



# Projektierung eines Visible Light Communication Senders

Bachelorarbeit  
von

**Edmond Krasniqi**

Fakultät für Elektro- und Informationstechnik  
Hochschule Karlsruhe Technik und Wirtschaft

Betreuer: Prof Dr.-Ing. Manfred Litzenburger  
Ko-Betreuer: Prof Dr.-Ing. Klaus Wolfrum

Karlsruhe, April 2021



# Declaration / Erklärung

Hiermit versichere ich, die vorliegende Bachelor-Thesis ohne unzulässige fremde Hilfe selbstständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie alle wörtlich oder sinngemäß übernommenen Stellen in der Arbeit gekennzeichnet zu haben.

Edmond Krasniqi  
Karlsruhe, 31.03.2021



# Acknowledgement / Danksagung

*Die Neugier steht immer an erster Stelle eines Problems, das gelöst werden will. — Galileo Galilei*

Im Studiengang Elektro- und Informationstechnik an der Hochschule Karlsruhe – Technik und Wirtschaft in Karlsruhe ist für die Beendigung des Bachelorstudiums eine Bachelor Thesis vorgesehen. Großes Interesse und Begeisterung an Elektrischen Schaltungstechnik bewegten mich zu meiner Entscheidung meine Thesis über das Thema Visible light communication (VLC) zu verfassen. Mir wurde die Möglichkeit zu Teil, vier Monate in dem Labor der Hochschule Karlsruhe zu forschen und meinen Lösungsansatz zu realisieren und zu Dokumentieren.

Hierbei konnte ich mein im Studium erlangtes Wissen anwenden, mir neue Fähigkeiten aneignen und auch umfangreiche Einblicke in die Elektrotechnik der aktuellsten kommunikationstechnischen Modelle bekommen. Deshalb möchte ich an dieser Stelle der Hochschule Karlsruhe für die produktive und intensive Zusammenarbeit danken. Mein besonderer Dank gilt Herrn Prof Dr. – Ing. Manfred Litzenburger, der mich während meiner Arbeit sehr engagiert betreut und stets konstruktiv und hilfsbereit unterstützt hat.

Des Weiteren gilt mein Dank Herrn Prof. Dr. - Ing. Manfred Litzenburger und seinen Professorenkollegen, die mich im Vorfeld in sehr umfangreichen Vorlesungen und Veranstaltungen mit sowohl einem theoretischen als auch einem praktischen Basiswissen ideal auf meine Abschlussarbeit vorbereitet haben.

Ein herzliches Dankeschön an meine Familie und Freunde ohne deren Unterstützung diese Arbeit nicht möglich gewesen wäre.

Zur Vereinheitlichung des Sprachgebrauches, verwende ich in diesem Bericht Abkürzungen. Ich verweise diesbezüglich auf das hier gegebene Abkürzungsverzeichnis.



# Kurzfassung

Hier steht die Kurzfassung





# Inhaltsverzeichnis

<b>Kurzfassung</b>	<b>vii</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Ziel der Arbeit . . . . .	1
1.3 Vorgehensweise . . . . .	1
<b>2 Grundlagen</b>	<b>3</b>
2.1 Audio . . . . .	3
2.2 Übertragungstechnik . . . . .	6
2.2.1 Aufbau eines Übertragungssystems . . . . .	6
2.2.2 Modulationsarten . . . . .	7
2.2.3 Konstellationsdiagramme . . . . .	8
2.2.4 OFDM . . . . .	10
2.2.5 DRM . . . . .	10
2.3 Elektrische Bauteile . . . . .	10
2.3.1 Die Leuchtdiode . . . . .	10
2.3.2 Der Operationsverstärker . . . . .	13
2.3.3 Der Feldeffekttransistor . . . . .	16
2.3.4 Das Digital Potentiometer . . . . .	17
2.3.5 Der DC/DC Spannungswandler . . . . .	17
2.3.6 Arduino UNO V3 . . . . .	18
2.4 Verwendete Tools . . . . .	19
2.4.1 LT-Spice . . . . .	19
2.4.2 Eagle . . . . .	19
2.4.3 Arduino Integrated Developer Environment (IDE) . . . . .	19
2.4.4 Dream . . . . .	20
<b>3 System</b>	<b>21</b>
3.1 Analoge Signalverarbeitung . . . . .	21
3.1.1 Simulation in LT-Spice . . . . .	23
3.1.2 Platinenlayout in Eagle . . . . .	23
3.1.3 Thermisches Management . . . . .	24
3.1.4 Planung und Aufbau des Gehäuses . . . . .	24
3.2 Software . . . . .	25
3.2.1 Automatisierte Amplituden-Regelung . . . . .	25
3.2.2 Übertragung mit Dream . . . . .	25
<b>4 Modell</b>	<b>31</b>
4.1 Testaufbau zur Funktionsprüfung . . . . .	31
4.2 Vorzeige Modell . . . . .	31
<b>5 Ergebnis</b>	<b>33</b>

<b>6 Zusammenfassung</b>	<b>35</b>
6.0.1 Abkürzung . . . . .	35
6.0.2 Symbol . . . . .	35
6.0.3 Bild . . . . .	35
6.0.4 Formel . . . . .	36
6.0.5 Tabelle . . . . .	36
6.0.6 Referenz und Zitat . . . . .	36
<b>Literaturverzeichnis</b>	<b>37</b>
<b>Online-Referenzen</b>	<b>37</b>
<b>Abbildungsverzeichnis</b>	<b>39</b>
<b>Tabellenverzeichnis</b>	<b>41</b>
<b>Symbolverzeichnis</b>	<b>43</b>
<b>Abkürzungsverzeichnis</b>	<b>45</b>

# **1. Einleitung**

Hier steht die Einleitung

## **1.1 Motivation**

## **1.2 Ziel der Arbeit**

## **1.3 Vorgehensweise**



## 2. Grundlagen

In diesem Kapitel werden die Grundlagen eines modernen Übertragungssystems vorgestellt. Da eine Audioübertragung mittels VLC-Technik implementiert werden soll, wird zunächst auf die Eigenschaften eines zu Übertragenden Audiosignals eingegangen. Darauf folgend werden einige Grundlagen zur Übertragungstechnik und der verwendeten elektronischen Bauteile erläutert. Hier werden grundlegende Funktionsweisen sowie deren Rolle in der analogen und digitalen Signalverarbeitung thematisiert.

### 2.1 Audio

Ziel der vorliegenden Abschlussarbeit ist die Übertragung eines Echtzeit-Audio-Signals mithilfe des projektierten VLC-Senders. Deshalb soll im nachfolgenden Kapitel anfänglich im Allgemeinen auf die Übertragung von Audiosignalen eingegangen werden. Praktische Beispiele zur Übertragung von Audiosignalen über größere Entfernung ergeben sich bei der Übertragung von Sprache und Musik. Hierbei wird zunächst der Hörschall mit Hilfe eines Mikrofon in ein proportionales elektrisches Signal umgewandelt. Dieses Signal wird anschließend auf drahtlosem oder drahtgebundenem Wege dem Empfänger übermittelt. Ein gängiges Beispiel für die drahtlose Übertragung ist die Rundfunkübertragung. Mit Telefon und Kabelrundfunk ergeben sich Beispiele für eine drahtgebundene Übertragung. Häufig kommt jedoch eine Kombination aus beidem zum Einsatz. Am Empfangsort wird das elektrische Signal mittels eines Wiedergabegerätes, wie zum Beispiel einem Lautsprecher, wieder in ein akustisches Signal überführt.

Der, durch den Lautsprecher entstandene, hörbare Schall besteht aus einer mechanischen Welle. Diese ist definiert durch eine Frequenz, welche die Tonlage bestimmt und einer Amplitude, die in diesem Fall die Lautstärke des Audiosignals vorgibt. Damit sich die Welle im Raum ausbreiten kann, um zum Empfänger zu gelangen, benötigt sie ein Übertragungsmedium. Die Dichte eines Mediums bestimmt die Geschwindigkeit, in der sich die Welle ausbreitet. Je dichter das Medium ist, desto schneller kann sich eine Welle ausbreiten.

Trifft die sich ausbreitende mechanische Welle auf das menschliche Trommelfell, wird dieses in Schwingung versetzt und regt somit die menschlichen Nerven an. Das Signal wird an dieser Stelle von einer mechanischen Welle in ein elektrisches Signal gewandelt, das über die Nervenbahnen zum Gehirn weitergeleitet wird. Im Gehirn wird das elektrische Signal analysiert und mit semantischen Informationen verbunden, die das Hören ermöglichen. Ob ein Ton von einem Menschen wahrgenommen werden kann, hängt einerseits von der Frequenz und der Amplitude des Audiosignals ab und andererseits von der wahrnehmenden

Person. So gibt es nur einen bestimmten Bereich von Frequenzen der von einem Menschen wahrgenommen werden kann. Frequenzen unterhalb des menschlichen Hörbereichs werden Infraschall genannt, darüber liegende Frequenzen werden als Ultraschall bezeichnet (siehe Abbildung 2.1). Die Grenzen dieses Hörbereiches sind nicht nur von Mensch zu Mensch unterschiedlich, sondern verschieben sich auch mit zunehmendem Alter. Daher kann der Frequenzbereich des menschlichen Hörens nicht einheitlich bestimmt werden. Dies erklärt warum zum Teil unterschiedliche Angaben in der Literatur zu finden sind. Plassmann gibt den Hörbereich zwischen 16 Hz und 16 kHz an [Pla16]. Stotz hingegen zwischen 20 Hz bis 20 kHz [Sto11]. Der hörbare Frequenzbereich eines menschlichen Ohres unterscheidet sich hingegen stark von zum Beispiel dem eines Hundes. Hunde haben ein viel feineres Gehör als Menschen. Ihr Hörbereich reicht von 65 Hz bis 45 kHz .

