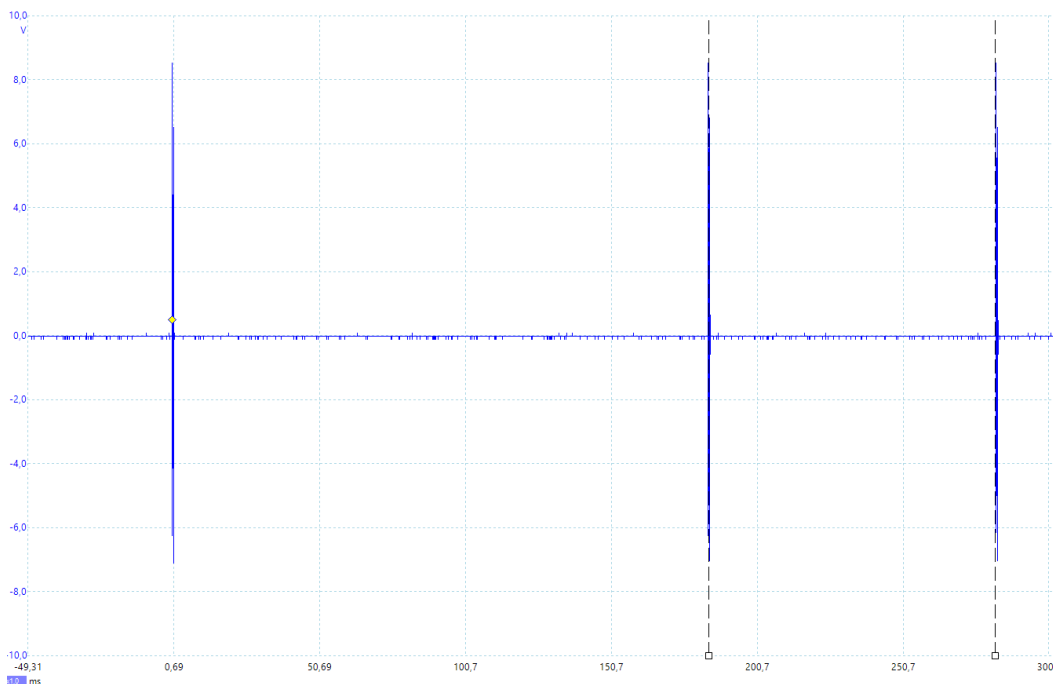


Abbildung 3.5: Zeitliche Abfolge der Ultraschall-Impulse des Valeo Beep&Park (50 ms/div, 10 V/div, 10x Tastkopf)

Weiters wurde ein Impuls vermessen. Dies ist in Abbildung 3.6 (S. 22) zu sehen. Der Sensor wird mit einer Peak-to-Peak Spannung von 120 V

mit einem sinusförmigen Signal über $500 \mu\text{s}$ angesteuert. Die Abklingzeit beträgt $250 \mu\text{s}$. Der Impuls besitzt eine Periodendauer T von $25 \mu\text{s}$, was eine Messfrequenz von 40 kHz bedeutet.

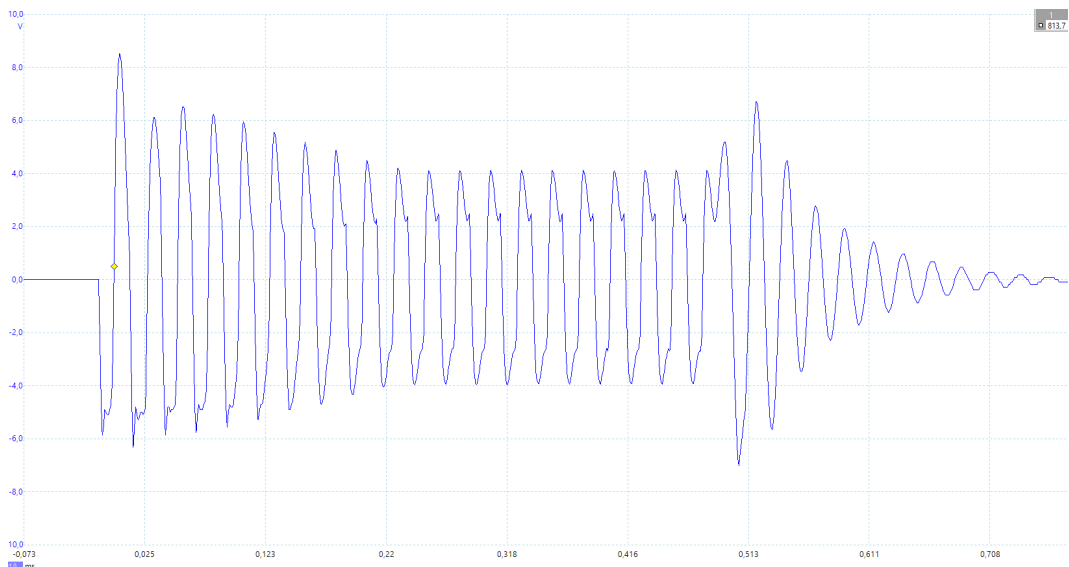


Abbildung 3.6: Valeo Beep&Park Messimpuls ($100 \mu\text{s}/\text{div}$, $1 \text{ V}/\text{div}$, $10\times$ Tastkopf)

Um zu untersuchen, ob eine Trilateration stattfindet, wurden zwei Sensoren in einem Abstand von 45 cm (siehe Einbauanleitung, vgl. Valeo, 2018) nebeneinander aufgestellt. Daraufhin wurde ein Objekt zwischen den beiden Sensoren manuell bewegt. Da die Anzeige nicht den minimalen Abstand anzeigte, wurde darauf geschlossen, dass keine Trilateration stattfindet.

3.3.2 Ultraschall-Transducer

In einem weiteren Schritt wurde ein konkreter Piezo-Transducer, auf seine Eignung als Ultraschall-Wandler für den Stimulator hin, analysiert. Der Ultraschall-Transducer ist ein Piezo-Transducer vom Typ *MCUSD14A40S09RS-30C*. Der Transducer besteht aus einem Aluminium-Zylinder mit einer Höhe von 6 mm und einem Radius von 14 mm. Die Resonanzfrequenz beträgt 40 kHz. Der Transducer besitzt eine maximale Peak-to-Peak-Spannung V_{pp} von 160 V (vgl. Multicomp, 2013).

Um das Systemdesign auszulegen, wurden am Transducer die Amplituden der, von der Parkassistentz ausgesendeten Impulse, vermessen. Der Transducer wird dazu in unterschiedlichen Abständen vor dem Ultraschallsensor des Beep&Park-Systems variiert. Der Abstand wurde dabei, nach den voraussichtlichen Abständen in einer Stimulationsanordnung im Fahrzeug, variiert (Stimulator-Transducer vor Ultraschall-Sensor). Ausgewählt wurden dabei die Abstände 5 cm, 2,5 cm und 1 cm. Gemessen wurde dabei die Peak-to-Peak-Spannung V_{pp} . Die gemessenen Amplituden sind die Tabelle 3.2 dargestellt.

Abstand cm	V_{pp} mV
1	380,10
2,5	241,60
5	95,08

Tabelle 3.2: Messergebnisse Ultraschall-Transducer <-> Valeo Beep&Park

Abbildung 3.7 (S. 24) zeigt die Messung eines empfangenen Impuls in einem Abstand von 1 cm. Charakteristisch für den Transducer ist die geringe Bandbreite des Signals. Die Dauer des Impulses stimmt mit der Dauer der Anregung seitens der Parkassistentz überein.

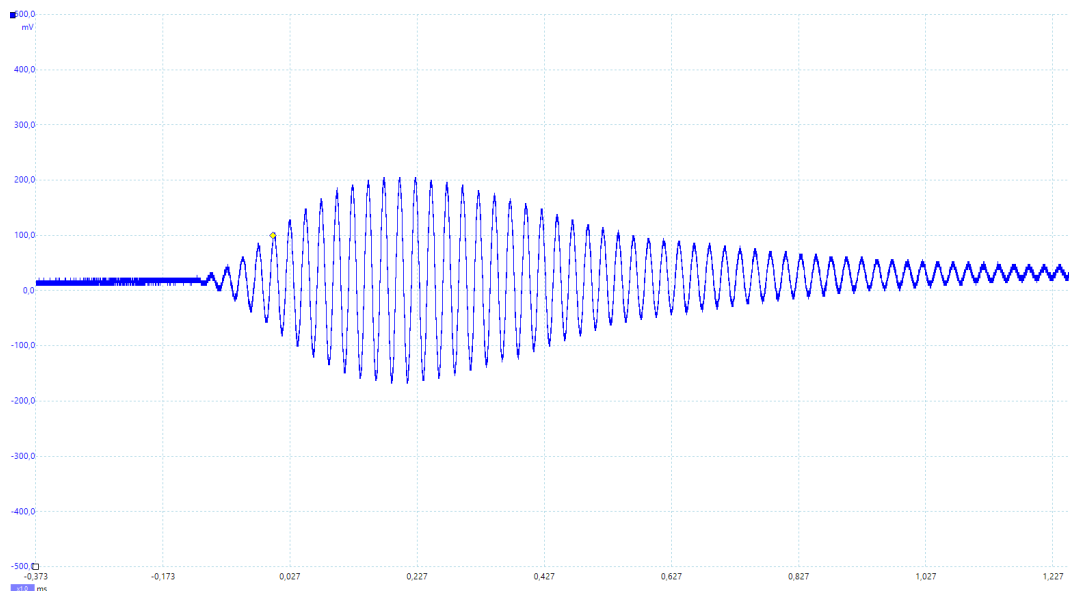


Abbildung 3.7: Aufgenommener Impuls Ultraschall-Transducer \leftrightarrow Valeo Beep&Park ($200 \mu\text{s}/\text{div}$, $100 \text{ mV}/\text{div}$)

Um die benötigte Amplitude des Ansteuersignals beim Stimulator abschätzen zu können, wurden die Amplituden bei verschiedenen Ansteuerspannungen gemessen. Hierzu wurde auf der einen Seite ein Transducer mit einem Funktionsgenerator periodisch mit einem 40 kHz Rechtecksignal angesteuert mit einer bestimmten Peak-to-Peak Spannung V_{pp} . Auf der anderen Seite wurde ein zweiter Transducer in einem bestimmten Abstand platziert und die Peak-to-Peak Spannung des Empfangssignals gemessen. Tabelle 3.3 (S. 25) zeigt die gemessenen Amplituden.

Abstand cm	V_{pp} Sender V	V_{pp} Empfänger mV
1	1	18,80
	5	73,60
	10	148,00
	15	206,00
	20	272,0
2,5	1	-
	5	13,20
	10	26,20
	15	38,00
	20	42,00
5	1	-
	5	17,00
	10	32,00
	15	46,40
	20	60,00

Tabelle 3.3: Messergebnisse Ultraschall-Transducer <-> Ultraschall-Transducer

3.4 Systemdesign

Ausgehend von der Analyse des Parkassistenten konnte ein Systemdesign ausgearbeitet werden. Das Systemdesign sieht einen Stimulator vor, der gleichzeitig mehrere Sensoren stimulieren kann und der eine Verbindung zu einer externen Umgebung besitzt, um Stimulationsdistanzen zu setzen.

Der Stimulator muss ein Interface zu einer externen Umgebung besitzen, während die Stimulatoren der einzelnen Sensoren untereinander mit einem Bussystem verbunden sind. Das Bussystem wird als *Daisy Chain* ausgeführt, um durch das Kommunikationsprotokoll eine Äquivalenz zwischen physikalischer und topologischer Struktur zu erreichen. Für diese *Daisy Chain* wird eine auf RS-485 basierende serielle Schnittstelle spezifiziert. Als Verbindung zu einer externen Umgebung wird eine CAN-Schnittstelle vorgesehen.

Abbildung 3.8 (S. 27) stellt schematisch das Systemdesign dar. Ein Master-Knoten übernimmt die Kommunikation mit der externen Umgebung, die Kommunikation mit den einzelnen Sensor-Knoten und deren Verwaltung. Die Sensor-Knoten besitzen je zwei RS-485-Schnittstellen und eine Ultraschall-Stufe. Die Ultraschall-Stufe der Sensorknoten bildet das Stimulator-Interface zu den Ultraschall-Sensoren des Fahrzeugs.

Aus dem übergeordneten Systemdesign wurden je ein Design für den Master- und einen Sensorknoten ausgearbeitet. Abbildung 3.9 (S. 28) zeigt den Aufbau des Master-Knotens. Dieser besteht aus einer Steuerung, einem CAN-Transceiver und einem RS-485-Transceiver. Die Steuerung übernimmt die Kommunikation am CAN-Bus, an der RS-485-Schnittstelle und die Verwaltung der Knoten.

