



Entwicklung eines Modellfahrzeugs mit Fahrerassistenzfunktionen

Studienarbeit T3_3100

für die Prüfung zum

Bachelor of Science

des Studiengangs Informationstechnik

an der Dualen Hochschule Baden-Württemberg Karlsruhe

von

Leander Gantert & Luca Müller

31. August 2026

Bearbeitungszeitraum
Matrikelnummer, Kurs
Betreuer der DHBW

XX.XX.2025 - XX.XX.2026
3854248, TINF23B3
Dipl.-Ing. (FH) Stefan Lehmann

Zusammenfassung

Das Thema dieser Bachelorarbeit ist die Entwicklung eines Prototyps für eine neue Softwarearchitektur. Diese Architektur basiert auf dem LIN-Bussystem und soll die Kommunikation zwischen Slaves und Mastern überarbeiten. Ziel dieser Arbeit ist es, eine...

Diese Bachelorarbeit behandelt die Entwicklung eines elektronischen Systems zur Temperaturbegrenzung von Strahlungsheizkörpern mithilfe von Thermoelementen. Ziel dieser Arbeit ist es, ein kostengünstigeres und flexibleres System zu entwickeln, das die bislang verwendeten mechanischen Temperaturbegrenzer ersetzt. Hintergrund ist die zunehmende Marktanforderung nach sensorbasierten Lösungen sowie die Notwendigkeit, den Oil-Ignition-Test der Norm UL 858 zu bestehen, der zukünftig auch für Strahlungsheizkörper verpflichtend sein könnte.

Der Kern der Arbeit liegt in der Auswertung von Thermoelementen zur Temperaturbegrenzung der Strahlungsheizkörper. Die Arbeit umfasst die Entwicklung der hierfür notwendigen Hardware sowie die Programmierung der Software für den Mikrocontroller. Zusätzlich wird eine PC-Anwendung entwickelt, die den Systemstatus der entwickelten Steuerung visualisiert und eine Konfiguration der Temperaturbegrenzung ermöglicht.

Abschließend werden Funktionstests des entwickelten Systems durchgeführt, gefolgt von einer Bewertung der Wirtschaftlichkeit des neuen Systems.

Abstract

The scope of this bachelor thesis is the development of an electronic system for temperature limitation of radiant heating elements using thermocouples. The aim of this thesis is to develop a more cost-effective and flexible system to replace the mechanical temperature limiters currently in use. The motivation for this work is the increasing market demand for sensor-based solutions and the need to pass the Oil-Ignition Test of the UL 858 standard, which could also apply to radiant heaters in the future.

The focus of this thesis is measuring the voltage from thermocouples to limit the temperature of the radiant heating elements. Furthermore, the thesis includes the development of the necessary hardware and the development of the microcontroller software. Additionally, a desktop application is developed to visualize the system status of the developed control system and to configure the temperature limits.

Finally, functional tests of the developed system will be conducted, followed by an assessment of the economic efficiency of the new system.

Erklärung

Wir versichern hiermit, dass wir unsere Studienarbeit T3_3100 mit dem Thema: *Entwicklung eines Modellfahrzeugs mit Fahrerassistenzfunktionen* selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben.

Karlsruhe, 31. August 2026

A handwritten signature in black ink, appearing to read 'L. Müller'.

Leander Gantert & Luca Müller

Inhaltsverzeichnis

Abbildungsverzeichnis	III
Tabellenverzeichnis	III
Listings	III
Abkürzungsverzeichnis	IV
1 Einleitung	1
1.1 Motivation	1
1.2 Problemstellung	1
1.3 Ziel dieser Arbeit	2
2 Hardware	3
2.1 Ultraschallsensor	3
2.2 Raspberry Pi	3
2.2.1 Aufbau des Raspberry Pi	3
2.2.2 Raspberry Pi als Steuergerät	4
2.3 Motoren	4
2.4 Raspberry Pi Camera Module	4
2.5 Stromversorgung	4
2.6 Gehäuse	5
2.7 Aufbau des Fahrzeugs?	5
3 Softwarearchitektur	6
3.1 Übersicht und Aufbau der Architektur	6
3.2 Python	6
3.2.1 Programmiersprache Python	6
3.2.2 Verwendete Bibliotheken und Frameworks	6
3.3 Verwendete Technologien und Tools	6
3.3.1 IDEs	6
3.3.2 Versionsverwaltung mit Git	6
3.3.3 Protokolle	6