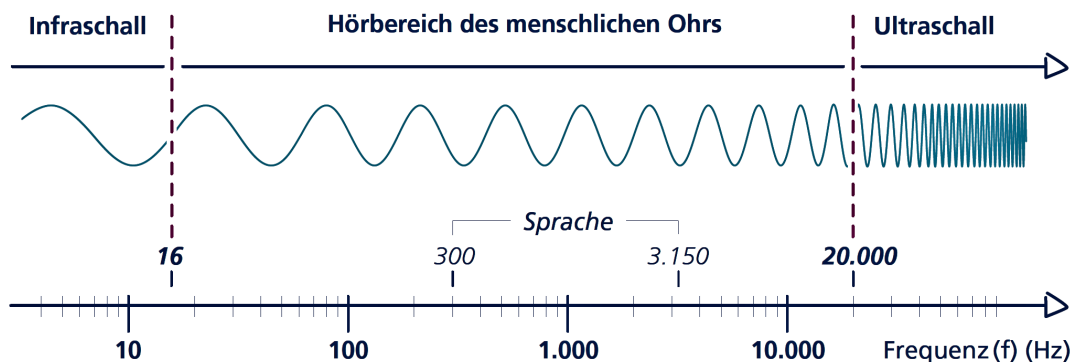


Abbildung 2.1: Hörbereich des menschlichen Ohres  
[?]

Neben dem beschreiben Hörbereich beeinflusst auch der Abstand zu einer Quelle die Wahrnehmung des Signals. Durch Freiraumdämpfung verringert sich die Amplitude einer sich ausbreitenden Welle, mit zunehmende Distanz wodurch die Lautstärke abnimmt. Befinden sich eine Quelle außerhalb der Hörreichweite, werden häufig analoge oder digitale Kommunikationssysteme verwendet, um eine Übertragung zu ermöglichen. Zur Übertragung und Weiterverarbeitung der beschriebenen mechanischen Welle, muss diese zunächst in eine elektrische Größe gewandelt werden. Für diese Wandlung wird für Audiosignale ein Mikrofon verwendet. Ähnliche wie es mit dem Trommelfell im menschlichen Ohr passiert, wird im Inneren des Mikrofons eine Membran in Schwingung versetzt und erzeugt hierdurch eine elektrische analoge Spannung. Zur digitalen Verarbeitung des Signals muss dieses mithilfe eines Analog-Digital-Wandlers (AD-Wandler) in digitale Werte gewandelt werden. Hierfür wird die Analogspannung in zeitlich regelmäßigen (äquidistanten) Abständen ausgewertet und abgespeichert. Dieser Vorgang wird Abtastung genannt. Wie in Abbildung 2.2 zu sehen, entstehen einzelne Impulse mit Werten des Signals. Zwischen den Abtastzeitpunkten wird jedoch keine Information abgefragt, hierdurch ist der Wert an diesen Stellen Null. Um das Originalsignal zu rekonstruieren, müssen durch einen Digital-Analog Wandler (DA-Wandler) die fehlenden Signalabschnitte interpoliert werden. Hierbei ist auf den Zusammenhang zwischen Abtastfrequenz und Signalfrequenz zu achten.

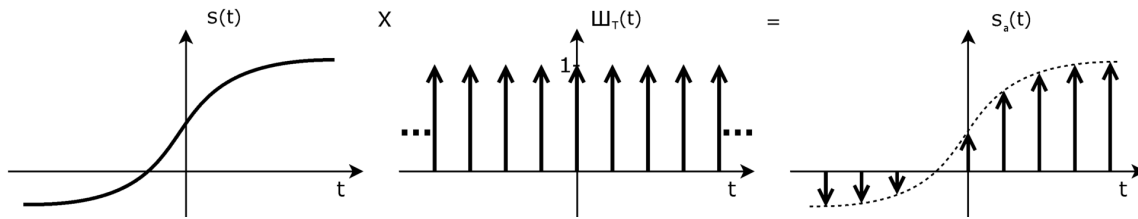


Abbildung 2.2: Abtastung eines Signals  
[?]

In Abbildung 2.2 ist zu erkennen, dass ausreichend viele Abtastwerte vorhanden sind, um den Verlauf des Signals detailliert wiederzugeben. Die gewählte Abtastfrequenz steht demnach in einem adäquaten Verhältnis zur Signalfrequenz. Mit einer sinkenden Anzahl von Abtastwerten vergrößern sich die Signallücken, wodurch die Rekonstruktion über den Analog-Digital Wandler (AD-Wandler) zunehmend erschwert wird. Nach dem Theorem von Shannon ist für die höchste Signalfrequenz pro Schwingung mindestens eine zweimalige Abtastung erforderlich, um das Signal perfekt rekonstruieren zu können. Hierbei gilt also:

$$F_A \geq 2 \cdot F_S \quad (2.1)$$

Wobei  $F_A$  für die Abtastfrequenz und  $F_S$  für die höchste im Signal enthaltene Frequenz steht. In Abbildung 2.3 befolgt die obere Abtastung die Regel von Shannon, während die unter sie verletzt. Wie zu erkennen ist, gelingt durch die richtige Abtastung eine gute Rekonstruktion, während bei Unterabtastung das Signal stark verfälscht rekonstruiert wird.

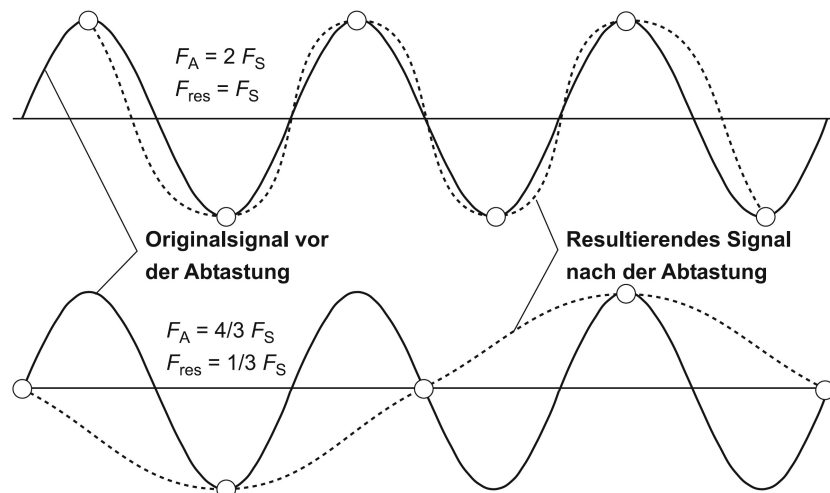


Abbildung 2.3: Unterabtastung eines Signals  
[?]

Des Weiteren ist zu erkennen, dass die obere Abtastung das Signal jeweils in seinen Minima bzw. Maxima erfasst. Bei einer Phasenverschiebung von  $90^\circ$  würde die obere Abtastung jedoch genau die Nulldurchgänge des Signals erfassen. Eine Rekonstruktion des Signals wäre dann nicht mehr möglich, da es nicht mehr von einem Nullsignal unterscheidbar wäre. Aus diesem Grund ist es zu empfehlen, die Abtastfrequenz mehr als doppelt so groß, wie die größte im Signal auftretenden Frequenz  $f$ , zu wählen. Da das menschliche

Gehör Töne, die oberhalb einer Frequenz von 20 kHz liegen, nicht mehr wahrnehmen kann, wird im Audiodbereich meistens mit 44,1 kHz oder mit 48 kHz abgetastet und somit das Shannon-Theorem eingehalten.[?]

## 2.2 Übertragungstechnik

In diesem Kapitel werden die Grundlagen der modernen Übertragungstechnik als Grundlage für die weiteren Ausführungen erklärt. Zunächst soll der generelle Aufbau eines Übertragungssystems erläutert werden, um im Folgenden einige gängige digitale Modulationsarten vorzustellen. Hierbei liegt der Fokus besonders auf einer Quadrature Phase Shift Keying (QPSK)- Modulation, welche in den später gezeigten Modellen Verwendung findet. Der letzte Abschnitt beschäftigt sich mit den Themen Bandbreitennutzung und Mehrkanalzugriff. Hier werden die aus der Mobilfunktechnik bekannten Begriffe Frequenzmultiplexverfahren (Frequency Division-Multiple-Access) (FDMA), Zeitmultiplexverfahren (Time-Division-Multiple-Access) (TDMA) und Codemultiplexverfahren (Code-Division-Multiple-Access) (CDMA) eingeführt und beschrieben.

### 2.2.1 Aufbau eines Übertragungssystems

Ein Übertragungssystem besteht stark vereinfacht dargestellt aus fünf verschiedenen Teilen. In Abbildung 2.4 ist eine Skizze eines solchen Übertragungssystems zu sehen, die im Folgenden näher beschrieben werden soll. Zunächst wird eine Quelle benötigt, welche ein zeitkontinuierliches Signal (z.B. Sprache, Musik, Bilddaten, analoge Messwerte) oder ein zeitdiskretes Signal (z.B. Buchstaben, Datensequenzen, abgetastete analoge Signale) zur Verfügung stellt. Der Sender übernimmt im System die Aufgabe, das Quellensignal in ein für die Übertragung geeignetes Format umzuwandeln. Dieses muss auf den später zur Weiterleitung des Signals genutzten Kanal abgestimmt werden. Dabei gilt es darauf zu achten, den zur Verfügung stehenden Kanal und seine Eigenschaften möglichst effizient zu nutzen. Hierbei wird zwischen drahtlosen Kanälen (elektromagnetisch, akustisch, Infrarot) und drahtgebundenen Kanälen wie beispielsweise Kupferleitungen oder Glasfaserkabeln unterschieden. Nach der Umwandlung des Originalsignals durch den Sender wird dieses über einen dieser physikalischen Kanäle gesendet. Ein Übertragungskanal ist im Allgemeinen fehlerbehaftet. Dies hat zur Folge, dass das Signal bei der Ausbreitung durch den Kanal verzerrt oder gedämpft wird. Es kann zudem zu Interferenzen mit anderen Kanälen oder Umwelteinflüssen kommen, welche das Übertragene Signal verfälschen. Die Aufgabe des Empfängers besteht in einer möglichst genauen Rekonstruktion des Quellensignals. Das reproduzierte Quellensignal gelangt schließlich an die Senke. Zum Eliminieren der im Kanal entstandenen Fehler wird dem übertragenen Signal durch verschiedene Verfahren Redundanz hinzugefügt. Die angefügte Redundanz enthält keinen Informationswert, sondern dient zur Absicherung der übertragenen Daten. Der Empfänger kann durch die Redundanz Fehler erkennen und im Optimalfall auch direkt korrigieren. Näheres hierzu wird im Kapitel 2.5 erklärt.