Abbildung 3.10 (S. 28) zeigt den Aufbau eines Sensor-Knotens. Die Pre-Conditioning-Stufe verstärkt das empfangene Signal und bereitet es so auf, dass es von der Steuerung ausgewertet werden kann. Die Steuerung triggert die Signalerzeugung nach der vom Master-Knoten vorgegebenen Distanzinformation. Die Signalerzeugung generiert anschließend ein, für die Parkassistenz passendes, Echo-Signal. Da die Schallgeschwindigkeit und damit die Dauer der Verzögerung von der Lufttemperatur abhängt, ist an die Steuerung auch ein Temperatursensor angeschlossen.

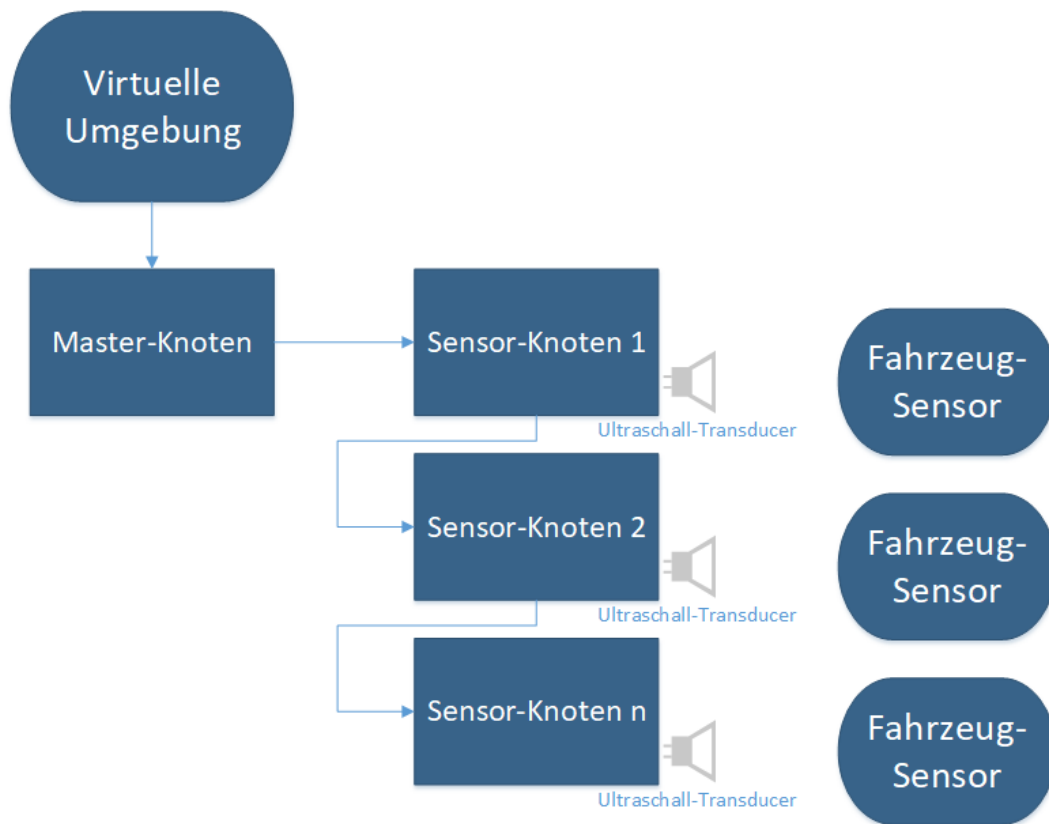


Abbildung 3.8: Aufbau des Ultraschall-Stimulators

Aus den Messungen der Analyse wurde klar, dass ein möglichst geringer Abstand zwischen Stimulations-Transducer und Fahrzeug-Sensor anzustreben ist. Dies vereinfacht die Auswertung des Empfangssignals. Auf der anderen Seite kann dadurch auch gleichzeitig die Verstärkung und Ansteuerung des Transducers vereinfacht werden. Der Ultraschall-Transducer soll daher nur rund 1 cm gegenüber des Fahrzeug-Sensors entfernt platziert werden. Die Verstärkung kann dadurch sehr gering ausfallen. Um mögliche Störreflexionen aus der Umgebung zu unterdrücken und um eine einfache Montage zu ermöglichen, soll außerdem ein Gehäuse, in dem die Elektronik des Sensor-Knotens und der Transducer Platz finden, entwickelt werden.

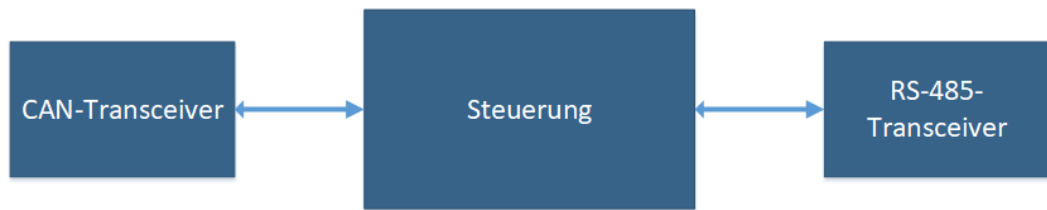


Abbildung 3.9: Design des Master-Knotens

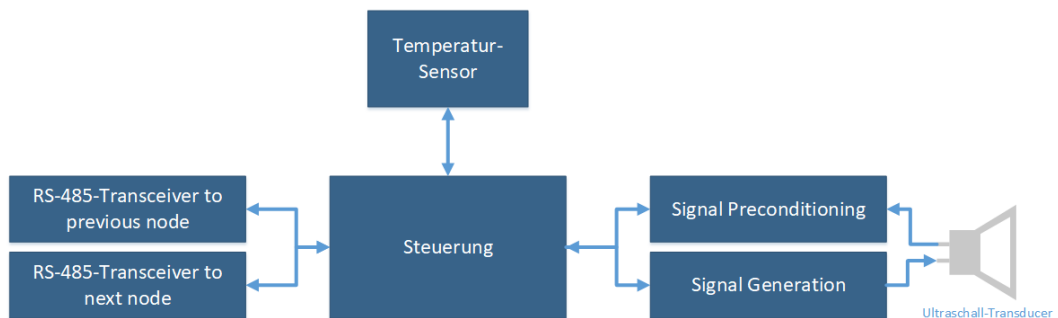


Abbildung 3.10: Design eines Sensor-Knotens

Aus der Analyse wurde darauf geschlossen, dass gängige Ultraschalldistanzmess-Systeme die Signal-Amplitude nur bis zu einem gewissen Schwellenwert auswerten und die Distanzmessung hauptsächlich aufgrund der Zeitdifferenzen geschieht. Bei der Signal-Generierung ist daher nach der Verzögerung nur eine Anregung des Transducers notwendig, um bei der Parkassistenz diesen Schwellenwert zu erreichen.

4 Implementierung der Ultraschall-Stimulation

Dieses Kapitel stellt den im Rahmen dieser Arbeit entwickelten Ultraschall-Stimulator vor.

4.1 Hardware-Implementierung

Nach den grundlegenden Designentscheidungen wurde das Design in eine konkrete Implementierung überführt. Im Folgenden sollen die Besonderheiten der Hardware-Implementierung ausgeführt werden. Im Appendix finden sich die kompletten Schaltpläne sowohl für den Master- als auch den Sensorknoten.

4.1.1 Mikrocontroller

Für die Steuerung wurde ein Mikrocontroller des Typs Atmel AT90CAN ausgewählt. Hinsichtlich der Anforderungen verfügt dieser 8-bit Mikrocontroller auf Basis der AVR-Mikroarchitektur über einen geringen Stromverbrauch, einen integrierten CAN-Controller, mehrere Timer/Counter-Module sowie zwei unabhängige UART-Module (vgl. *AT90CAN Datasheet*, 2008). Zur Programmierung und für Debuggingzwecke wurde eine JTAG-Schnittstelle im Design vorgesehen.

Als Programmiersprache für die Firmware wurde C gewählt. Als externe Bibliotheken wurden die Open-Source-Bibliotheken *AVR-UART* von Andy

Gock (vgl. Gock, 2017) und die *avrc-can-lib* von Fabian Greif (vgl. Greif, 2016) verwendet.

Der Mikrocontroller wird mit einer Taktfrequenz von 16 MHz über einen externen Quarzoszillator betrieben. Die Beschaltung der Stützkondensatoren und der Reset-Leitung wurde nach Vorgabe des Datenblattes ausgeführt.

4.1.2 Spannungsversorgung

Die notwendige Spannungsversorgung ergab sich sowohl aus den Nennspannungen der eingesetzten Komponenten, als auch deren Stromverbrauch im Betrieb. Aufgrund der Anforderung hinsichtlich des Platzverbrauches und der Anzahl der Bauteile, schied daher eine direkte Spannungsversorgung über das 230 V-Stromnetz und einem Schaltnetzteil aus.

Es wurde entschieden, dass die einzelnen Knoten über das Kabel, das auch zur Kommunikation dient, mit Strom versorgt werden. Es wurde daher vorgesehen, dass dieses Kabel auch gleichzeitig der Spannungsversorgung mit 12 V Gleichspannung dient. Diese 12 V können leicht durch ein externes Schaltnetzteil erzeugt werden und über einen Linearregler auf die für Mikrocontroller und andere Bauteile benötigten 5 V reduziert werden. Die höhere Spannung von 12 V wurde ausgewählt, da über das gemeinsame Kabel auch die Spannungsversorgung in die Daisy Chain eingebunden ist und es so eine Toleranz bezüglich Spannungsabfalles über längere Leitungen gibt. Um die Spannung von 5 V am Knoten zu erzeugen, wurde ein Linearregler vom Typ *LM7805* verwendet. Dieser benötigt als externe Beschaltung zwei Kondensatoren, je einen am Ein- und Ausgang. Der Regler hat eine Drop-Out-Spannung von typischerweise 2 V. Es muss daher sichergestellt sein, dass jeder Knoten mit zumindest 7 V versorgt wird.

Abbildung 4.1 (S. 31) zeigt die Schaltung des *LM7805* sowohl am Master als auch am Sensorknoten. Die Kondensatoren wurden dabei entsprechend dem Datenblatt (vgl. Texas Instruments, 2016) dimensioniert.

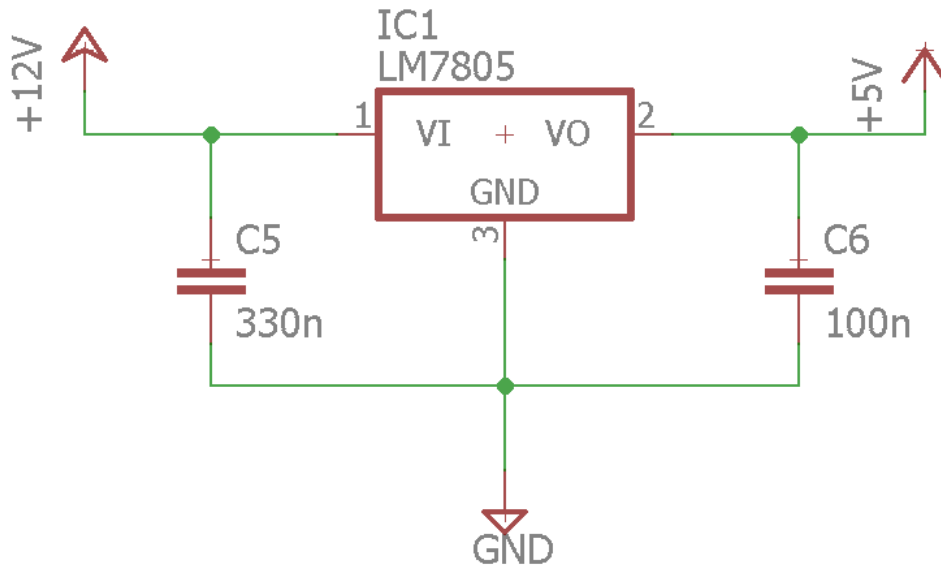


Abbildung 4.1: Beschaltung LM7805

4.1.3 Preconditioning und Signalverarbeitung

Das Signal-Preconditioning bereitet das Rohsignal vom Transducer so auf, dass es vom Mikrocontroller zur Detektierung eines empfangenen Ultraschall-Impulses verwendet werden kann. Das Preconditioning besteht dazu aus einem invertierenden Verstärker und ist in Abbildung 4.2 (S. 33) abgebildet.

Der Kondensator C10 dient als Koppelkondensator und blockiert Gleichstrom.