3.4	Softwarekomponenten und deren Kommunikation	7
3.4.1	GUI-Anwendung zur Steuerung	7
3.4.2	Backend-Anwendung zur Steuerung der Hardware	7
3.4.3	Kommunikation zwischen GUI und Backend über gRPC	7
3.5	Implementierte Features	7
3.5.1	Schilderkennung	7
3.5.2	Hinderniserkennung	7
3.5.3	Follow the line	7
3.5.4	Fernsteuerung	7
3.5.5	etc.	7
4	Implementierung und Integration	8
4.1	Hardware	8
4.2	Software	8
5	Testing und Qualitätssicherung	9
5.1	Teststrategie für die Überarbeitung	9
5.2	Definition der Testfälle	9
5.3	Testdurchführung zur Funktionalitätsprüfung	9
5.4	Evaluation der Testergebnisse	9
6	Fazit und Ausblick	10

Abbildungsverzeichnis

2.1	Funktionsweise Ultraschallsensor	3
-----	--	---

Tabellenverzeichnis

Listings

Abkürzungsverzeichnis

GPIO	General Purpose Input/Output
LAN	Local Area Network
USB	Universal Serial Bus
PCIe	Peripheral Component Interconnect Express

1 Einleitung

1.1 Motivation

In den letzten Jahren hat die Entwicklung von Fahrassistenzsystemen und autonomen Fahren erheblich zugenommen und ist mittlerweile ein zentrales Thema in der Automobilindustrie. Für viele Autofahrer sind diese Technologien bereits zu einem unverzichtbaren Bestandteil ihres Fahrerlebnisses geworden.

Fahrassistenzsysteme bieten eine Vielzahl an Vorteilen. Sie erhöhen die Sicherheit im Straßenverkehr, indem sie den Fahrer in kritischen Situationen unterstützen und Unfälle verhindern können. Zudem tragen sie zur Reduzierung der Belastung des Fahrers bei, da sie repetitive Aufgaben übernehmen können und so die Ermüdung verringern. Mithilfe von Sensoren und Kamerasystemen können Fahrassistenzsysteme die Umgebung des Fahrzeugs überwachen und auf potenzielle Gefahren reagieren. Dies ermöglicht es dem Fahrer, sich auf andere Aspekte des Fahrens zu konzentrieren und gleichzeitig ein höheres Maß an Sicherheit zu gewährleisten.

1.2 Problemstellung

Bevor solche Fahrassistenzsysteme in realen Fahrzeugen implementiert werden können, ist es wichtig, diese in kontrollierten Umgebungen ausgiebig zu testen. Doch selbst in solchen Testumgebungen können Fehler zu hohen Kosten durch Schäden an Hardware und noch schlimmer zu Gefährdung Anderer führen. Daher ist es sinnvoll, solche Systeme zunächst in einem kleineren Maßstab zu entwickeln und zu testen.

1.3 Ziel dieser Arbeit

Ziel dieser Arbeit ist die Entwicklung eines kleinen ferngesteuerten Fahrzeuges und die Implementierung einiger grundlegender Fahrassistenzsysteme. Dabei soll das Fahrzeug in der Lage sein eine Notbremsung durchzuführen, einer Linie zu folgen und Schilder zu erkennen und darauf zu reagieren.

Das Fahrzeug soll über eine Weboberfläche gesteuert und überwacht werden können. Hierzu sollen Sensordaten und Kamerabilder in Echtzeit übertragen werden, um dem Benutzer eine umfassende Kontrolle über das Fahrzeug zu ermöglichen.

Die Fernsteuerung soll noch weiter zur Sicherheit und Benutzerfreundlichkeit während der Vorführung und des Testens dieser Fahrassistenzsysteme dienen.

2 Hardware

In diesem Kapitel wird der Aufbau der Hardware und die Funktionsweise der einzelnen Komponenten beschrieben.

2.1 Ultraschallsensor

Ein Ultraschallsensor nutzt hochfrequente Schallwellen, um die Entfernung zu einem Objekt zu messen. Der Sensor besteht aus einem Sender, sowie einem Empfänger. Der Sender sendet Schallwellen aus, die von einem Objekt reflektiert werden und nach deren Rückkehr vom Empfänger aufgenommen werden. Daraufhin wird die Zeit gemessen, die die Schallwellen benötigt haben, um zum Objekt und zurück zum Sensor zu gelangen. Anhand dieser Zeit und der bekannten Schallgeschwindigkeit kann die Entfernung zum Objekt berechnet werden.