Nach dieser allgemeinen Darstellung des Übertragungssystems soll nun eine genauere Beschreibung der Übertragungsstrecke folgen. Diese wird sich aufgrund des Schwerpunkts dieser Abschlussarbeit auf die Übertragung durch einen drahtlosen Kanal fokussieren. Zum Aufbau eines drahtlosen Übertragungssystems muss zunächst der physikalische Kanal über welchen sich das Signal ausbreitet definiert, sowie die Signalform festgelegt werden. So verwenden Fernbedienungen im Bereich der Unterhaltungselektronik beispielsweise als Signalform Licht im Infrarot-Bereich, um Daten zu Übermitteln. Hierfür muss jedoch der direkte Einfall des Lichts auf den Empfänger gegeben sein, wodurch die Reichweite dieser Art der Datenübertragung stark begrenzt ist und sich daher nicht für andere Anwendungsgebiete wie beispielsweise den Mobilfunk eignet. Für Mobilfunkanwendungen werden in der Regel



elektromagnetische Wellen im Megahertz bis Gigahertz Frequenzbereichen verwendet. Diese Wellen bestehen aus einem elektrischen und einem magnetischen Feld, wobei die beiden Felder sich orthogonal zueinander ausbreiten. In Abbildung 2.5 ist die Ausbreitung einer solchen Welle veranschaulicht. Diese Art von Welle ist nicht auf ein Ausbreitungsmedium angewiesen und kann sich somit auch im Vakuum (z.B. im Weltraum) ausbreiten. Somit ermöglicht sie auch eine Kommunikation mit Satelliten. [But13] [An19]

Zur Erzeugung einer elektromagnetischen Welle wird zunächst eine hochfrequente Wechselspannung benötigt. Diese wird meist durch einen Quarzoszillator erzeugt. Damit sich die Welle nun ausbreiten kann, benötigt sie eine Antenne von welcher sie sich ablösen kann. Eine elektromagnetische Welle kann sich jedoch erst dann von einer Sendeantenne lösen, wenn die Wellenlänge  $\lambda$  in die Größenordnung der Antennenabmessung kommt [Fis10]. Es gilt:

$$\lambda = \frac{c}{f} \quad (2.2)$$

Ein Audiosignal erstreckt sich über einen Bereich von 20Hz bis 20kHz, wodurch nach dieser Formel eine extrem große Antenne benötigt werden würde. Hieraus ergibt sich auch die Notwendigkeit eine geeignete, d.h. im richtigen Frequenz-, also Wellenlängenbereich liegende Sende-, bzw. Trägerfrequenz zu wählen und die zu versendende Information dieser durch Modulation aufzuprägen.[Höh13]

### 2.2.2 Modulationsarten

In Kapitel 2.2.1 wurde beschrieben wie ein Übertragungssystem aufgebaut ist und aus welchen Komponenten es besteht. Es wurde auch gezeigt wie eine Welle entsteht und diese sich ausbreitet. Des Weiteren wurde die Notwendigkeit einer hochfrequenten Trägerfrequenz verdeutlicht. Im nächsten Schritt soll nun gezeigt werden, wie der Trägerwelle die geforderte Information aufgeprägt wird. Hierfür stehen die drei Parameter zur Verfügung, welche eine Welle beschreiben. Dies sind die Frequenz, die Amplitude sowie die Phase der Welle. Dabei wird die Variation der Amplitude als Amplitudenmodulation (AM) oder auch Amplitude Shift Keying (ASK) bezeichnet, eine Variation der Phase als Phasenmodulation (PM) oder auch Phase Shift Keying (PSK) sowie die Variation der Frequenz als Frequenzmodulation (FM) oder auch Frequency Shift Keying (FSK). Im Folgenden wird erläutert, wie diese Parameter verändert werden können, um der Welle die Information einzuprägen. Zunächst soll die analoge Modulation anhand eines AM-Signals und eines FM-Signals gezeigt werden. Hierbei wird das analoge Ausgangssignal nicht digital gewandelt, sondern unabgetastet verarbeitet. Bei der Amplitudenmodulation wird das zu übertragende Signal mithilfe eines Mischers, mit einer sinusförmigen Trägerfrequenz multipliziert. Das Blockschaltbild eines solchen Mischers ist in Abbildung 2.6 zu sehen. Die deutlich langsamere Frequenz des übertragenen Signals bestimmt die Amplitude der hochfrequenten Trägerfrequenz und wird deshalb Einhüllende genannt. In Abbildung 2.7 ist gut zu erkennen, wie die Amplitude der Trägerfrequenz des AM -Signals dem Signalverlauf des Eingangssignals folgt. Das Signal hat nun eine ausreichend hohe Trägerfrequenz, welche sich von einer Antenne lösen und ausbreiten kann[Fis10]. Dem Empfänger ist es wiederum möglich, durch eine Filterung mit einem Tiefpassfilter die zuvor eingefügten hohen Frequenzanteile herauszufiltern und so das Originalsignal zu reproduzieren. Zu Problemen kann es kommen, wenn sich der Empfänger bewegt. Durch diese Bewegung variiert die Ausbreitungsstrecke, was durch die Freiraumdämpfung zu einer Signalabschwächung bzw. Signalstärkung führen kann. Bei einer Audioübertragung würde dann die Lautstärke schwanken. Im Gegensatz zu einer Amplitudenmodulation liegt bei einer Frequenzmodulation die Information nicht in der Amplitude des empfangenen Signals sondern in dessen Frequenz. Wie in Abbildung 2.7 zu sehen, verändert sich bei dem frequenzmodulierten Signal lediglich die Trägerfrequenz, nicht aber die Amplitude. Hierdurch ist diese Art der Modulation störresistenter gegenüber

Amplitudenschwankungen. Allerdings können auch hier Effekte wie der Dopplereffekt das Signal beeinflussen, wenn sich Sender oder Empfänger bewegen.[Höh13]

Im Folgenden sollen nun die Grundlagen digitaler Modulationsverfahren dargestellt werden. Wie bei einer analogen Modulation, nutzen auch digitale Modulationsverfahren die Veränderung von Frequenz, Amplitude und Phase des Trägersignals. Als eine der ersten digitalen Übertragungsarten wird heute die "Übertragung von Telegrammen angesehen. Hierbei wurde eine Gleichspannung aus und ein geschaltet. Ein langes Einschalten stand dabei für eine Eins, ein kurzes Einschalten für eine Null. Da diese Übertragung anfänglich noch drahtgebundenen stattfand, wurde keine Trägerfrequenz benötigt. Auf der Empfängerseite brachte die Gleichspannung eine Lampe zum leuchten oder lies ein akustisches Signal erklingen. Je nach Länge des Einschaltens konnte somit eine Null oder eine Eins erkannt werden.[Fis10] Aufgrund der Limitierung dieses binären Systems musste die Nachricht im Vorfeld kodiert werden. Hierfür wurde das Morsealphabet verwendet. Um die Länge der übertragenen Nachrichten möglichst kurz zu halten, wurde hierbei bereits versucht, eine Redundanzreduktion zu erzeugen. Kurze Codes wurden für Buchstaben des Alphabets verwendet, die gehäuft im Sprachgebrauch auftreten, lange Codes wurden für Buchstaben verwendet, die sich weniger häufig in der Sprache wiederfinden.[Fis10] Diese Grundlage werden bis heute in der modernen Signalübertragung angewandt. In Abbildung 2.8 sind die drei digitalen Grundmodulationsarten AM, FM und PM dargestellt. Hierbei entspricht der Verlauf von Kennlinie (b) einem AM-Signal, Kennlinie (c) zeigt ein FM-Signal und in (d) ist eine Phasenmodulation zu sehen. Das dargestellte AM Signal hat bei der Übertragung einer 1 eine definierte Amplitude und bei der Übertragung einer 0 wird die Amplitude auf Null gesetzt. Da so nicht erkannt werden kann ob bei ausbleibender Amplitude die Übertragung beendet ist oder einfach nur mehrere Nullen übertragen werden, wird häufig der digitalen Null ein zweiter definierter Amplitudenwert zugeordnet. So entspricht beispielsweise 1V Amplitude einer digitalen Eins und 0,5V Amplitude einer digitalen Null. Die verschiedenen übertragenen Amplituden werden auch Symbole genannt. Die hier gezeigten Modulationsarten sind zweiwertige Modulationen. Ein übertragenes Symbol entspricht auch immer einem Bit digitale Information. Moderne Techniken verwenden allerdings höherwertige Modulationsarten, die es ermöglichen mehrere Bits pro Symbol zu übertragen. Um mit einem Symbol zwei Bit zu übertragen benötigen die Modulationsarten mindestens vier Zustände. Diese werden dann als 4-ASK respektive 4-FSK und 4-PSK genannt. Die Ziffer Vier steht hier bei für die vier verschiedenen Zustände. [But13] Oft wird die Ziffer Vier auch mit dem Buchstaben Q für Quadratur ersetzt. Des Weiteren bestehen auch Kombinationen dieser drei Grundmodulationsarten. Bei einer Quadraturamplitudenmodulation (QAM) werden beispielsweise sowohl Amplitude als auch Phasenänderung benutzt, um Informationen zu modulieren. Die aktuelle 5G-Mobilfunktechnologie nutzt im Optimalfall eine 256-QAM-Modulation und kann pro Symbol bis zu 8 Bit Daten versenden. Dies gelingt aber nur bei einem störungsfreien Kanal. Sollte der Kanal schlechter werden wird dynamisch auf niederwertige Modulationsarten wie 64-QAM gewechselt. [MN16]

### 2.2.3 Konstellationsdiagramme

Im Verlauf dieser Abschlussarbeit wurde vorwiegend mit einer QPSK-Modulation gearbeitet. Diese Modulationsart findet auch Verwendung in Mobilfunksystemen, Satellitenkommunikation sowie Kabelmodems [DEZ16]. Da QPSK eine Art der PSK-Modulation ist, findet sich die übertragene Information in der Phase wieder. Bei den modulierten Wellen handelt es sich um sinusförmige Wellen, welche wie folgt definiert sind:

Bei der verwendeten QPSK-Modulation darf  $i$  die Werte 1, 2, 3 und 4 annehmen. Hier bei gilt für die Kreisfrequenz  $\omega$ :

Die Variable  $A$  steht für die Amplitude und ist bei einer QPSK-Modulation eine Konstante.  $\phi_0$  ist die Phasenlage des Referenzsignals und  $\phi_i$  beschreibt die Phasenverschiebung des

jeweiligen Symbols. Damit die unterschiedlichen Symbole möglichst weit von einander entfernt sind, wird bei einer QPSK-Modulation jeweils um  $90^\circ$  verschoben. Dies bedeutet  $\phi_i$  kann die Werte  $0^\circ$ ,  $90^\circ$ ,  $180^\circ$ , und  $270^\circ$  annehmen. In Abbildung 2.9 sind die vier verschiedenen Wellen einer QPSK-Modulation dargestellt.