Die gemessene V_{pp} Spannung um 0 V bei 1 cm Abstand bei Verwendung des Valeo Beep&Park beträgt 380,10 mV (siehe Tabelle 3.2 auf S. 23). Für eine Auswertung am Mikrocontroller, der mit Spannungen von 0 V - 5 V arbeiten kann, wird das Signal auf einen Arbeitspunkt von 2,5 V gehoben und gleichzeitig verstärkt. Die Widerstände R5 und R4 setzen als Spannungsteiler den Arbeitspunkt von 2,5 V.

Als untere Schranke für eine sichere Detektion wurden 95 mV (V_{pp} bei 5 cm Abstand) angenommen. Um das Signal nun sicher zu verstärken, und auch Raum für Schwankungen der Amplitude zu lassen, wird eine Verstärkung von 20 gefordert. Diese Verstärkung lässt das Signal um den Arbeitspunkt zwischen 1,5 V und 3,44 V schwingen. Durch die Versorgung des Operationsverstärkers mit 5 V, kann das verstärkte Signal auch bei höheren Amplituden des Rohsignals nie mehr als die Maximalspannung des Mikrocontrollers von 5 V erreichen. Um diese Verstärkung bei einer Frequenz von 40 kHz zu realisieren, wird ein Operationsverstärker mit einem Gain-Bandwidth-Product von zumindest $GBP = \frac{40kHz}{20} = 800kHz$ benötigt.

Ausgewählt wurde hierzu ein Operationsverstärker vom Typ *LM321*. Dieser besitzt ein Gain-Bandwidth-Product von 1 MHz (vgl. Texas Instruments, 2014). Die geforderte Verstärkung wird über die beiden Widerstände R6 und R7 eingestellt. Diese setzen eine Verstärkung von $V = -\frac{220k\Omega}{10k\Omega} = -22$. Für die resultierende Bandbreite ergibt sich $\frac{1MHz}{22} = 45,46kHz$.

Bei einem sinusförmigen Signal mit V_{pp} von 380,10 mV ergibt sich eine Slew Rate von $SR = \frac{380,10mV * 2 * \pi}{25\mu s} = 95,53 \frac{mV}{\mu s}$. Diese Slew Rate ist deutlich unterhalb der für den LM321 angegebenen Slew Rate von $0,4 \frac{V}{\mu s}$.

Die Kondensatoren C9 und C11 dienen als Stützkondensatoren.

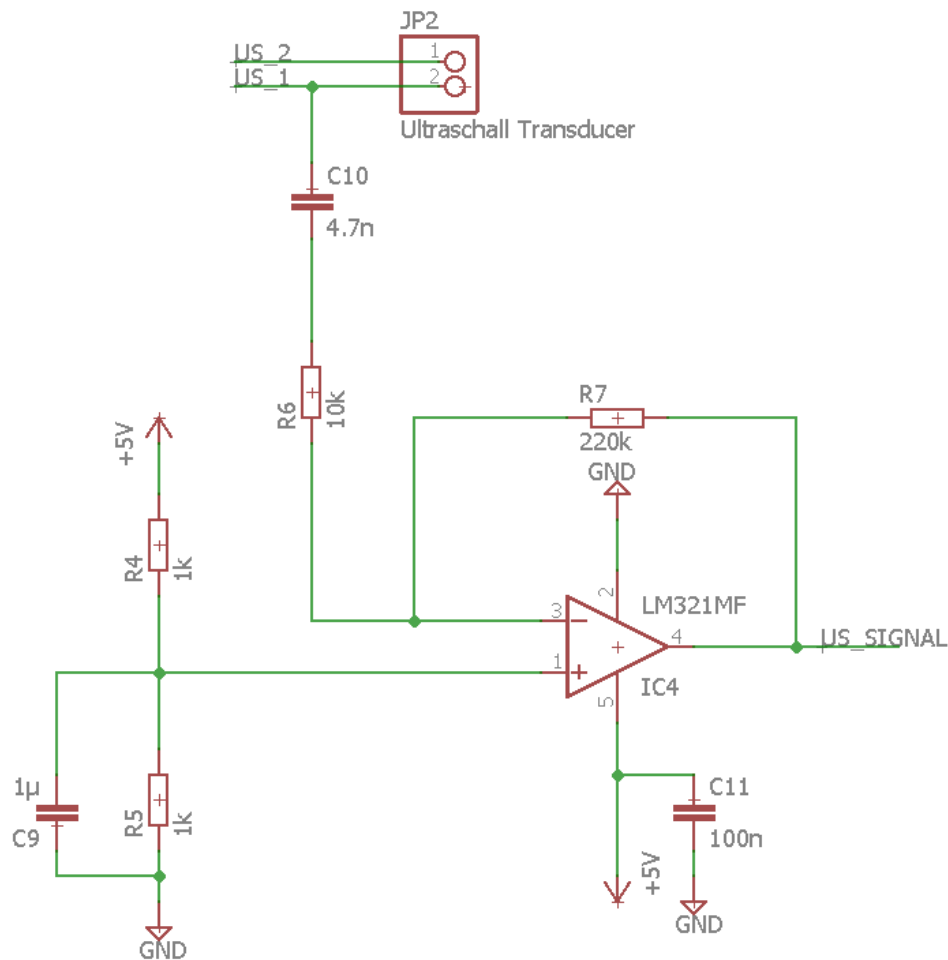


Abbildung 4.2: Schaltplan Preconditioning

Für die weitere Verarbeitung wird das verstärkte Signal auf den positiven Eingang des internen Analog-Komparators gelegt. Am negativen Eingang des Komparators wird durch einen weiteren Spannungsteiler eine Eingangsspannung von 2,72 V festgelegt. Der Komparator vergleicht kontinuierlich die beiden Eingangsspannungen. Steigt der Spannungspegel am positiven Eingang über den Pegel am negativen Eingang, so wird der Ausgang des Komparators gesetzt. Die hohe Sensitivität des Transducers bei der Reso-

nanzfrequenz von 40 kHz führte zur Wahl dieser geringen Pegeldifferenz zum Arbeitspunkt (vgl. Multicomp, 2013, S. 3). Der Transducer schwächt alle Frequenzen ober- und unterhalb von 40 kHz stark ab, somit werden Einkopplungen von unerwünschten Frequenzen minimiert. So wurde auch auf einen Filter verzichtet.

Abbildung 4.3 zeigt das Signal nach dem Preconditioning am Ausgang des Operationsverstärkers. Der Arbeitspunkts des Signals liegt bei 2,5V. Nachdem ein Signal erkannt wird, schwingt sich das Signal aus dem Ruhezustand kurz ein und erzeugt für die Dauer des Ultraschall-Impulses ein Signal für den Eingang am Mikrocontroller. Der Impuls dauert bis zum vollständigen Abklingen ca. 1,18 ms. Das ergibt sich aus der Zeitdauer der Anregung seitens des Fahrzeugsensors plus der Abklingzeit des Ultraschall-Transducers.

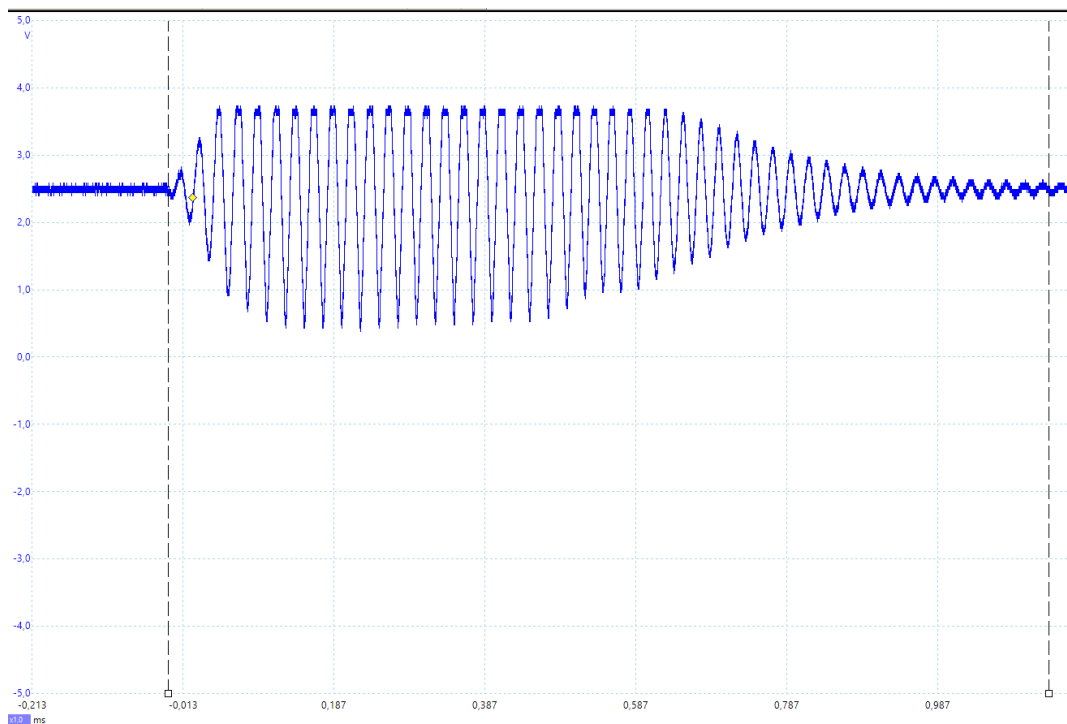


Abbildung 4.3: Signal nach Preconditioning (200 $\mu\text{s}/\text{div}$, 1 V/div)

4.1.4 Echo-Erzeugung

Wie die Messungen (siehe Tabelle 3.2 auf S. 23) ergeben haben, reichen bei kurzen Distanzen schon geringe V_{pp} -Spannungen aus, um eine Amplitude am gegenüberliegenden Sensor zu erzeugen, die voraussichtlich die Schaltschwelle übertrifft. Bei einer Distanz von 1 cm reichen schon 5 V V_{pp} aus, um ungefähr den gleichen Spannungspegel, wie bei einem Abstand von 5 cm und einem V_{pp} von 20 V, zu erzeugen. Da sich der Abstand in der Stimulations-Anordnung in einem Größenbereich von 1 cm bewegt, konnte daher die Signalamplitude sehr gering gehalten werden.

Daher wird der Transducer direkt über zwei GPIO-Pins vom Mikrocontroller angesteuert. Nach erfolgter Detektion und Verzögerung erzeugt der Mikrocontroller über zwei GPIO-Pins ein Rechteck-Signal mit V_{pp} von 5 V und einer Frequenz von 40 kHz. Das Signal wird für 500 μ s erzeugt.

4.1.5 Kommunikationsschnittstelle

Für die CAN-Kommunikation wird ein CAN-Bus-Treiber vom Typ MCP2551 verwendet. Dieser ist mit dem CAN-Controller des Mikrocontroller verbunden. Der pyhsische Anschluss ist als D-Sub DE-9 Stecker ausgeführt.

Die serielle Kommunikation zwischen den Knoten wird über eine RS-485-Schnittstelle realisiert. Hierzu wird ein halb-duplex Transceiver vom Typ *SN65HVD485* verwendet. Als physikalisches Übertragungsmedium wurde ein 4-adriges Niederspannungskabel mit Steckern der Firma Lemo (vgl. Lemo, 2018) ausgewählt. Je zwei Leitungen werden dabei für die differentielle RS-485-Übertragung verwendet und zwei weitere Leitungen für die Spannungsversorgung.

4.1.6 Temperatur-Sensor

Der Temperatur-Sensor besteht aus einem Sensor vom Typ *DS1820*. Der Sensor ist über einen GPIO-Pin an den Mikrocontroller angebunden. Die

Implementierung der Kommunikationsschnittstelle (*1-Wire-Bus*) erfolgt in Software.

4.1.7 Platine und Gehäuse

Für Master- und Sensor-Knoten wurde je eine Platine gelayoutet. Die Platinen bestehen aus 2 Layer und als Material wurde *FR4* verwendet. Fast alle Bauteile sind als SMD-Bauteile ausgeführt. Der Master-Knoten hat eine Abmessung von 80 mm x 50 mm, der Sensor-Knoten von 60 mm x 40 mm. Es wurden vor der Produktion der Platinen 3D-Renderings durchgeführt, welche in den Abbildungen 4.4a und 4.4b ersichtlich sind.