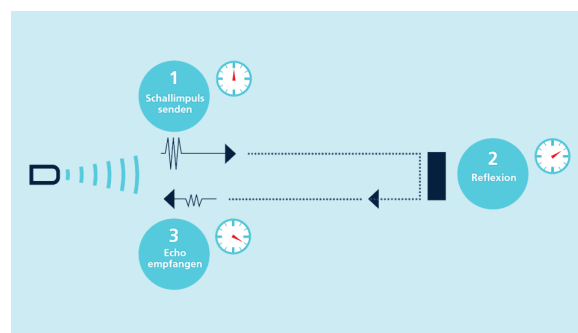


Abb. 2.1 Funktionsweise Ultraschallsensor

2.2 Raspberry Pi

2.2.1 Aufbau des Raspberry Pi

Ein Raspberry Pi ist ein Einplatinencomputer, der eine Brücke zwischen Hard- und Software bildet. Er verbindet einige der Vorteile eines herkömmlichen Computers und

eines Mikrocontrollers. Zum einen besitzt er konfigurierbare GPIO¹ Pins und Anschlüsse für Versorgungsspannung und Erdung, wodurch externe Hardwarekomponenten direkt an den Raspberry Pi angeschlossen werden können. Zum anderen ist er leistungstärker als ein Mikrocontroller. Das neueste Modell ist ausgestattet mit bis zu 16 Gigabytes an RAM, einem Arm-basierten 2,4 GHz Prozessor und einigen Schnittstellen wie beispielsweise einem LAN² Port, einer PCIe³ Schnittstelle und sowohl USB⁴-A als auch USB-C Ports.

2.2.2 Raspberry Pi als Steuergerät

Als Steuergerät für das Fahrzeug wird ein Raspberry Pi verwendet. Der Raspberry Pi verfügt über ausreichend Leistung und Schnittstellen, um die Anforderungen des Projekts zu erfüllen. Ein Mikrocontroller wurde aufgrund der benötigten Rechenleistung und geplanten Features wie der Bildverarbeitung nicht in Betracht gezogen.

Der Raspberry Pi ermöglicht die Ansteuerung der Motoren mit mithilfe der GPIO Pins, die Verarbeitung der Sensordaten und die Kommunikation mit der Weboberfläche. Zudem bietet er die Möglichkeit, verschiedene Bibliotheken und Frameworks zu nutzen, um die Implementierung

2.3 Motoren

2.4 Raspberry Pi Camera Module

Was über MP, FOV, autofokus und so schreiben?

2.5 Stromversorgung

glaub nicht nötig

¹ General Purpose Input/Output

² Local Area Network

³ Peripheral Component Interconnect Express

⁴ Universal Serial Bus

2.6 Gehäuse

2.7 Aufbau des Fahrzeugs?

- Raspberry Pi als Steuergerät - Ultraschallsensoren zur Hinderniserkennung - Motoren zur Fortbewegung - Stromversorgung - Gehäuse - Kamera zur visuellen Unterstützung/Schilderkennung

3 Softwarearchitektur

3.1 Übersicht und Aufbau der Architektur

3.2 Python

3.2.1 Programmiersprache Python

3.2.2 Verwendete Bibliotheken und Frameworks

3.3 Verwendete Technologien und Tools

3.3.1 IDEs

3.3.2 Versionsverwaltung mit Git

3.3.3 Protokolle

gRPC

HTTP

Protokolle als eigene section?

3.4 Softwarekomponenten und deren Kommunikation

3.4.1 GUI-Anwendung zur Steuerung

3.4.2 Backend-Anwendung zur Steuerung der Hardware

3.4.3 Kommunikation zwischen GUI und Backend über gRPC

3.5 Implementierte Features

3.5.1 Schilderkennung

3.5.2 Hinderniserkennung

3.5.3 Follow the line

3.5.4 Fernsteuerung

3.5.5 etc.

Grundlagen: - tools - Programmiersprache: - Bibliotheken - Frameworks - Technologien
- IDEs? - Protokolle - gRPC - HTTP - etc.

- GUI-Anwendung zur Steuerung - Backend-Anwendung zur Steuerung der Hardware
- Kommunikation zwischen GUI und Backend über gRPC - Features: - Schilderkennung
- Hinderniserkennung - Follow the line - Fernsteuerung - etc.

4 Implementierung und Integration

4.1 Hardware

4.2 Software

5 Testing und Qualitätssicherung

5.1 Teststrategie für die Überarbeitung

5.2 Definition der Testfälle

5.3 Testdurchführung zur Funktionalitätsprüfung

5.4 Evaluation der Testergebnisse

6 Fazit und Ausblick

Wir haben n richtig geiles Auto gebaut und Lehmann fand die Ausarbeitung und das Gefährt so krass, dass er uns eine 0,8 gegeben hat für unser Werk.

- Ergebnisse als eigene Datei?