Zur Veranschaulichung der Darstellung wird in der Regel ein Konstellationsdiagramm verwendet. Dieses stellt die verschiedenen Symbole im komplexen Raum dar. Im Konstellationsdiagramm werden der Betrag und die Phasenlage der Symbole markiert. In Abbildung 2.10 ist ein solches Konstellationsdiagramm einer QPSK-Modulation dargestellt.

Wie zu erkennen ist, sind die einzelnen Symbole kreisförmig um den Ursprung angeordnet. Der Abstand der Symbole zum Ursprung ist dabei konstant. Auch der Abstand der Symbole zu benachbarten Symbolen ist identisch. Bei Betrachtung der Zuordnungen der Symbole zu den jeweiligen Dezimalwerten fällt jedoch auf, dass diese nicht aufsteigend angeordnet sind. Diese zunächst willkürlich wirkende Zuordnung bezweckt, dass die Hamming-Distanz "benachbarter Symbole maximal Eins ist, um die Fehlerwahrscheinlichkeit zu minimieren. Die Hamming-Distanz gibt an, wie viele Bitwerte sich gegenüber dem vorherigen Symbol verändern. Bei einer dezimal aufsteigenden Anordnung verhält sich die Hamming-Distanz wie folgt:

Für den Sprung von dezimal Null auf dezimal Eins, muss sich lediglich das letzte Bit verändern. Die Distanz ist also Eins. Für den Sprung von dezimal Eins auf dezimal Zwei müssen sich jedoch beide Bits ändern, somit ist die Distanz hier Zwei. Das Selbe gilt für den Sprung von Drei auf Null. Um eine konstante Hamming-Distanz zu erzeugen wird der Gray-Code verwendet. Dieser ordnet die Dezimalzahlen nicht nach ihrem Wert aufsteigend, sondern erzeugt eine konstante Hamming-Distanz von eins. Wird nun durch einen Übertragungsfehler fälschlicherweise ein benachbartes Symbol anstelle des gesendeten erkannt, so ergibt sich maximal ein Bitfehler. [Man17] Wie in Abbildung 2.10 zu erkennen, sind die Symbole zunächst auf der reellen- bzw. der imaginären Achse angeordnet. Die Symbole sind durch einen kleinen Punkt gekennzeichnet. Auf der Seite des Senders werden diese Symbole perfekt im komplexen Zahlenraum angeordnet und versendet, allerdings werden sie auf dem Weg zum Empfänger in der Phase und auch in der Amplitude gestört. In Abbildung 2.11 ist ein Konstellationsdiagramm zu sehen, bei welchem den Symbolen vor dem Empfang ein Rauschen aufaddiert wurde.

Die empfangenen Symbole sammeln sich um die vier definierten Symbolpunkte. Da das Rauschen nur gering ist, gelingt es mit dem bloßen Auge jedes empfangene Symbol dem eigentlich gesendete Symbol zuzuordnen. Bei stärkerem Rauschen wurde die Entscheidung allerdings zunehmend schwerfallen. Deshalb werden klare Grenzen definiert, welche die Symbole von einander abgrenzen. Im ersten Moment wird davon ausgegangen, dass die Symbole statistisch unabhängig voneinander sind. Dies bedeutet, dass ein empfangenes Symbol in keinem Zusammenhang mit dem zuvor empfangenen Symbol steht und auch keine Auswirkung auf das folgende Symbol hat. Hierdurch ergibt sich bei einer QPSK-Modulation für jedes Symbol eine Auftrittswahrscheinlichkeit von 25%. Die Entscheidungsgrenzen müssen also den Empfangsraum in vier gleichgroße Felder unterteilen und somit mittig zwischen den benachbarten Symbolen verlaufen. In den gezeigten Konstellationsdiagrammen laufen diese Entscheidungsgrenzen im  $45^\circ$ -Winkel und im  $-45^\circ$  Winkel durch den Ursprung. Der Empfänger muss nun die empfangenen Symbole mit den Entscheidungsgrenzen vergleichen. Hierfür müssen allerdings für jedes Symbol zwei Geradengleichungen gelöst und die Ergebnisse mit dem empfangenen Wert verglichen werden. Zur Vereinfachung dieses Prozesses wird bei einer QPSK-Modulation in der Praxis eine Phasenverschiebung des Referenzsignals um  $\phi_0=45^\circ$  verwendet. Ein Konstellationsdiagramm mit den neu angeordneten Symbolen ist in Abbildung 2.12 zu sehen.

Durch die Phasenverschiebung des Referenzsignals um  $\phi_0=45^\circ$  sind die Symbole zwar

nicht mehr auf der reellen- bzw. der imaginären Achse, allerdings verlaufen die Entscheidungsgrenzen nun auf genau diesen Achsen. Hierdurch muss der Empfänger lediglich das Vorzeichen des Imaginär- und des Realwerts bestimmen. Die Zuordnung des Symbols kann dann anhand von 2.3 vorgenommen werden:

Bei der Betrachtung einer höherwertigen PSK-Modulation ist dieses Verfahren allerdings nicht mehr anwendbar. In Abbildung 2.13 ist beispielsweise auf der linken Seite eine 16-PSK zu sehen. Hier ist auch zu erkennen, dass die Symbole dichter aneinander gerückt sind und der Übertragungskanal nicht sehr stark verrauscht sein darf, um eine fehlerfreie Übertragung zu gewährleisten. Es bietet sich deshalb an, die Information nicht nur in einer Variablen der Wellengleichung, sondern in mehreren unterzubringen. Beispielhaft hierfür ist in Abbildung 2.13 auf der rechten Seite eine 16-QAM zu sehen. Hierbei wird sowohl Phase als auch Amplitude des Signals verändert.

## 2.2.4 OFDM

## 2.2.5 DRM

# 2.3 Elektrische Bauteile

In diesem Kapitel werden grundlegende Bauteile eines VLC-Senders dargestellt und näher erläutert. Diese sind essenziell um die Datenübertragung mittels dem optischen Kanal zu ermöglichen.

## 2.3.1 Die Leuchtdiode

Eine Light Emitting Diode (LED) ist ein licht emittierendes Halbleiter-Bauelement mit einem pn-Übergang. Ihre elektrischen Eigenschaften stimmen mit der einer Standard Diode überein, wodurch sie in nur eine Richtung leitend ist und in die entgegengesetzte Stromrichtung sperrt. Wenn durch einen eingekoppelten elektrischen Strom die LED in Durchlassrichtung betrieben wird, findet eine Lichtemission statt.[?]

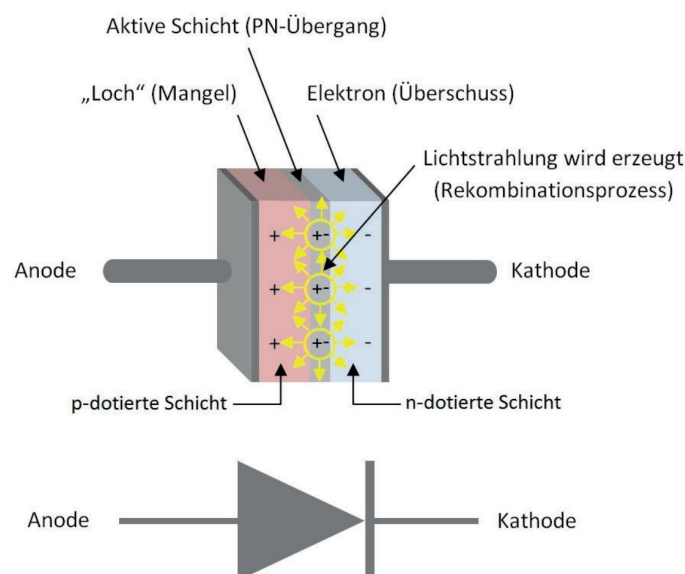


Abbildung 2.4: Strahlungserzeugung in der LED am pn-Übergang  
[?]

Der Grundsatz der Lichterzeugung in einer LED beruht auf einem Halbleiterkristall, der durch das Einbringen von Fremdatomen so dotiert ist, dass in der Diode zwei Gebiete

entstehen. In einem Gebiet entsteht ein Elektronenüberschuss und in dem anderen Gebiet entstehen Löcher. Durch das injizieren von Elektronen aus der positiv dotierten Seite in die Sperrschicht, rekombinieren sich Löcher und Elektronen, wodurch Energie in Form von Licht abgegeben wird.[?] Die Farbe hängt dabei vom Halbleitermaterial und der genauen Dotierung ab. Zudem ist dieser Rekombinationsprozess stark Temperaturabhängig. Dies wird in einem noch folgenden Kapitel näher erläutert.

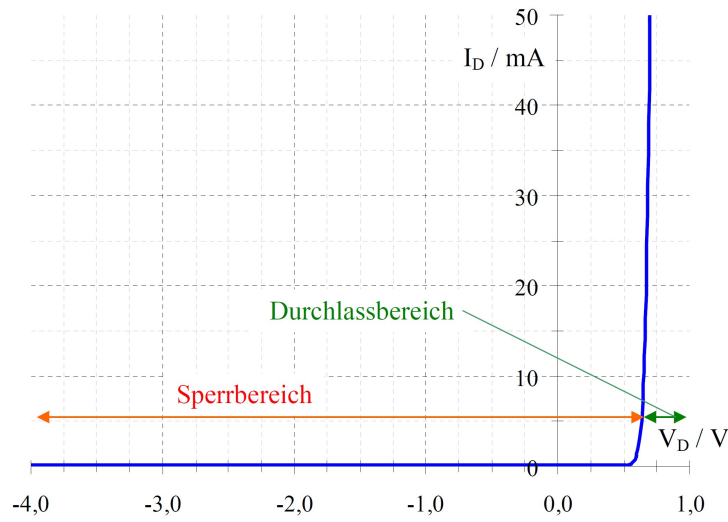


Abbildung 2.5: Diodenkennlinie  
[?]