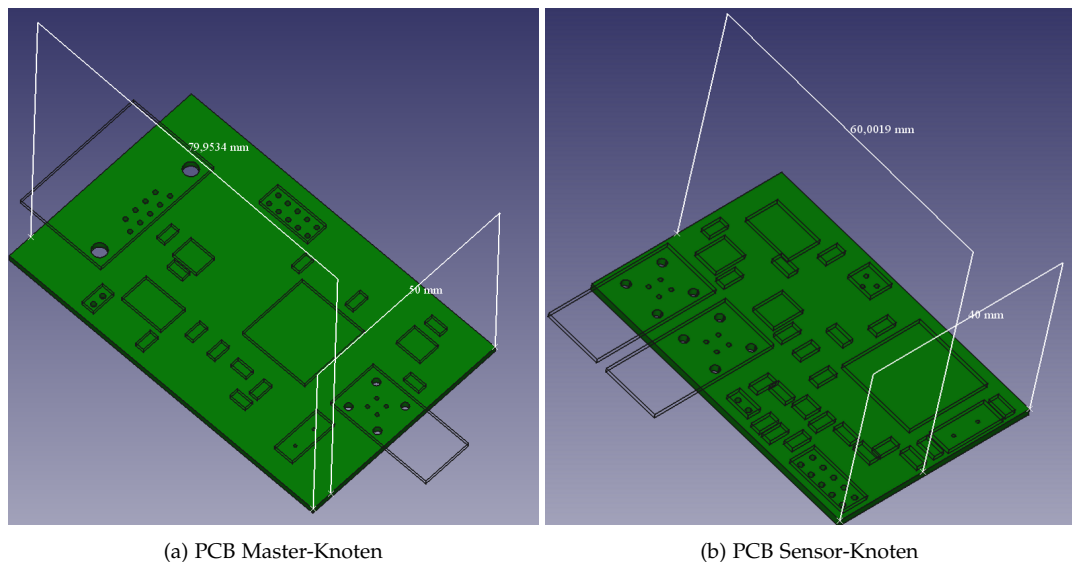


Abbildung 4.4: 3D-Renderings der Platinen

Um den Ultraschall-Transducer vor einem Fahrzeug-Sensor zu platzieren wurde außerdem ein Gehäuse ausgearbeitet. Zur Vermeidung von Reflexionen wird das Gehäuse mit Schaumstoff ausgefüllt. Der Transducer wird durch die Rückseite in das Gehäuse geführt und vor dem Fahrzeug-Sensor fixiert. Der Sensor-Knoten kann dabei an das Gehäuse festgemacht werden. Die Frontöffnung hat dabei eine Abmessung von 59x27 mm und ist

in Abbildung 4.5 (S. 37) dargestellt. Es wurde mittels einem 3D-Drucker gedruckt.

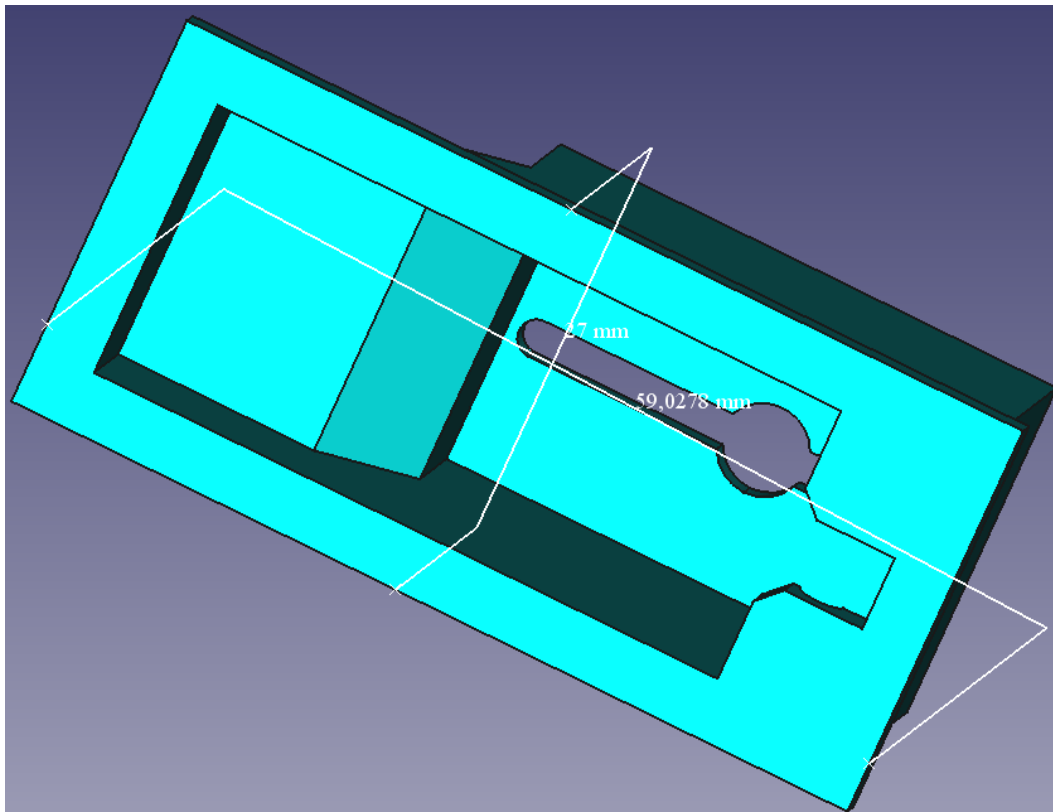


Abbildung 4.5: 3D-Rendering Gehäuse

4.2 Signal-Detektion und -Verzögerung

Wird der Ausgang des Analog-Komparators am Mikrocontroller durch ein anliegendes Signal gesetzt, so wird ein Capture-Interrupt am Timer1 ausgelöst. Timer1 ist mit einer Auflösung von 500 ns konfiguriert. Im Capture-Interrupt wird die Frequenz des Eingangssignals gemessen und nach fünf erfolgten Messungen von 40 kHz die eigentliche Verzögerung ausgelöst. Es

wurden fünf Messungen gewählt, um dem Transducer Zeit zum Einschwingen zu geben.

Der 16-bit Timer3 wird verwendet, um die vorgegebene Verzögerung zu realisieren. Hierzu wird der Timer im *Clear Timer on Compare match*-Modus konfiguriert. Die Verzögerung wird in Timer-Ticks umgerechnet und in das *Output Compare Register* geschrieben.

Timer3 läuft dabei mit einer Auflösung von $64 \mu\text{s}$ (Clock Prescaler 1024). Die Wahl dieser Zeitbasis ergibt sich aus der notwendigen minimalen und maximalen Verzögerung bei der minimalen und maximalen Messdistanz der Parkassistentz (0,3 m bzw. 1,7 m). Bei einer Schallgeschwindigkeit von $340 \frac{\text{m}}{\text{s}}$ ergibt sich hierfür eine Verzögerung von $t = \frac{2*s}{c} = \frac{2*0,3}{340} = 1,76\text{ms}$ bzw. $t = \frac{2*s}{c} = \frac{2*1,7}{340} = 10\text{ms}$. Durch die gewählte Zeitbasis kann diese Verzögerung mit Timer-Counter-Werte von $T = \frac{1760\mu\text{s}}{64\mu\text{s}} \approx 27$ bis $T = \frac{10000\mu\text{s}}{64\mu\text{s}} \approx 156$ realisiert werden.

Die Parkassistentz hat eine räumliche Auflösung von 0,1 m. Dadurch ergibt sich eine zeitliche Auflösung von $t = \frac{2*s}{c} = \frac{2*0,1}{340} = 588\mu\text{s}$. Mit der Zeitbasis von $64 \mu\text{s}$ wird diese Auflösung erreicht.

Erreicht nun der Timer-Counter den Wert im Compare-Register, wird der Output-Compare-Interrupt ausgelöst. Im Interrupt wird die Signal-Erzeugung durch eine software-getriggerte Schleife generiert. Dabei wird ein 40 kHz-Rechtecksignal generiert, dass über zwei GPIO-Pins (je ein Pin immer *HIGH* bzw. *LOW*) an den Transducer ausgegeben wird. Das Signal wird für $500 \mu\text{s}$ ausgegeben.

Die Umrechnung einer Distanz in Timer-Ticks erfolgt über die Formel $Ticks = \frac{2*s}{64*10^{-6}*c}$. s bezeichnet dabei die gewünschte Distanz in m und c die Schallgeschwindigkeit in $\frac{\text{m}}{\text{s}}$.

Abbildung 4.6 (S. 39) stellt diesen Ablauf in einem Flowchart grafisch dar.

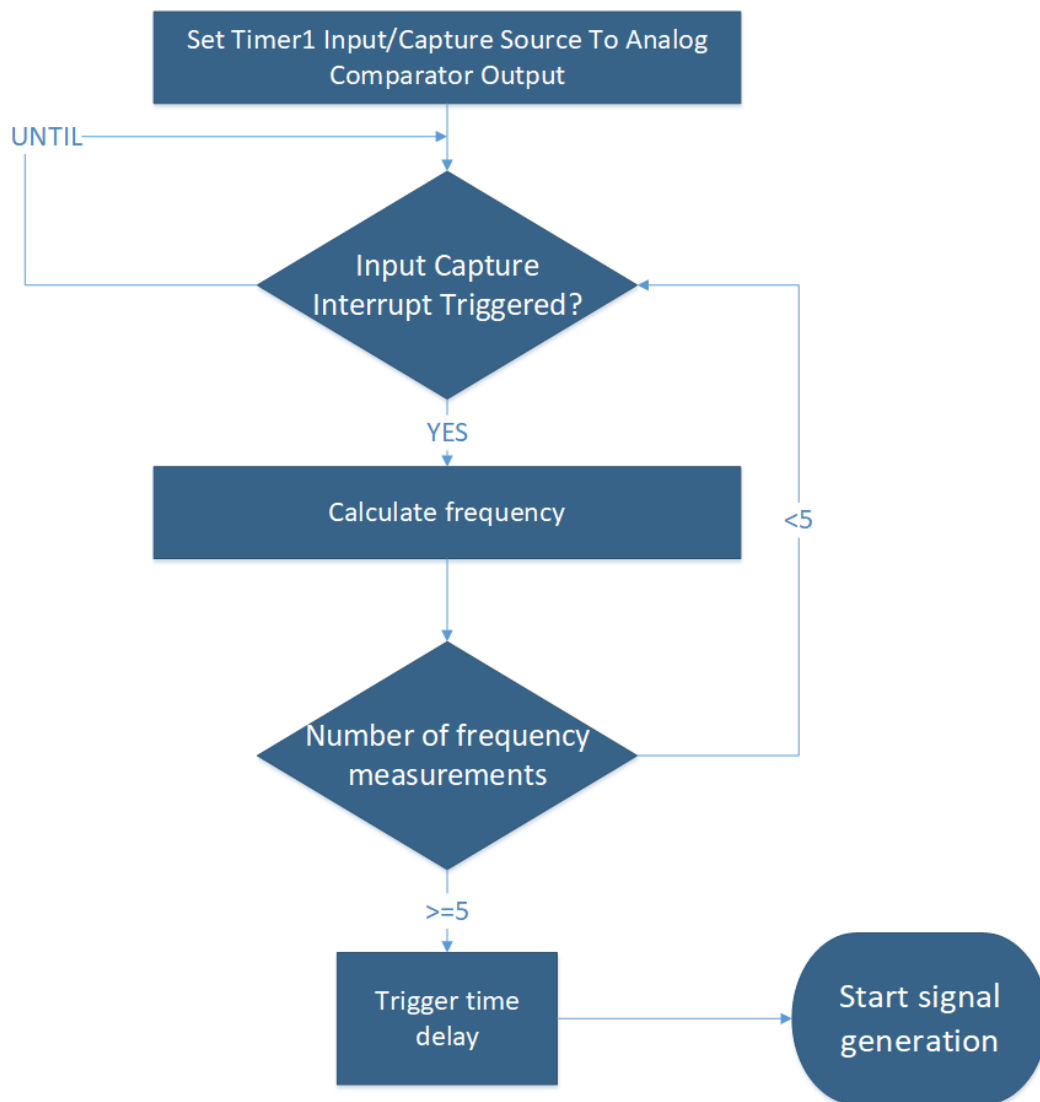


Abbildung 4.6: Flowchart Signal-Detektion und -Verzögerung

4.3 Kommunikation

4.3.1 Kommunikation zwischen den Knoten

Der Master-Knoten ist mit dem ersten Sensor-Knoten verbunden. Durch die Verkabelung in der *Daisy Chain* ist der physikalische Vorgänger und Nachfolger gleichzeitig der logische Vorgänger und Nachfolger eines Knotens. Falls ein Knoten keinen Nachfolger hat, ist dieser Knoten der Endknoten. Die *Daisy Chain* benötigt keinen speziellen Abschluss oder eine Rückverbindung zum Master-Knoten. Diese Struktur ist in Abbildung 3.8 (S. 27) ersichtlich.

Aus der physikalischen Reihenfolge der Knoten ergibt sich eine logische Nummerierung der Sensoren ausgehend vom Master-Knoten. Der Master-Knoten muss daher bei der Initialisierung die Sensor-Knoten abzählen und jedem von ihnen eine eindeutige ID zuweisen. Dies geschieht direkt nach dem Einschalten im Discovery-Verfahren.