In der Abbildung 2.5 ist die Strom-/Spannungskennlinie einer Diode in Durchlassrichtung dargestellt. Die Kennlinie einer LED hat den selben Verlauf, jedoch ist die Durchlassspannung je nach gewählter Farbe nicht bei ca. 0,7V sondern bei bis zu 4V bei einer blauen LED.

Wie in Abbildung 2.5 illustriert ist, ändert sich die Spannung in ihrem Verlauf ab einem gewissen Punkt nur noch minimal. Das bedeutet, dass sich ab einer gewissen angelegten Spannung lediglich der Strom noch weiter erhöhen kann. Da die Leuchtintensität der LED von dieser Höhe des Stromdurchflusses abhängt, führt dies zu der Betrachtung, den durchfließenden Strom statt der angelegten Spannung zu regulieren. Stellt man nun den durch die LED fließenden Strom mit der Ausgangsleistung ins Verhältnis, ergibt sich ein Zusammenhang wie ihn Abbildung 2.7 zeigt.

Farbe	Wellenlänge $\lambda$ in nm	Flussspannung $V_{D0}$ in Volt	Werkstoff
Infrarot	$\lambda > 760$	$V_{D0} < 1,9$	Galliumarsenid (GaAs) Aluminiumgalliumarsenid (AlGaAs)
■ Rot	$610 < \lambda < 760$	$1,63 < V_{D0} < 2,03$	Aluminiumgalliumarsenid (AlGaAs) Galliumarsenidphosphid (GaAsP) Aluminiumgalliumindiumphosphid (AlGaInP) Galliumphosphid (GaP)
■ Orange	$590 < \lambda < 610$	$2,03 < V_{D0} < 2,10$	Galliumarsenidphosphid (GaAsP) Aluminiumgalliumindiumphosphid (AlGaInP) Galliumphosphid (GaP)
■ Gelb	$570 < \lambda < 590$	$2,1 < V_{D0} < 2,18$	Galliumarsenidphosphid (GaAsP) Aluminiumgalliumindiumphosphid (AlGaInP) Galliumphosphid (GaP)
■ Grün	$500 < \lambda < 570$	$2,18 < V_{D0} < 2,48$	Indiumgalliumnitrid (InGaN) Galliumnitrid (GaN) Galliumphosphid (GaP) Aluminiumgalliumindiumphosphid (AlGaInP) Aluminiumgalliumphosphid (AlGaP)
■ Blau	$450 < \lambda < 500$	$2,48 < V_{D0} < 3,7$	Zinkselenid (ZnSe) Indiumgalliumnitrid (InGaN) Siliziumkarbid (SiC)
■ Violett	$400 < \lambda < 450$	$2,76 < V_{D0} < 4,0$	Indiumgalliumnitrid (InGaN)
Ultraviolett	$230 < \lambda < 400$	$3,1 < V_{D0} < 4,4$	Aluminiumnitrid (AlN) Aluminiumgalliumnitrid (AlGaN) Aluminiumgalliumindiumnitrid (AlGaInN)

Abbildung 2.6: Flussspannungen von LEDs verschiedener Farben  
[?]

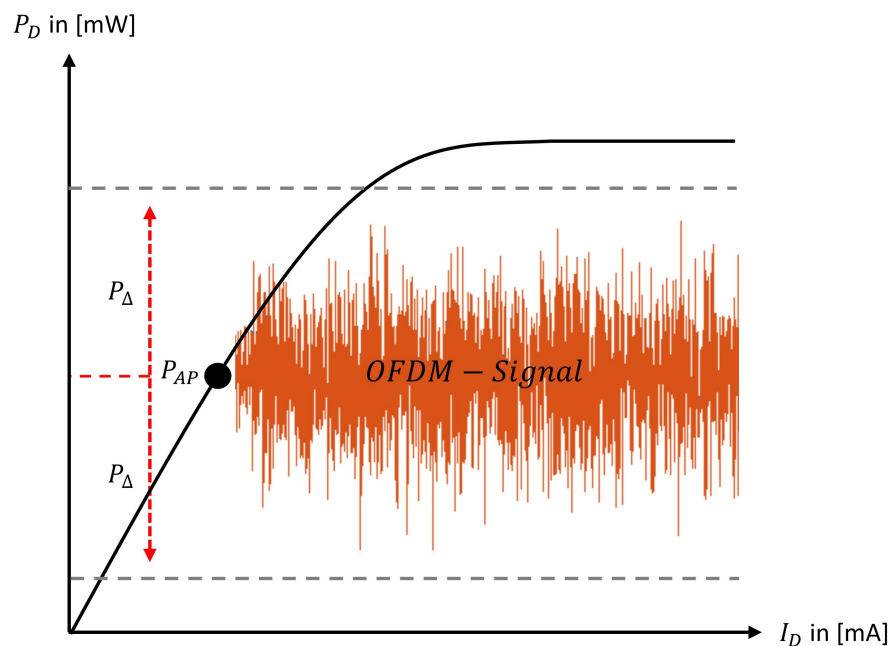


Abbildung 2.7: Kennlinie einer Diode – Lichtleistung zu fließendem Strom  
[?]

An dieser Stelle wird verdeutlicht, dass es zwei außerordentlich nicht-lineare Abschnitte in der Kennlinie gibt, welche sich durch ihre stark nicht-linearen Eigenschaften keineswegs zur Datenübertragung eignen. Zwischen diesen Bereichen um die etwa mittlere Lichtleistung, gibt es jedoch einen linearen Bereich, welcher sich ausgezeichnet für die Übertragung von Daten eignet. Dies bedeutet, dass die LED auf einem festen Arbeitspunkt AP betrieben werden muss. Dieser Arbeitspunkt liefert der Diode einen immensen Aktionsradius. Wodurch eine maximale Aussteuerung der Amplitude des Signals und somit eine verbesserte Konstellation zur Signalübertragung gewährleistet wird.

### 2.3.2 Der Operationsverstärker

„Im Grunde besteht kein Unterschied zwischen einem normalen Verstärker und einem Operationsverstärker. Beide dienen dazu, Spannungen bzw. Ströme zu verstärken. Während die Eigenschaften eines normalen Verstärkers jedoch durch seinen inneren Aufbau vorgegeben sind, ist ein Operationsverstärker so beschaffen, dass seine Wirkungsweise überwiegend durch eine äußere Gegenkopplungs-Beschaltung bestimmt werden kann. Um dies zu ermöglichen, werden Operationsverstärker als gleichspannungsgekoppelte Verstärker mit hoher Verstärkung ausgeführt. Damit keine zusätzlichen Maßnahmen zur Arbeitspunkteinstellung erforderlich werden, verlangt man ein Eingangs- und Ausgangsruhepotential von 0V. Deshalb sind in der Regel zwei Betriebsspannungsquellen erforderlich: eine positive und eine negative.“([?],S.491)

Ein Operationsverstärker (OP) kann also mit einem Differenzverstärker mit einer theoretisch unendlichen Verstärkung verglichen werden. Diese bezeichnet man als Leerlaufverstärkung. Schließlich führt das dazu, dass eine Gesamtverstärkung der Schaltung von einer zusätzlichen externen Verdrahtung abhängt. Diese wird von einem rückgekoppelten Netzwerk hergestellt und nennt sich Schleifenverstärkung.[?]

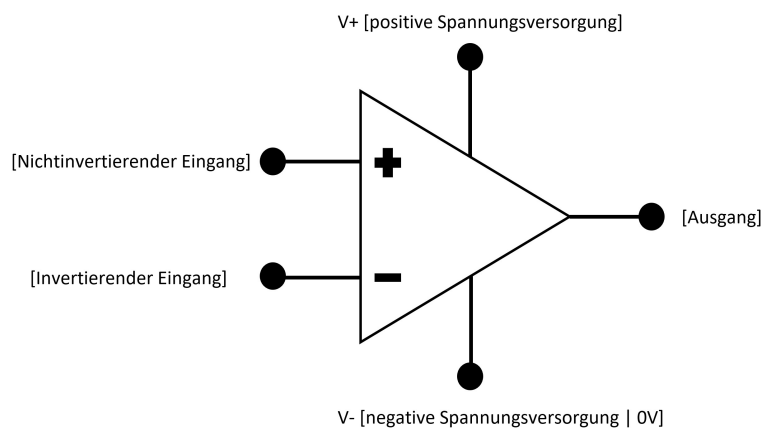


Abbildung 2.8: Operationsverstärker Anschlussschema  
[1]

Sie werden im Allgemeinen in Form einer integrierten Schaltung (Integrated Circuit (IC)) hergestellt, da die in dem Datenblatt eines Operationsverstärkers angegebenen Werte ausreichend beschrieben werden. Zudem verfügt der OP über einen invertierenden (-) und nicht-invertierenden (+) Eingang. Daraus können zwei der OP-Grundsaltungen extrahiert werden. Um das Verhältnis zwischen der Ausgangsspannung  $U_a$  und der Eingangsspannung  $U_e$  zu berechnen werden häufig die Parameter A und G verwendet. [Halbleiter Buch]

$$A = \frac{U_a}{U_e} \quad (2.3)$$

Außerdem haben OPs einen unendlich großen Eingangswiderstand und einen sehr geringen Ausgangswiderstand. Das Ruhepotential zwischen dem invertierenden und nichtinvertierenden Eingang ist beinahe Null weshalb zwischen beiden Eingängen keine Spannung abfällt. Zudem fließt kein Steuerstrom, d.h. es fließt kein Strom in die Eingänge des OP, da diese einen unendlichen Eingangswiderstand besitzen.[?]

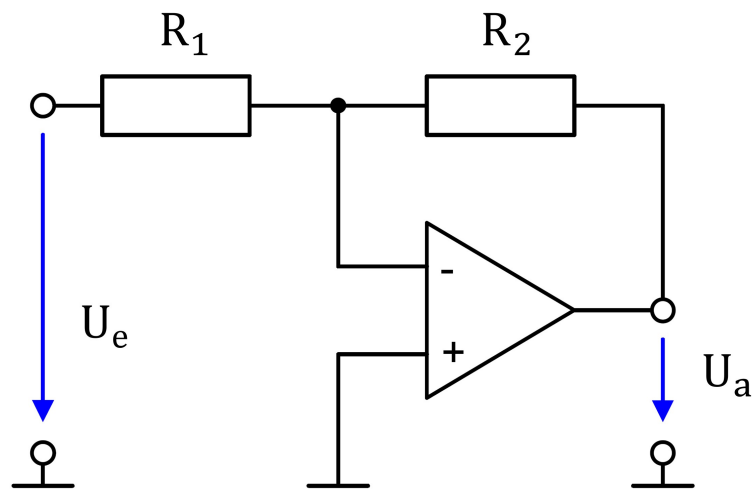


Abbildung 2.9: Invertierender Operationsverstärker  
[1]

Abbildung 2.9 zeigt die klassische Schaltung eines invertierenden Operationsverstärkers mit Rückkopplungszweig. Dieser definiert über den Widerstand  $R_2$  die Schleifenverstärkung, indem auf den invertierten Eingang ein Teil des Ausgangssignals zurückgeführt wird. Bei einem steigenden Ausgangssignal wird so dem steigenden Eingangssignal entgegengewirkt. Dies gewährleistet, dass das Ausgangssignal nicht unendlich verstärkt werden kann. Der invertierende Eingang wird mit Masse verbunden. Zuletzt wird hier die Spannungsverstärkung  $A$  der invertierenden Grundsaltung über den Rückkopplungspfad bestimmt. Diese ergibt sich hier also zu:

$$A = -\frac{R_2}{R_1} \quad (2.4)$$

und das negative Vorzeichen bedeutet, dass hier eine Phasendrehung des Signals um  $180^\circ$  stattgefunden hat. Der invertierende Verstärker besitzt zudem die Fähigkeit Signale zu dämpfen. Daher wird er häufig für Filterschaltungen und messtechnische Zwecke benutzt.

Die folgende Abbildung 2.9 illustriert den nichtinvertierenden Operationsverstärker in seiner Grundsaltung. Die Spannungsverstärkung berechnet sich anhand der Spannungsteiler Beziehung

$$U_a = U_e \cdot \frac{R_1 + R_2}{R_1} = U_e \cdot \left(1 + \frac{R_2}{R_1}\right) \quad (2.5)$$

daraus ergibt sich dann

$$A = \frac{U_a}{U_e} = 1 + \frac{R_2}{R_1} \quad (2.6)$$



Wegen seines kleinen Ausgangs- und großen Eingangswiderstandes eignet sich diese OP-Schaltung sehr gut als Wechselspannungsverstärker und Impedanzwandler. Die Übertragungskennlinie ist in Abbildung 2.11 dargestellt.

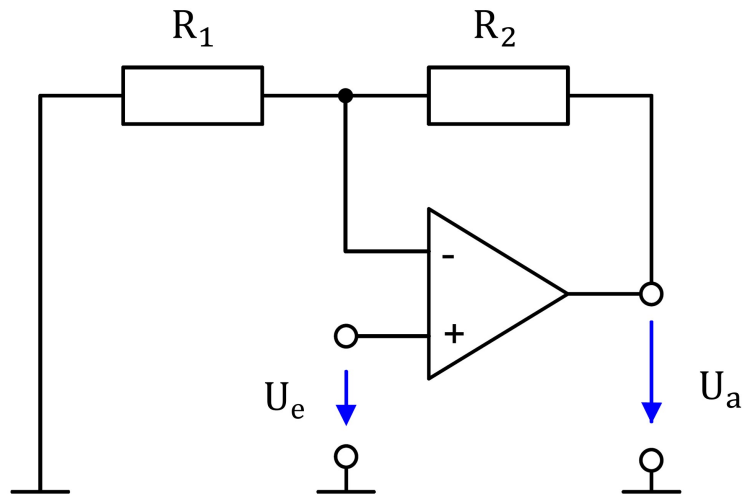


Abbildung 2.10: Nichtinvertierender Operationsverstärker  
[1]

Es zeigt auch die maximale Aussteuerbarkeit am Ausgang der OPs, welche innerhalb  $V- < U_a < V+$  liegt. Werden entweder positive oder negative Grenze erreicht, kann  $U_a$  nicht weiter ansteigen. Dieser Zustand nennt sich Übersteuerung. Die klassischen Aussteuerungen liegen in etwa 1V unter der Versorgungsspannung.[?]

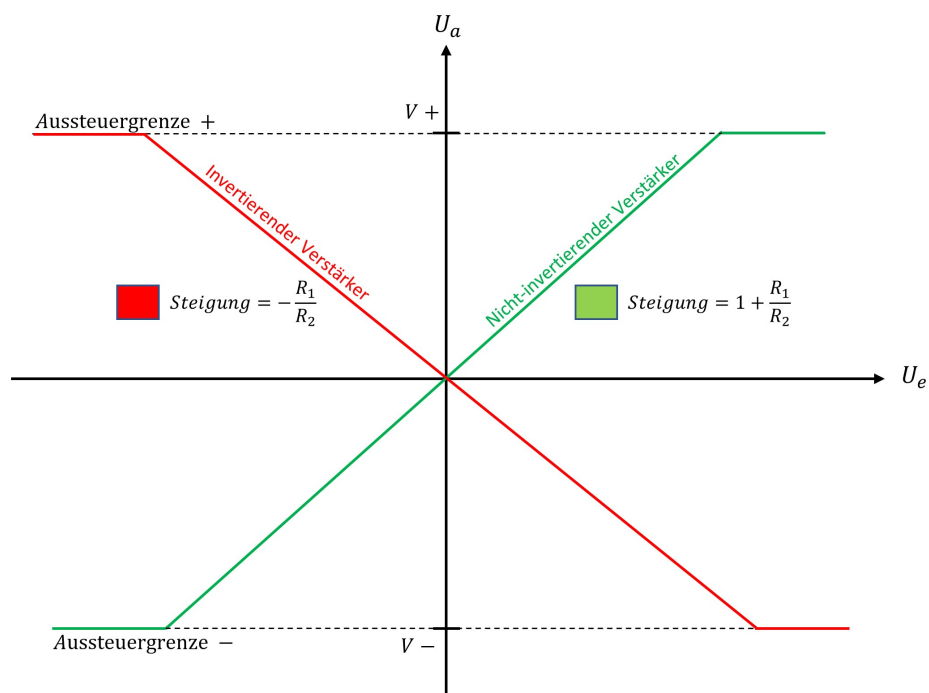


Abbildung 2.11: Operationsverstärker  
[1]

### 2.3.3 Der Feldeffekttransistor

Die am häufigsten eingesetzten Leistungsschutzschalter im Spannungsbereich von bis etwa 250V sind Metall-Oxid-Halbleiter-Feldeffekttransistor (MOSFET).[7]

Sie sind eine Sonderform der Transistoren welche auch unipolare Transistoren genannt werden. Sie besitzen einen Kanal aus Halbleitermaterial, auf dem Horizontal zur Stromrichtung ein elektrisches Feld entsteht, welches den Querschnitt des Kanals verändert. Damit wird der Stromfluss durch das Bauteil geregelt. Im Field-Effect Transistor (FET) steuert eine Spannung den Strom. Es gibt vier Grundbauformen von MOSFET. Einen Negativ-Metall-Oxid-Halbleiter (NMOS) und einen Positiv-Metall-Oxid-Halbleiter (PMOS) Typus und davon jeweils eine selbstsperrende und eine selbstleitende Variante. Im Anwendungsfall dieser Thesis wird ein PMOS verwendet. [?]

### 2.3.4 Das Digital Potentiometer

Um den Strom durch die LED digital zu regulieren und demgemäß für die Datenübertragung die Helligkeit der LED zu variieren, wird ein Digitalpotentiometer benötigt. Hierfür wurde der IC MCP42010 in der 10k $\Omega$ -Variante mit 2 Kanälen ausgewählt. Dieses Digitalpotentiometer kann über ein Serial Peripheral Interface (SPI) mittels eines Arduino angesteuert werden um dessen Widerstandswert in 256 Schritten zu verändern. Dies ermöglicht eine Variation des Widerstandes in einem Bereich zwischen 52 $\Omega$  (00h) und 10k $\Omega$  (FFh) in 39 $\Omega$  Schritten. Die vom IC vorgeschriebene Versorgungsspannung von 5V stellt der Spannungsversorgungspin des Arduino bereit. Außerdem kann dem Datenblatt des ICs die Information entnommen werden, dass dessen Eingangsspannungsbereich an den Widerstandseingängen maximal zwischen -0.5V und 6V liegen darf. Da der IC in dieser Schaltung mit 5V betrieben wird, dürfen die Widerstandseingänge mit einer Spannung von maximal -0.6V bis 6V belastet werden. Das ist ein äußerst wichtiges Kriterium für die Auslegung der Schaltung, da hier über die Grenzen der maximalen Amplitude entschieden werden muss, um vor unerwünschten Nebeneffekten vorzubeugen.

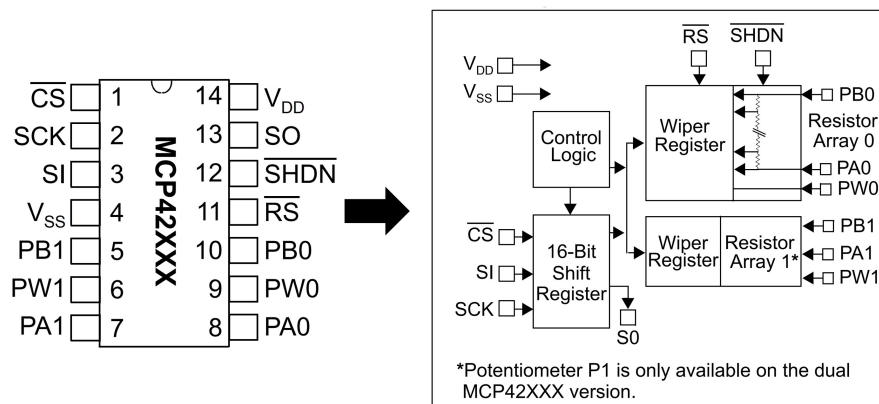


Abbildung 2.12: Aufbau des MCP42010

[?]