Sobald alle Knoten eine ID besitzen, kann der Master-Knoten an die Sensoren Befehle schicken. Hierbei wurde ein *Request < - > Response-Modell* gewählt, bei dem die Sensor-Knoten nur dann eine Nachricht senden, wenn diese vom Master zuvor einen entsprechenden Befehl erhalten haben. Dies wurde aus praktischen Überlegungen hinsichtlich des verwendeten Protokolls festgelegt. Technisch wäre es auch möglich, dass die Sensor-Knoten untereinander kommunizieren oder ungefragt Nachrichten an den Master-Knoten senden.

Die Kommunikation erfolgt mit einer Baud-Rate von 115200 bps. Die verwendeten RS-485-Treiber sind Halb-Duplex-Treiber. Sie können also entweder Empfangen oder Senden. Das Umschalten des Treibers zwischen Senden und Empfangen erfolgt über die Firmware des Mikrocontroller.

Discovery

Unmittelbar nach dem Einschalten befinden sich alle Sensor-Knoten in einem Ruhezustand und besitzen die vorläufige ID 0. Im Ruhezustand

werden keine Nachrichten vom Master-Knoten an weitere angeschlossene Knoten weitergesendet.

Der Master-Knoten startet das Discovery-Verfahren beim Empfang der Nachricht mit der ID 0xA über den CAN-Bus. Es wird ein Timeout-Counter gestartet und zunächst eine Count-Nachricht mit dem Inhalt (je ein Byte) C 0 0 \n gesendet. Das zweite und dritte Byte bezeichnen hierbei die aktuelle Anzahl von bereits erkannten Knoten. In diesem Fall, bei der ersten Discovery-Nachricht, sind dies null Knoten. Der erste Sensor-Knoten empfängt diese Nachricht, sendet sie aber nicht weiter, da er immer noch die ID 0 besitzt. Nach Empfang dieser Nachricht übernimmt der erste Knoten die ID 1 und sendet eine Bestätigung an den Master-Knoten mit dem Inhalt N 1 \n . Das zweite Byte kennzeichnet hierbei die vom Knoten übernommene ID. Der Master-Knoten erhöht nun den internen Knoten-Zähler und sendet eine weitere Count-Nachricht. Falls innerhalb des Timeouts keine Antwort-Nachricht empfangen wird, bricht der Master-Knoten das Discovery ab und überträgt die Anzahl der erkannten Knoten über den CAN-Bus an die virtuelle Umgebung mit einer CAN-Nachricht mit der ID 0x701.

In Abbildung 4.7 (S. 42) ist das Discovery-Verfahren als Flowchart dargestellt.

Nach erfolgtem Discovery sind alle Knoten abgezählt und damit bereit, eine Distanz-Information übermittelt zu bekommen und die Stimulation zu beginnen.

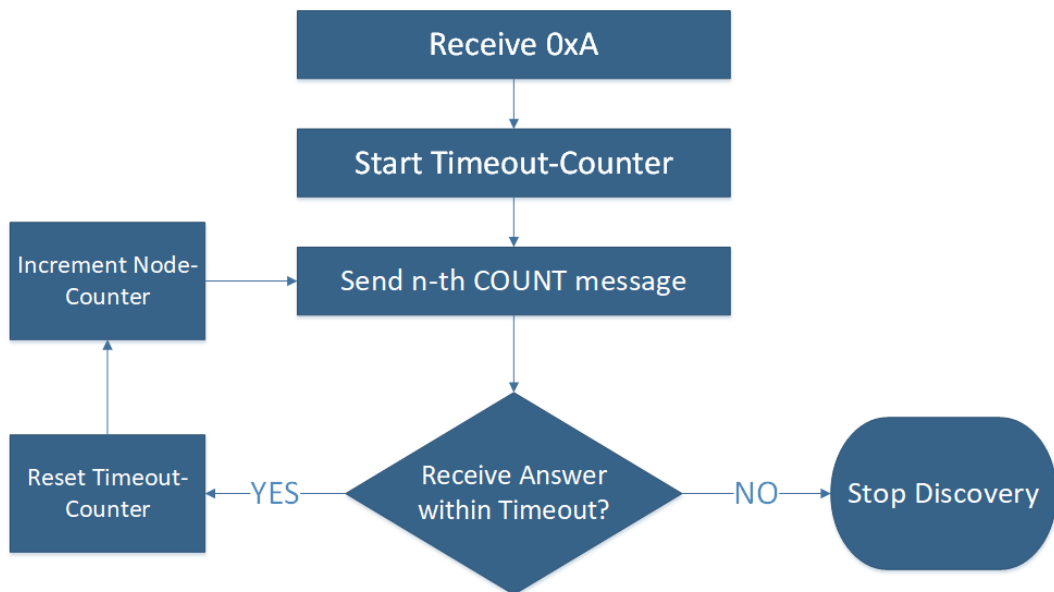


Abbildung 4.7: Flowchart Discovery-Verfahren

Nachrichten

Alle Nachrichten werden im ASCII-Format übertragen. Dabei identifiziert das erste übertragene Byte immer den Befehl. Anschließend werden, je nach Befehl, mehrere Befehls-spezifische Datenbytes übertragen. Zusätzlich wird noch eine 8-Bit CRC-Prüfsumme berechnet und gesendet. Ein Newline (ASCII-Code 10) kennzeichnet das Ende einer Nachricht.

Eine Nachricht wird nur dann behandelt, wenn die berechnete CRC-Prüfsumme mit der Übertragenen übereinstimmt. Zur Berechnung der Prüfsumme wird dabei das Polynom $x^8 + x^2 + x + 1$ verwendet. Die Implementierung nutzt eine speziell für AVR-Mikrocontroller optimierte Berechnungsroutine in der Standardimplementierung der libc von Atmel (Microchip, 2018). Falls die übertragene CRC-Prüfsumme nicht mit der berechneten Prüfsumme übereinstimmt, so wird eine Error-Nachricht an den Master-Knoten gesendet.

Nach jedem Befehl vom Masterknoten wird eine Rückantwort des jeweiligen Sensorknotens erwartet. Falls diese Antwort nicht innerhalb eines definierten Timeouts empfangen wird, so setzt der Masterknoten ein Error Flag im

Statusbyte. Dieses Statusbyte wird anschließend per CAN übertragen.

Tabelle 4.1 fasst die Nachrichten zusammen.

Befehl	Beschreibung
C	Spezielle Discovery/Zählnachricht
D	Setze Distanz für Sensorknoten
E	Error-Nachricht

Tabelle 4.1: Befehle

Bei der Distanz-Nachricht werden zwei Bytes zum Übertragen der Distanz verwendet. Das erste Byte kodiert die Zahl vor dem Komma und das zweite Byte die Zahl nach dem Komma.

4.3.2 CAN-Kommunikation

Die CAN-Schnittstelle ist mit einer Baud-Rate von 250 kbit/s konfiguriert. Die CAN-Kommunikation erfolgt zwischen virtueller Umgebung/Integrationstool und dem Master-Knoten.

Tabelle 4.2 zeigt die festgelegten CAN-Messages und den Inhalt der Datenbytes.

ID	Anzahl Datenbytes	Bedeutung Datenbytes
0xA	0	Start Discovery
0x14	1	Heartbeat Request
0x15	1	Heartbeat Answer
0x1E	2	NodeID Distance
0x28	1	NodeID
0x701	1	Count

Tabelle 4.2: Beschreibung CAN-Bus

Beim Empfang der Nachricht mit der ID 0xA (dez. 10) startet der Master-Knoten den Discovery-Prozess (siehe Kapitel 4.3.1). Nach dem Discovery-Prozess überträgt der Master-Knoten die Anzahl der erkannten Knoten mit einer Nachricht mit der ID 0x701 (dez. 1793).

Mit der Nachricht 0x14 (dez. 20) wird ein *Heartbeat Request* an den Master-Knoten gestellt. Der Master-Knoten antwortet daraufhin mit einer Nachricht mit der ID 0x15 (dez. 21, *Heartbeat Answer*). Das übertragene Datenbyte wird in der Antwort inkrementiert.

Über eine Nachricht mit der ID 0x1E (dez. 30) wird eine neue Distanz von der virtuellen Umgebung für einen bestimmten Sensor-Knoten übermittelt. Die beiden Bytes der Nachricht kodieren dabei den Abstand. Das erste Byte kodiert die Zahl vor dem Komma und das zweite Byte die Zahl nach dem Komma.

Die Nachricht mit der ID 0x28 (dez. 40) informiert die Umgebung über eine empfangene Error-Nachricht von einem Knoten.

5 Evaluierung und Integration der Ultraschall-Stimulation

Dieses Kapitel zeigt Messergebnisse der entwickelten Ultraschall-Stimulation an der Parkassistentz Valeo Beep&Park. Weiters wird eine konkrete VEHIL-Umgebung in Form des *AVL DrivingCube* vorgestellt, der Ultraschall-Stimulator in diese Umgebung integriert und weitere Messergebnisse gezeigt.

5.1 Räumliche Auflösung

Die Parkassistentz hat eine räumliche Auflösung von 0,1 m, was einer zeitliche Auflösung von $588 \mu\text{s}$ bedeutet (siehe Kapitel 4.2 auf S. 37). Verzögerungen bei der Verarbeitung durch den Stimulator von der Signaldetektion bis zur Signalgenerierung, dürfen also nicht größer als $588 \mu\text{s}$ sein, damit die räumliche Auflösung von 0,1 m realisiert werden kann.

Der Zusammenhang zwischen räumlicher Auflösung und eingestellter Verzögerung wurde auch gemessen. Die Verzögerung wird durch den Timer in diskreten Schritten von $64 \mu\text{s}$ eingestellt. Es wurde daher gemessen, bei welcher Anzahl an Timer-Ticks die Anzeige der Parkassistentz zur nächsten Distanz überspringt. Tabelle 5.1 (S. 46) zeigt das Intervall der Timer-Ticks, in welchem eine Distanz an der Parkassistentz angezeigt wurde. Außerdem wird das Intervall der tatsächlich eingestellten Verzögerungen anhand der Timer-Ticks berechnet.

Timer-Ticks	Anzeige m	Timer-Verzögerung ms
bis 20	P	bis 1,18
21 - 29	0,3	1,34 - 1,85
30 - 38	0,4	1,92 - 2,43
39 - 47	0,5	2,50 - 3,00
48 - 56	0,6	3,07 - 3,58
57 - 65	0,7	3,65 - 4,16
66 - 73	0,8	4,22 - 4,67
74 - 83	0,9	4,73 - 5,31
84 - 92	1,0	5,38 - 5,89
93 - 101	1,1	5,95 - 6,46
102 - 110	1,2	6,53 - 7,04
111 - 119	1,3	7,10 - 7,62
120 - 128	1,4	7,68 - 8,19
129 - 137	1,5	8,26 - 8,77
138 - 146	1,6	8,83 - 9,34
147 - 155	1,7	9,40 - 9,92
über 156	-	über 9,98

Tabelle 5.1: Messergebnisse Timer-Ticks

Aus der Tabelle ist ersichtlich, dass die Anzeige alle 8 Timer-Ticks überspringt. Dies entspricht einer Auflösung von $8 * 64\mu s = 512\mu s$. Die räumliche Auflösung, wieder bei einer Schallgeschwindigkeit von $340 \frac{m}{s}$, beträgt damit $s = \frac{512\mu s * 340}{2} = 0,09m$. Der Stimulator erreicht damit also die räumliche Auflösung der Parkassistentz und kann auch für Parkassistentzen mit einer besseren räumlichen Auflösung als 0,1 m eingesetzt werden.

5.2 Updaterate

Wie in Abbildung 3.5 (S. 21) ersichtlich, sendet die Parkassistentz kontinuierlich alle 180 ms bzw. 100 ms einen Sendeimpuls aus.

Es wird davon ausgegangen, dass es möglich sein soll, bei jedem Messimpuls der Parkassistentz bereits einen neuen Wert von der virtuellen

Umgebung verfügbar zu haben. Dazu darf die Übertragung eines Wertes von der virtuellen Umgebung über den Master-Knoten bis zu einem Stimulator-Knoten nicht länger als 100 ms dauern. Bei der verwendeten Baud-Rate von 115200 bps und einem Overhead von zwei Byte (Newline + CRC-Prüfsumme) ergibt sich eine gemessene Zeitdauer von $521,2 \mu s$ für Übertragung von einer Distanz-Nachricht. Da die Nachrichten innerhalb der Daisy-Chain weitergeschleift werden, ergibt sich eine Maximalanzahl von $n = \frac{100ms}{0,521ms} \approx 191$ Knoten, die mit diesem System innerhalb der 100 ms angesteuert werden können.

Dies ist weit oberhalb jener Zahl an Ultraschall-Sensoren, die üblicherweise in Fahrzeugen verbaut ist.