Die Funktion der einzelnen Pins und wie man den IC mit dem Arduino verbindet wird in nachfolgender Tabelle 2.1 beschrieben. Aufgrund des Anwendungsfalles in dieser Thesis, werden Pin 11 und Pin 12 des MCP42010 nicht benötigt und aus diesem Grund nicht am Arduino angeschlossen.

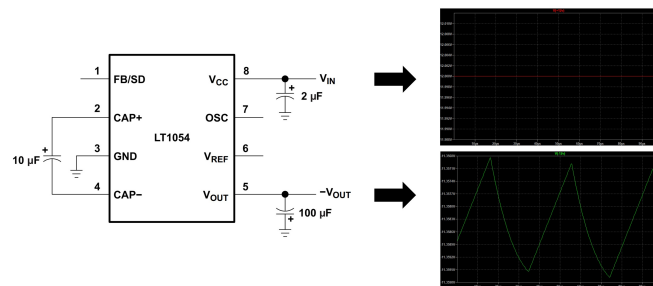
### 2.3.5 Der DC/DC Spannungswandler

Ein Gleichspannungswandler oder auch DC-DC-Wandler, wandelt eine der Schaltung zugeführte Eingangsgleichspannung in eine geregelte Ausgangsgleichspannung, welche ein anderes Spannungsniveau als die Eingangsspannung aufweist. Diese kann beispielsweise niedriger, höher oder auch Invertiert sein. Da die zu übertragenden Daten am Ausgang der Soundkarte als reine Wechselspannung anliegen, wird eine negative Spannungsquelle für die vorhandenen Operationsverstärker benötigt, um kein Risiko auf Datenverluste einzugehen. Gleichspannungswandler werden grundsätzlich immer dort eingesetzt, wo die zu Verfügung stehende Eingangsspannung nicht zur Versorgung der im Schaltkreis folgenden elektronischen Bauteile passt.[AbcderPowerModuleGrundlagen.pdf] Aufgrund der Unkonventionalität sowohl eine negative als auch eine positive Spannungsquelle simultan an die Platine anzuschließen, wandelt der DC/DC-Spannungswandler diese negative Spannung direkt auf der Platine um. Der LT1054 wird auch "negative voltage generator" genannt. Das ist ein Bauteil welches eine negative Ausgangsspannung erzeugt. Diese ist proportional

Pin Nr.	Name	Funktion	Arduino Pin Nr.
1	$\overline{CS}$	Chip Select	10
2	SCK	Serial Clock	13
3	SI	Serial Data Input	11
4	$V_{SS}$	Ground	Ground
5	PB1	Terminal B Connection For Pot 1	X
6	PW1	Wiper Connection For Pot 1	OP-1 Minus
7	PA1	Terminal A Connection For Pot 1	OP-1 Out
8	PA0	Terminal A Connection For Pot 0	5V
9	PW0	Wiper Connection For Pot 0	OP-2 Offset
10	PB0	Terminal B Connection For Pot 0	Ground
11	$\overline{RS}$	Reset Input	X
12	$\overline{SHDN}$	Shutdown Input	X
13	SO	Data Out for Daisy-Chaining	X
14	$V_{DD}$	Power	5V

Tabelle 2.1: Anschlussinformation und Pinbelegung des MCP42010

zur Eingangsspannung  $V_{CC}$ . Zum nutzen dieses Bauteiles als einfachen DC-DC-Wandler mit invertierender funktion muss es laut Datenblatt an den Ausgängen mit zusätzlichen Kondensatoren beschalten werden. Aufgrund dessen, dass der DC-DC-Wandler mit negativen Spannungen arbeitet, ist es signifikant auf die Polung der Kondensatoren zu achten, im Falle der Nutzung von Elektrolytkondensatoren. Jene sind unidirektional. Dies ist in Abbildung 2.13 illustriert. Zusätzlich ist zu erkennen, dass am negativen Ausgang des DC-DC-Wandlers nicht genau -12V anliegen sondern etwa -11.35V. Dies ist auf den Wirkungsgrad des DC-DC-Wandlers zurückzuführen und ist unter - Voltage Loss - im Datenblatt zu finden.

Abbildung 2.13: Beschaltung und erzeugte Spannung des LT1054  
[?] [?]

Der Spannungswandler arbeitet außerdem intern für die Invertierung mit einer Nennfrequenz von 25kHz. Da die zu übertragenden Daten auch nahe diesem Bereich liegen, empfiehlt es sich den Pfad der Signalverarbeitung und den der Spannungsinvertierung auf der Platine möglichst weit auseinander zu platzieren um möglichen Parasitären Störungen vorzubeugen.

### 2.3.6 Arduino UNO V3

Die Arduino-Plattform besteht prinzipiell aus der auf C und C++ basierenden Entwicklungsumgebung und der zu Programmierenden Hardware[2]. Vorteilhaft ist hier die gute

Dokumentation und eine große Menge an Bibliotheken zum Einbinden externer Hardwarekomponenten. Ein zusätzlicher Vorteil der Plattform liegt in ihrer Quelloffenheit. Alle zugehörigen Komponenten sind demnach 'Open Source'. Das hat zur Folge, dass Codes der Bibliotheken und des Bootloaders frei zur Verfügung stehen und nach Belieben verändert werden können. Auch die Architektur der Hardware ist offengelegt, weshalb Hersteller eigene kompatible Boards konstruieren und verkaufen können. Durch die Unabhängigkeit von Programmierer und Hersteller entsteht eine erschwingliche Hardware, welche bezüglich der Quelloffenheit ohne Einschränkungen programmiert werden kann.[10 - Evans, Beginning Arduino Programming. Apress, 2011, ISBN: 978-1-4302-3777-8] Diese eignet sich hervorragend zur Funktionsprogrammierung von ansteuerbaren Komponenten im analogen Signalverarbeitungspfad.

## 2.4 Verwendete Tools

### 2.4.1 LT-Spice

LT-Spice ist eine SPICE-basierte Computersoftware zur Simulation analoger elektronischer Schaltungen. Diese wurde vom Halbleiterhersteller Analog Devices (ursprünglich von Linear Technology) entwickelt. Es ist die am weitesten verbreitete und verwendete SPICE-Software in der Elektro-Simulationsbranche. Obwohl es sich um freie Software handelt, ist LT-Spice nicht künstlich eingeschränkt, um seine Funktionalität einzuschränken.[3] Diese Simulationssoftware würde im Rahmen dieser Arbeit zur Erprobung und Simulation des analogen Signalverarbeitungsschaltkreises verwendet.

### 2.4.2 Eagle

EAGLE ist eine Software zum Designen und Erstellen von Leiterplattenlayouts. Die Software verfügt über einen Schaltplan- und einen Layouteditor, sowie über eine sehr umfassende Bibliothek an Bauteilen, welche sehr unkompliziert und individuell erweiterbar ist.[5][6]

### 2.4.3 Arduino IDE

Die Sprachen, welche in der Entwicklungsumgebung genutzt werden, sind C und in kleineren Umfängen auch C++. In den Standardbibliotheken der Entwicklungsumgebung sind die für die Programmierung erheblichsten Funktionen zusammengefasst. Um weitere Funktionen zu nutzen, können dementsprechend zusätzliche Bibliotheken eingebunden werden. Der Editor mit integriertem Compiler ist ein weiterer Bestandteil der IDE. In diesem wird der Code geschrieben, kompiliert und auf das Board überspielt. User benötigen also nur eine Software, um ihre Mikrocontroller in vollen Umfängen nutzen zu können. Dieses IDE bietet außerdem die Möglichkeit, Bibliotheken oder gar Programmbeispiele herunterzuladen.

[10B. Evans, Beginning Arduino Programming. Apress, 2011, ISBN: 978-1-4302- 3777-8.].

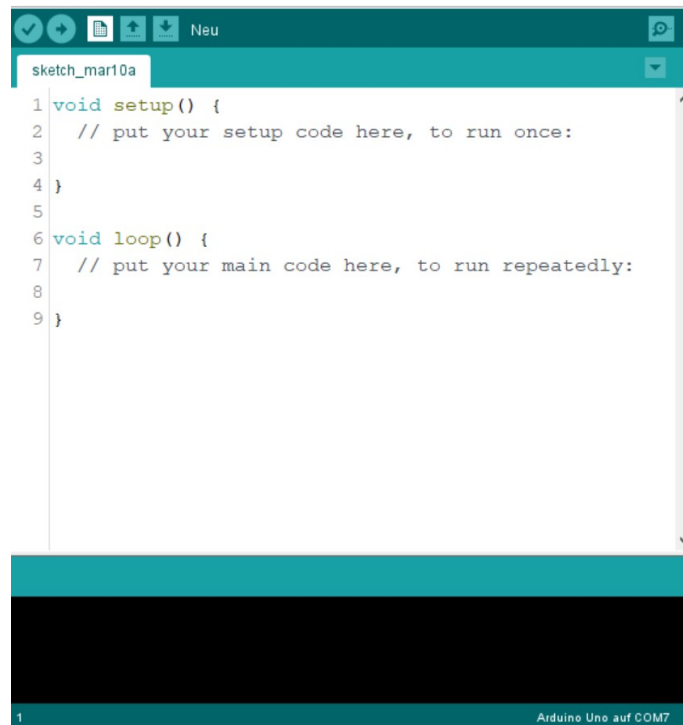


Abbildung 2.14: Arduino IDE  
[1]

Arduino-Programme basieren immer auf einem einheitlichen Grundaufbau. In Abb. ist dieser zu erkennen. Es sind zwei Funktionen namens `setup()` und `loop()` deklariert. `Setup()` wird einmalig zu Beginn des Programms aufgerufen, während `loop()` direkt im Anschluss aufgerufen wird, bis der Mikrocontroller stromlos geschaltet wird. Oberhalb der Funktion `setup()` können außerdem noch Präprozessorbefehle programmiert werden.