5.3 Stimulationsdistanz

Zur Beurteilung der Stimulationsdistanz wurden zwei Messungen durchgeführt: Eine Messung, bis zu welchem Abstand der Stimulator das Signal vom Fahrzeugsensor noch erkennt und die Verzögerung triggert. Eine zweite Messung wurde durchgeführt, bis zu welcher Abstand die Parkassistentz noch eine eingestellte Distanz anzeigt.

Bei der ersten Messung wurde ein Maximalabstand von 30 cm ermittelt.

Die zweite Messung ergab einen Maximalabstand von 1,5 cm.

5.4 Stromverbrauch

Während der Stimulation wurde an einem Sensorknoten ein Stromverbrauch von 39,59 mA gemessen. Die aufgenommene Leistung beträgt $P = U * I = 12V * 39,49mA = 0,475W$. Die Verlustleistung über den Spannungsregler ist $P = (U_E - U_A) * I = (12V - 5V) * 39,49mA = 0,27W$.

Im Datenblatt des verwendeten LM7805 im TO-263 Package ist die maximale Verlustleistung für bestimmte Umgebungstemperaturen angegeben (Texas Instruments, 2016, S. 19). Oberhalb dieser angegebenen Verlustleistung muss

aufgrund der Temperaturentwicklung ein Kühlkörper verwendet werden. Die Verlustleistung liegt unterhalb der maximalen Verlustleistung. Der Stimulator ist daher auch für den Dauerbetrieb ohne Kühlung verwendbar.

5.5 Integration am AVL DrivingCube

Der AVL DrivingCube ist eine *vehicle hardware-in-the-loop*-Simulation, mit der man Fahrerassistenzfunktionen im Kontext des vollständig integrierten Gesamtfahrzeuges auf einem Fahrzeugprüfstand testen kann. Das reale Fahrzeug befindet sich dabei auf einem klassischen Automobil-Prüfstand und ist an eine Simulation einer virtuellen Umgebung mit anderen Verkehrsteilnehmern angebunden. Die Fahrt des Fahrzeuges am Prüfstand ist dabei direkt an die Simulation gekoppelt. Damit können verschiedene Fahrscenarien durchgeführt werden und die simulierten Verkehrsteilnehmer der virtuellen Umgebung dienen als Input für die Sensor-Stimulatoren des Fahrzeuges.

Zur Demonstration, dass sich der Ultraschall-Stimulator für eine Integration am DrivingCube eignet, wurde zunächst ein passendes Simulationsmodell mit *Model.Connect* erstellt, anschließend ein Fahrscenario und ein virtueller Sensor definiert und schließlich der Stimulator an die Simulation gekoppelt. Dann wurde wieder die Parkassistenz Valeo Beep&Park stimuliert. Es wurde gemessen, ob die Ausgabe des virtuellen Sensors mit der Anzeige an der Parkassistenz übereinstimmt.

Der DrivingCube besteht dabei aus folgenden Komponenten:

- Prüfstand: Der Prüfstand des DrivingCube besteht aus einem Rollen- oder Powertrain-Prüfstand. Der Prüfstand misst die aktuelle Drehzahl an den vier Reifen/Halbachsen und setzt die Straßenlast.
- Umgebungssimulation und Fahrscenarien: Die virtuelle Umgebung wird durch die Simulation *Virtual Test Drive (VTD)* des Herstellers Vires realisiert (Vires, 2018). In dieser Umgebung können auch die Fahrscenarien definiert werden.

- Fahrzeugsimulation: Mit der Software *AVL VSM* (AVL, 2018) werden aus den Drehzahlen des Prüfstandes alle fahrdynamischen Parameter (z.B. Beschleunigungen) des Fahrzeuges berechnet.
- Integrations-Tool: Die Co-Simulations Plattform *AVL Model.Connect* wird zum Verbinden der Simulation mit dem Fahrzeug am Prüfstand und der Integration der Sensoren verwendet.

Der grundlegende Aufbau des DrivingCube ist dabei in Abbildung 5.1 ersichtlich. Neben den bereits erwähnten Komponenten, besitzt der Driving-Cube auch die Möglichkeit des *environment conditioning*. Damit können am Prüfstand verschiedenste Umgebungsbedingungen (Temperatur, Luftfeuchtigkeit...) reguliert werden.

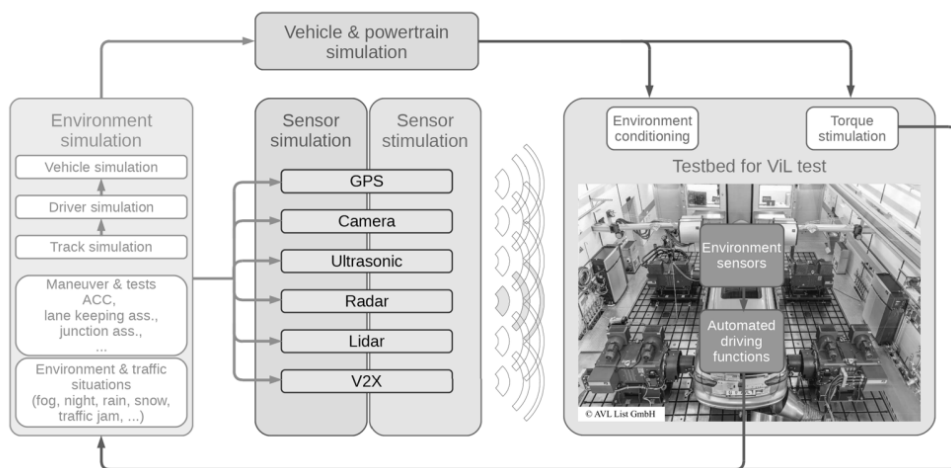


Abbildung 5.1: AVL DrivingCube Aufbau (aus Gadringer u. a., 2018)

Der Ultraschall-Stimulator ist ebenfalls ein Teil des DrivingCube.

5.5.1 Simulationsmodell

Die Simulationsmodelle wurden in *Model.Connect* erstellt. *Model.Connect* ist eine Plattform, um reale Komponenten zusammen mit virtuellen Modellen in Echtzeit zu co-simulieren. Die Topologie der Co-Simulation kann dabei

über einen grafischen Editor definiert werden. Als Topologie werden dabei die als Blöcke abstrahierten Komponenten der Simulation und deren Signalverbindungen untereinander bezeichnet. Es lassen sich auch Variationen der Parameter in der Umgebungssimulation, den Fahrscenarien und der Fahrzeugsimulation durchführen.

Das erste Modell (Abbildung 5.2) dient für Testzwecke unter Labor-Bedingungen. Das Fahrzeug wird dabei vollständig von *VSM* simuliert, es entfällt der Prüfstand. In der Umgebungssimulation wird ein Parkszenario simuliert und dabei Gas- und Bremspedalstellung sowie der Lenkwinkel an *VSM* übertragen. Von einem virtuellen Sensor wird die aktuelle Distanzinformation über einen CAN-Block an den Ultraschall-Stimulator gesendet.

Die Ultraschall-Stimulation stimuliert dann einen realen Fahrzeugsensor. Die Distanzinformation aus der virtuellen Umgebung kann dann an der Anzeige des Fahrzeugsensors abgelesen werden.

Das zweite Modell (Abbildung 5.3) zeigt den Aufbau einer VEHIL-Simulation mit der Integration eines realen Prüfstands. In diesem Modell dienen die Raddrehzahlen des Prüfstandes als Input für *VSM*. Die Umgebungssimulation simuliert weiterhin ein Parkszenario und gibt den Lenkwinkel vor.

In *VSM* wird die Fahrdynamik des Fahrzeuges simuliert. Das Fahrzeugmodell muss mit dem realen Fahrzeug möglichst übereinstimmen. Da der reale Prüfstand die Drehzahl der vier Halbachsen misst, muss für die korrekte Berechnung der Fahrzeuggeschwindigkeit die Reifendimension korrekt gesetzt sein. Für die Simulation ohne Prüfstand wird der Fahrer in der virtuellen Umgebung simuliert. Dieser gibt entweder eine Geschwindigkeit v_{soll} vor, welche die Fahrdynamik dann in eine Gas- und Bremspedalstellung übersetzt oder der virtuelle Fahrer gibt direkt diese Stellung vor. *VSM* berechnet daraufhin die Reaktion des Fahrzeuges auf den Input und führt diese Reaktion in Form einer Bewegungsänderung zurück an die virtuelle Umgebung. Für die Simulation am Prüfstand und damit einem realen Fahrzeug besteht der Input für *VSM* aus den gemessenen Raddrehzahlen. Diese Drehzahlen werden von *VSM* integriert, wiederum in eine Bewegungsänderung überführt und in die virtuelle Umgebung geführt.

Die Querdynamik wird durch den Lenkradwinkel des virtuellen Fahrers vorgegeben. Dieser Lenkradwinkel dient ebenfalls als Input für *VSM*.

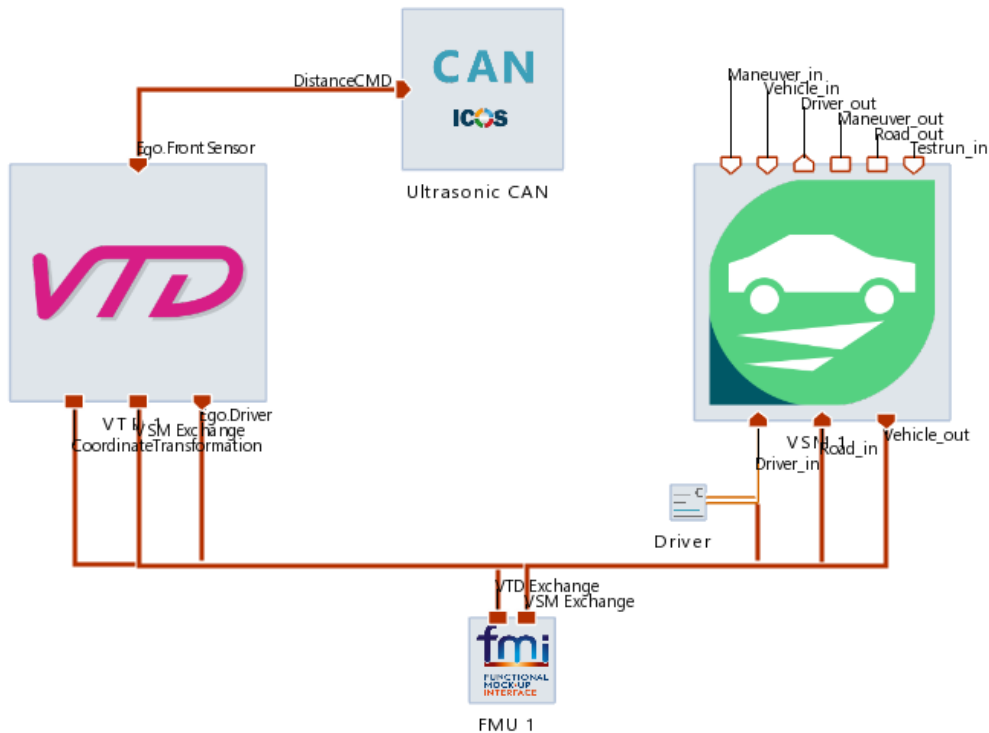


Abbildung 5.2: Topologie in Model.Connect mit Simulation aller Komponenten

Aus der virtuellen Umgebung *VTD* wird die Distanzinformation eines virtuellen Sensors an den Ultraschall-Stimulator übertragen. Der Stimulator stimuliert am Prüfstand einen oder mehrere der Ultraschall-Sensoren des Fahrzeuges.

5.5.2 Virtueller Sensor

Zur Ausgabe der Distanz wurde ein virtueller Sensor in *VTD* erstellt. Der Sensor wurde am Fahrzeug unterhalb des linken Scheinwerfers platziert. Grundsätzlich ist es so möglich, alle Ultraschall-Sensoren des Fahrzeuges

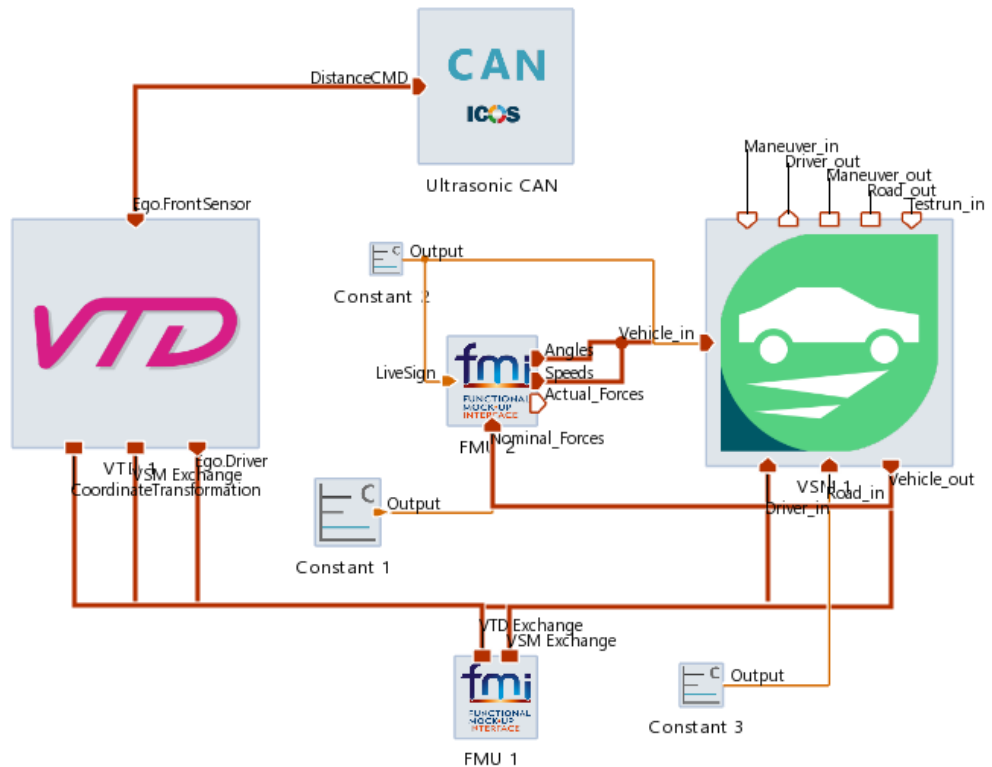


Abbildung 5.3: Topologie in Model.Connect mit Schnittstelle zum Prüfstand

zu modellieren. Aufgrund von Testzwecken wurde hier nur ein Sensor verwendet. In der Abbildung 5.4 ist ein Screenshot aus einer laufenden Simulation mit dem, zu Debug-Zwecken eingeblendeten, virtuellem Sensor zu sehen.

Der Sensor besitzt ein kegelförmiges *Field-of-View* von 10° in horizontaler und vertikaler Richtung. Er gibt die Distanz zu einem erkannten Fahrzeug bis zu einer Maximaldistanz von 3 m aus. Da keine Daten von realen Fahrzeug-Sensoren bekannt waren, mussten diese Parameter geschätzt werden.

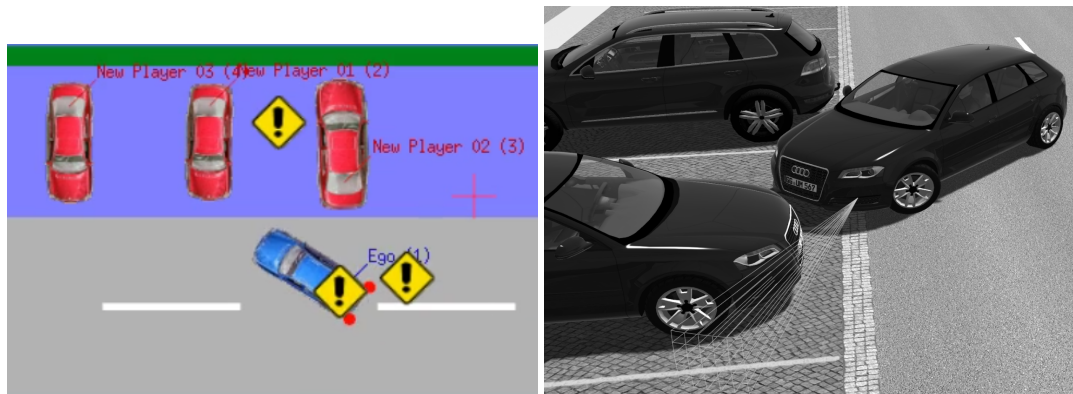


Abbildung 5.4: Frontansicht des virtuellen Sensors

5.5.3 Einpark-Szenario

Zum Testen wurde in der Umgebungssimulation *VTD* ein Einparkszenario erstellt. Das Szenario besteht aus einem Fahrzeug, welches sich einer Parklücke nähert. Beiderseits der Parkmöglichkeit befinden sich zwei parkende Autos. Das simulierte Fahrzeug parkt nun vorwärts in diese Parklücke ein. Während des vordefinierten Einparkmanövers wird die Distanz zum linken Fahrzeug über den virtuellen Sensor ausgegeben und an die Ultraschall-Stimulation weitergeleitet.

In der Abbildung 5.5a ist ein Screenshots des *Scenario Designers* von *VTD* zu sehen. Das simulierte Fahrzeug, *Ego-Fahrzeug* genannt, wird in blauer Farbe dargestellt, die parkenden Fahrzeuge in rot. Das Ausrufezeichen-Symbol zwischen den beiden parkenden Fahrzeugen markiert den Endpunkt des Parkmanövers. Ein weiterer Screenshot des Szenarios ist in Abbildung 5.5b



(a) Schematischer 2D-Überblick des Szenarios

(b) 3D-Visualisierung in VTD

Abbildung 5.5: Parkmanöver

ersichtlich. Hier führt das Ego-Fahrzeug das Parkmanöver gerade aus und im virtuellen Sensor-Kegel befindet sich bereits das parkende Fahrzeug.

Die Ausgabe der Distanz des virtuellen Sensors findet sich als Diagramm in Abbildung 5.6 (S. 55). Die Ausgabe steigt sprunghaft von Null an, sobald sich das parkende Fahrzeug innerhalb des definierten *Field-of-View* des Sensors befindet und springt wieder auf Null zurück, wenn sich das Fahrzeug außerhalb des *Field-of-Views* befindet. Zwischen den beiden Sprungstellen ist die Distanz zwischen dem Aufhängepunkt des virtuellen Sensors und dem Fahrzeug ersichtlich.

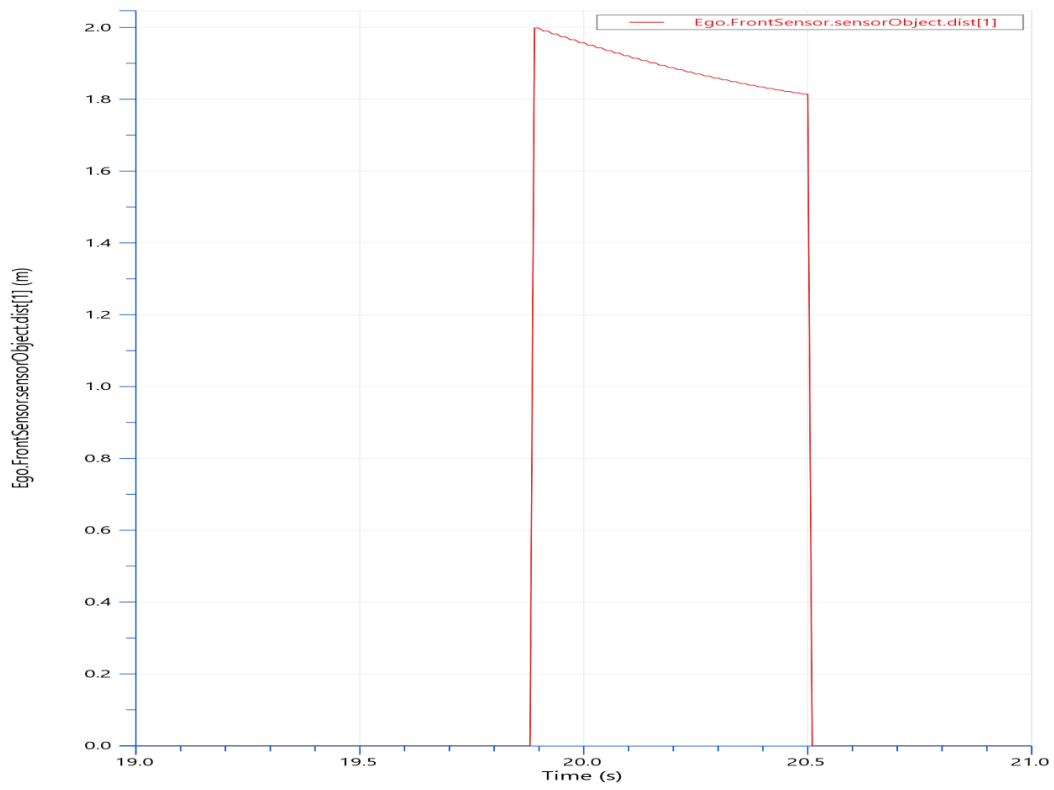


Abbildung 5.6: Schematischer 2D-Überblick des Szenarios

6 Diskussion

Im Rahmen dieser Arbeit wurde gezeigt, wie einer Ultraschall-basierten Einparkassistentz mit Hilfe einer Stimulations-Plattform ein Abbild einer virtuellen Umgebung vorgegeben werden kann. Die Ableitung der Anforderungen an eine Ultraschall-Stimulation wurde diskutiert und die Implementierung auf Hard- und Softwareebene vorgestellt. Schließlich wurden verschiedene Messergebnisse dargestellt. Anschließend wurde die konkrete *vehicle hardware-in-the-loop*-Simulation *DrivingCube* der Firma AVL List GmbH vorgestellt und die Integration der Stimulations-Plattform in diese Umgebung wurde gezeigt.

Der vorgestellte Stimulator muss im Rahmen der Applikation am *DrivingCube* noch weiteren Tests mit verschiedenen Fahrzeug-Sensoren unterzogen werden. Dadurch soll sichergestellt sein, dass die Stimulation auch mit einer Vielzahl an verschiedenen Sensoren anderer Hersteller kompatibel ist und noch vorhandene Fehler gefunden werden. Hierbei ist auch die Weiterentwicklung der Fehlerbehandlung erwähnenswert. Dadurch kann die einwandfreie Funktion und Integration des Stimulators beurteilt werden. Ebenso kann eine Adaptierung auf andere Sensor-Frequenzen behandelt werden.

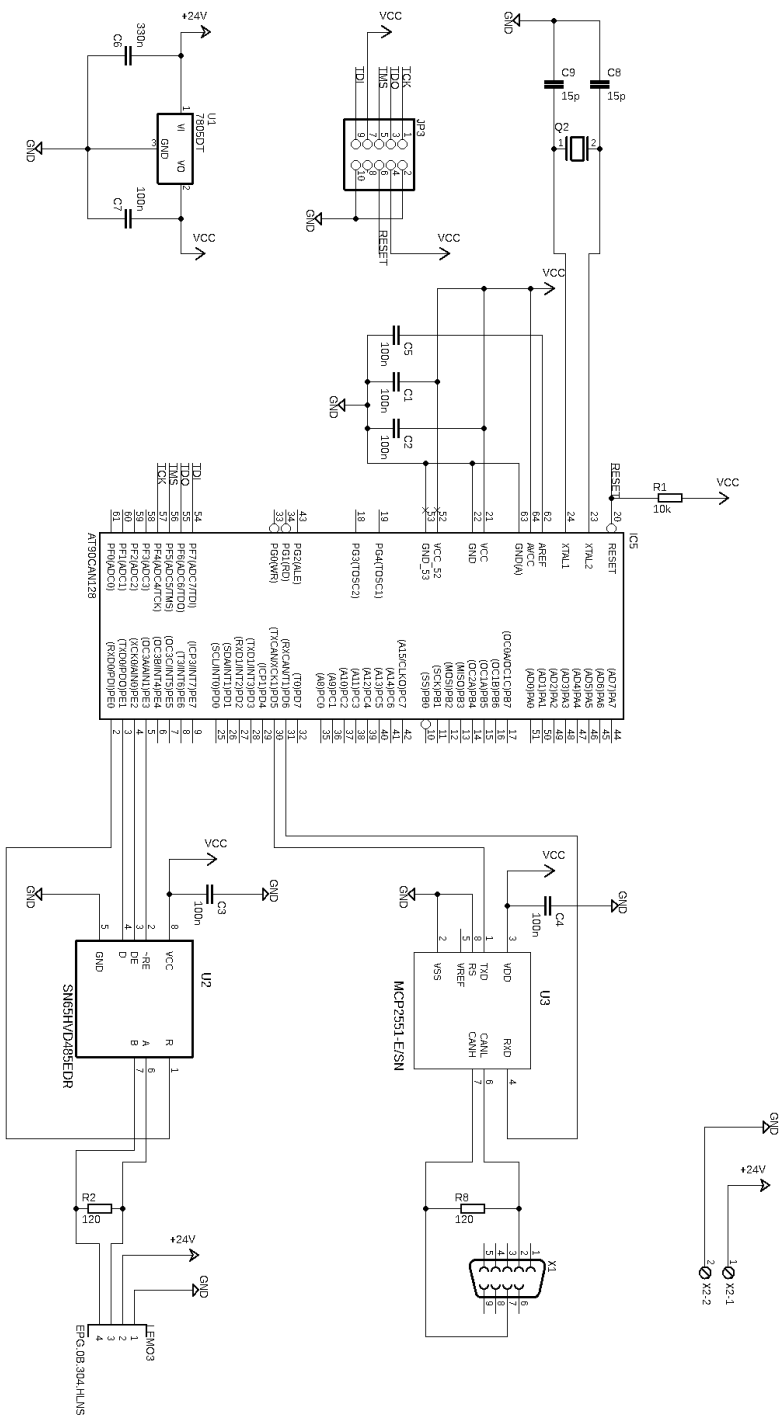
Die Stimulationsdistanz ist für vorgestellte Anwendung ausreichend. Eine geeignete Verstärkerschaltung würde diese Stimulationsdistanz jedoch noch erhöhen.

Eine Erweiterung des Stimulators hinsichtlich Trilateration und Kreuzecho wäre möglich, bedürfte jedoch größerer Änderung hinsichtlich Synchronisation und Zusammenspiel der einzelnen Sensor-Knoten.

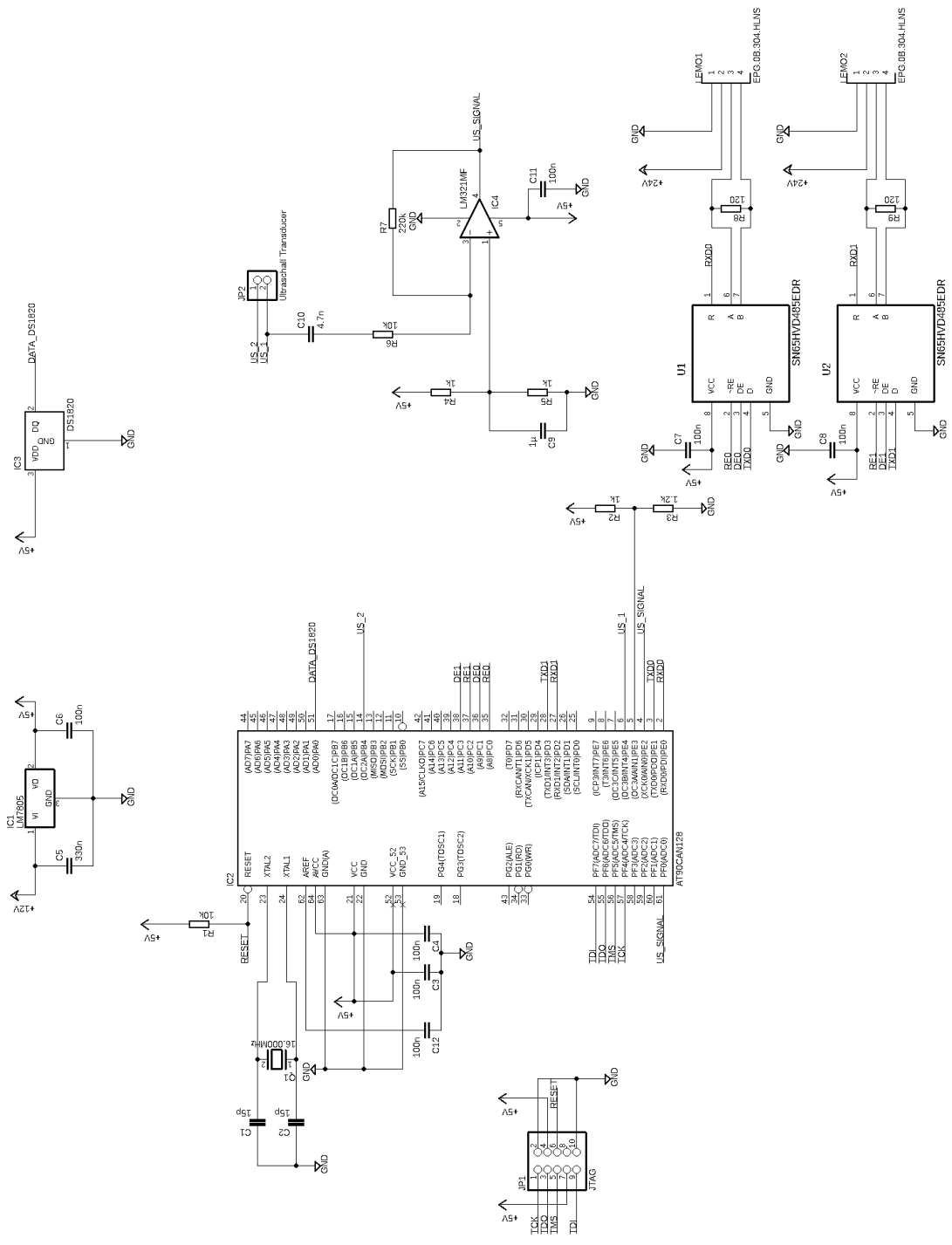
Die Evaluierung der Stimulation erfolgte bisher nur mit einer informierenden Einparkassistentz. In einer weiteren aufbauenden Arbeit könnte eine vollautomatische Einparkassistentz mit Hilfe einer *vehicle hardware-in-the-loop*-Umgebung getestet werden.

Die in Kapitel 1.1 beschriebenen Ziele dieser Arbeit wurden erreicht.

Appendix



Schaltplan Master-Knoten



Schaltplan Sensor-Knoten

Abbildungsverzeichnis

2.1	Building Blocks von Fahrerassistenzfunktion aus Watzenig u. a., 2017, S. 6	4
2.2	Validation Toolchain aus Watzenig u. a., 2017, S.444	10
3.1	Schematische Darstellung des Puls-/Laufzeitprinzips (Reif, 2016, S. 50)	12
3.2	Elektrisches Ersatzschaltbild piezoelektrischer Schallwandler (Winner u. a., 2015, S. 248)	14
3.3	Beispielhafte Schalldruckverteilung für einen Ultraschallwandler (Müller u. a., 2017, S. 13)	16
3.4	Beispielhafte Field-of-View-Messung für ein Ultraschallsystem mit vier Sensoren (Winner u. a., 2015, S. 256)	17
3.5	Zeitliche Abfolge der Ultraschall-Impulse des Valeo Beep&Park (50 ms/div, 10 V/div, 10x Tastkopf)	21
3.6	Valeo Beep&Park Messimpuls (100 μ s/div, 1 V/div, 10x Tastkopf)	22
3.7	Aufgenommener Impuls Ultraschall-Transducer <-> Valeo Beep&Park (200 μ s/div, 100 mV/div)	24
3.8	Aufbau des Ultraschall-Stimulators	27
3.9	Design des Master-Knotens	28
3.10	Design eines Sensor-Knotens	28
4.1	Beschaltung LM7805	31
4.2	Schaltplan Preconditioning	33
4.3	Signal nach Preconditioning (200 μ s/div, 1 V/div)	34
4.4	3D-Renderings der Platinen	36
4.5	3D-Rendering Gehäuse	37
4.6	Flowchart Signal-Detektion und -Verzögerung	39
4.7	Flowchart Discovery-Verfahren	42

5.1	AVL DrivingCube Aufbau (aus Gadringer u. a., 2018)	49
5.2	Topologie in Model.Connect mit Simulation aller Komponenten	51
5.3	Topologie in Model.Connect mit Schnittstelle zum Prüfstand	52
5.4	Frontansicht des virtuellen Sensors	53
5.5	Parkmanöver	54
5.6	Schematischer 2D-Überblick des Szenarios	55

Literatur

- ANALOG DEVICES, 2015. *CN-0343 - Ultrasonic Distance Measurement* [online] [besucht am 2019-01-03]. Abgerufen unter: <https://www.analog.com/en/design-center/reference-designs/hardware-reference-design/circuits-from-the-lab/cn0343.html> (siehe S. 16, 18).
- AT90CAN Datasheet, 2008. 7679H (siehe S. 29).
- AVL, 2018. *VSM* [online] [besucht am 2018-08-13]. Abgerufen unter: <https://www.avl.com/-/avl-vsm-4-/> (siehe S. 49).
- BUNDESANSTALT FÜR STRASSENWESEN, 2012. *Rechtsfolgen zunehmender Fahrzeugsautomatisierung*. Gemeinsamer Schlussbericht der Projektgruppe, Heft F 83 (siehe S. 4).
- CYPRESS, 2017. *AN-76530 - PsoC® 1 Automotive Ultrasonic Distance Measurement For Park-Assist Systems* [online] [besucht am 2019-01-03]. Abgerufen unter: <https://www.cypress.com/documentation/application-notes/an76530-psoc-1-automotive-ultrasonic-distance-measurement-park> (siehe S. 18).
- GADRINGER, Michael E.; SCHREIBER, Helmut; GRUBER, Andreas; VORDERDERFLER, Michael; AMSCHL, Dominik; BÖSCH, Wolfgang; METZNER, Steffen; PFLÜGL, Horst; PAULWEBER, Michael, 2018. *Virtual reality for automotive radars*. *Elektrotech. Inftech.* (2018) 135: 335. <https://doi.org/10.1007/s00502-018-0620-9> (siehe S. 11, 49).
- GOCK, Andy, 2017. *avr-uart*. Auch verfügbar unter: <https://github.com/andygock/avr-uart> (siehe S. 30).
- GREIF, Fabian, 2016. *avr-can-lib*. Auch verfügbar unter: <https://github.com/dergraaf/avr-can-lib> (siehe S. 30).

- HÄLKER, Jens; BARTH, Harald, 2018. *Lidar als Schlüsseltechnik für das automatisierte und autonome Fahren*. H. ATZ Automobiltech Z (2018) 120(Suppl 3): 70. <https://doi.org/10.1007/s35148-018-0108-2> (siehe S. 6).
- LEMO, 2018. *Lemo Connectors* [online] [besucht am 2018-08-12]. Abgerufen unter: <https://www.lemo.com/> (siehe S. 35).
- MICROCHIP, 2018. *AVR Libc Reference Manual*. Auch verfügbar unter: https://www.microchip.com/webdoc/AVRLibcReferenceManual/group__util__crc_1gab27eaaef6d7fd096bd7d57bf3f9ba083.html (siehe S. 42).
- MÜLLER, Gerhard; MÖSER, Michael, 2017. *Ultraschall in Medizin und Technik*. Springer Vieweg (siehe S. 14, 16).
- MULTICOMP, 2013. *MCUSD14A40S09RS-30C Datasheet*. Auch verfügbar unter: http://www.farnell.com/datasheets/1759991.pdf?_ga=2.39006371.1211542189.1528980736-704578842.1524133720 (siehe S. 23, 34).
- NIEDRIG, H.; STERNBERG, M., 2014. *Das Ingenieurwissen Physik*. Springer Vieweg (siehe S. 16).
- NXP, 2014. *AN-4841 - S12ZVL LIN Enabled Ultrasonic Distance Measurement* [online] [besucht am 2019-01-03]. Abgerufen unter: <https://www.nxp.com/docs/en/application-note/AN4841.pdf> (siehe S. 18).
- REIF, Konrad, 2016. *Sensoren im Kraftfahrzeug*. 3. Auflage. Springer Vieweg (siehe S. 6, 12, 13).
- TEXAS INSTRUMENTS, 2014. *LM321 Datasheet*. Auch verfügbar unter: <http://www.ti.com/lit/ds/symlink/lm321.pdf> (siehe S. 32).
- TEXAS INSTRUMENTS, 2015. *TIDUAI8A - Ultrasonic Distance Measurement BoosterPack*. Auch verfügbar unter: <https://www.ti.com/lit/ug/tiduai8a/tiduai8a.pdf> (siehe S. 18).
- TEXAS INSTRUMENTS, 2016. *LM7805 Datasheet*. Auch verfügbar unter: <http://www.ti.com/lit/ds/symlink/lm7800.pdf> (siehe S. 30, 47).
- VALEO, 2018. *Valeo Beep & Park* [online] [besucht am 2018-12-23]. Abgerufen unter: <https://www.valeoservice.de/de/pkw/fahr-und-parkassistenzsysteme-pkw/valeo-beeparktm-4x-sensoren-1-lautsprecher> (siehe S. 20, 22).

- VIRES, 2018. *Virtual Test Drive* [online] [besucht am 2018-08-13]. Abgerufen unter: <https://vires.com/vtd-vires-virtual-test-drive/> (siehe S. 48).
- WATZENIG, Daniel; HORN, Martin, 2017. *Automated Driving*. Springer (siehe S. 4, 9, 10).
- WINNER, Hermann; HAKULI, Stephan; LOTZ, Felix; SINGER, Christina, 2015. *Handbuch Fahrassistenzsysteme*. Dritte Auflage. Springer Vieweg (siehe S. 3, 5–9, 13–18).