#### 2.4.4 Dream

Die Software Dream bietet eine alternative Radiosignale auf dem Computer zu empfangen und auch zu senden. Sie war ursprünglich als Forschungsprojekt des Fraunhofer Instituts angesetzt. Signale, welche nicht über den PC empfangen werden können, werden über den Mikrofoneingang der Soundkarte empfangen. Bei alledem benutzt das Konzept internationale Direktiven für Amateurfunk, Radio und Informationsdienste. Diese Eigenschaft ermöglicht es dem User FM, AM und Digital Radio Mondiale (DRM) zu empfangen. Außerdem visualisiert die Software in Echtzeit zahlreiche Daten, Statistiken und Diagramme des empfangenen Signals. [4] Die Modulation und das Senden des OFDM-Signals über die konstruierte Hardware erfolgt durch diese Software und wird in folgenden Kapiteln noch weiter intensiviert.

## 3. System

### 3.1 Analoge Signalverarbeitung

Bei der Signalübertragung über den optischen Kanal mit dem in dieser Arbeit gewählten Medium Licht, müssen einige grundlegende Dinge beachtet werden. Im Kapitel 2.3.1 der Leuchtdiode wurden ihre grundlegenden Eigenschaften erklärt, die es sich hier nun zu verwenden gilt. Eine bekannte Schwierigkeit ist die Übertragung von negativen Wellen. Diese können nicht übertragen werden, da Licht keinen negativen Wert annehmen kann. Hinzu ist man bei der Übertragung von Signalen darauf bedacht, nur im linearen Bereich der LED-Kennlinie Daten zu übertragen. Wenn man diese nämlich im nichtlinearen Bereich der LED überträgt, können Verzerrungen auftreten und somit die Übertragung stark gestört werden. Um jenes Problem zu lösen, wird ein Offset verwendet, welcher das Signal so positiv verschiebt, dass dieses keine negativen Anteile mehr besitzt. Zudem wurde ein Spannungspuffer eingerichtet um zusätzlich die Amplitude zu verstärken. Dies liefert Gewissheit, dass nur positive Spannungsanteile übertragen werden und somit kein Signalverlust verzeichnet werden muss.

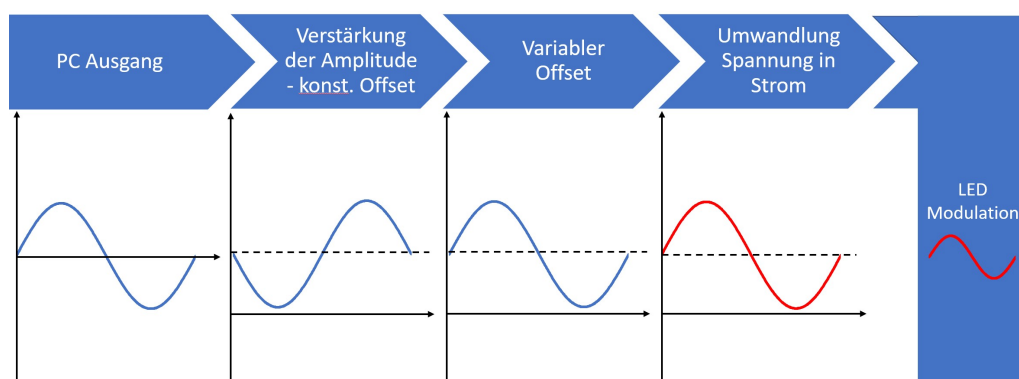


Abbildung 3.1: Signalverarbeitungsschritte  
[1]

In Abbildung 3.1 werden die Verschiedenen Stufen der Signalübertragung veranschaulicht. Da es sich an dieser Stelle um ein nichtlineares System handelt, dürfen die Signalverarbeitungsstufen nicht beliebig vertauscht werden. Eine solche Veränderung der Reihenfolge könnte zur Verfälschung des Signals führen.

Die analoge Signalverarbeitungsschaltung wurde in drei Stufen aufgeteilt. Die OP- Grundschaltungen wurden hierfür so modifiziert, dass Amplitude und Offset variabel einstellbar sind und möglich auftretende Fehler vermieden werden. In Abbildung 3.2 ist die Eingangssignalverarbeitungsstufe illustriert.

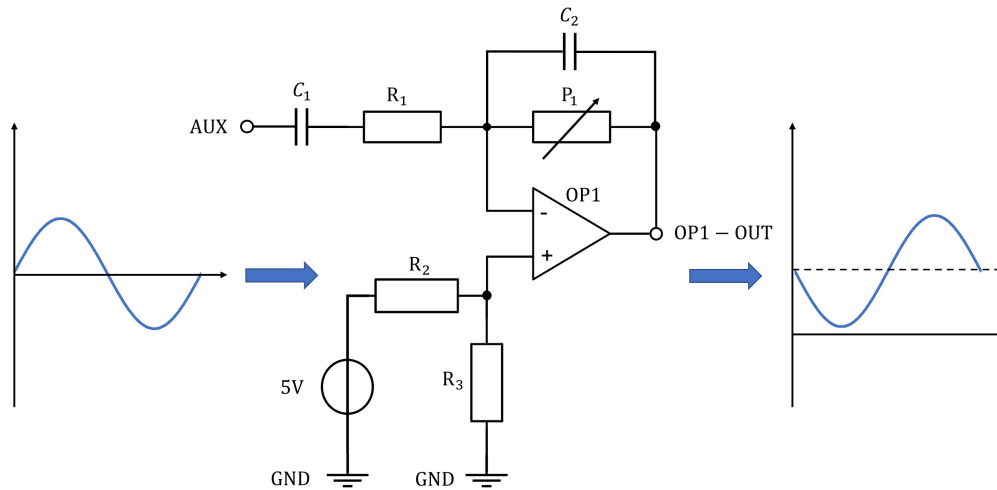


Abbildung 3.2: Eingangssignalverarbeitungsstufe  
[1]

Der Schlüsselfaktor in der Eingangsstufe ist das variable im Rückkopplungspfad des OPs. Hierbei handelt es sich um ein Digitales Potentiometer, welches sich über SPI vom Arduino zur Laufzeit beliebig verändern lässt. Die Schwierigkeit in der Implementierung dieses Bauteiles liegt jedoch darin, dass es keine negativen Spannungen verträgt. Aus dem Datenblatt ist ersichtlich, dass es nur mit Spannungen von  $-0,6V < U < 6V$  sorgfältig arbeiten kann.

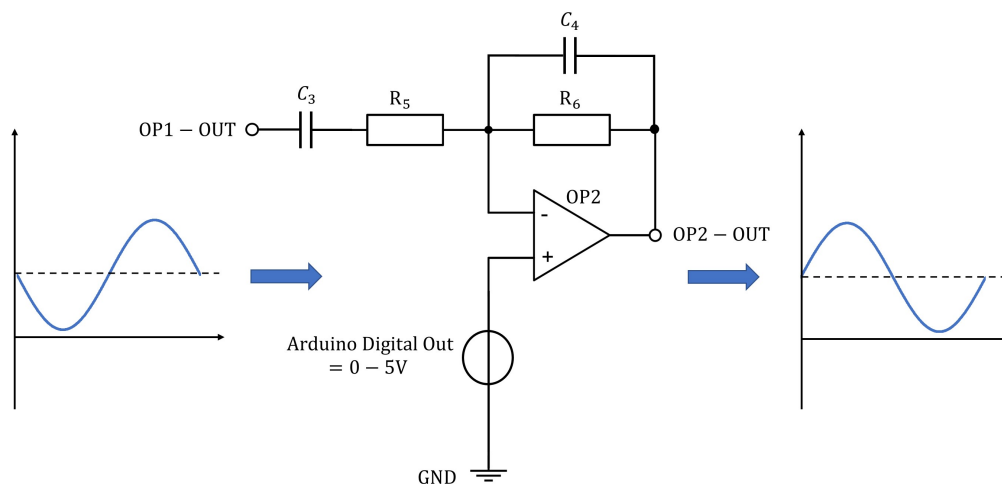


Abbildung 3.3: Signalverarbeitungsstufe 2  
[1]

Die sehr hohe Vorwärtsverstärkung und die differentielle Eingangscharakteristik des OPs können genutzt werden, um eine nahezu ideale spannungsgesteuerte Stromquelle oder einen Spannungs-zu-Strom-Wandler zu realisieren. Es ist jedoch zu beachten, dass die umzuwandelnde Eingangsspannung an den nicht invertierenden Eingang des OPs angelegt wird. Der



invertierende Eingang ist in Rückkopplung mit einem Ende des Widerstands  $R_1$  und der Source des Transistors  $M_1$  verbunden. Dies sorgt dafür, dass zwischen den Eingängen des OPs kein Spannungsunterschied herrscht. Der Ausgang des OPs steuert also das Gate des MOSFETs. Seine hohe Leerlaufverstärkung zwingt das Gate von  $M_1$  auf die erforderliche Spannung. Dadurch wird die Spannung, welche an  $OP2 - OUT = U_{R1}$  anliegt auf die Source des MOSFETs gespiegelt. Draus ergibt sich der Strom zu:

$$I_{R1} = \frac{U_{R1}}{R_1} \quad (3.1)$$

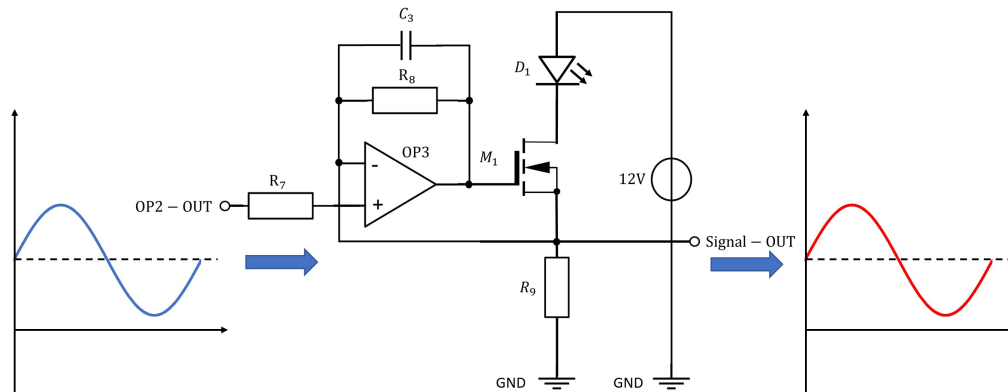


Abbildung 3.4: LED- Treiber als Endstufe  
[1]

Durch die Bekanntheit des Stromflusses durch  $R_1$  kann nun durch kirchhoffsche Regeln auf den Stromfluss in der LED geschlossen werden.[8] Dieser Strom wiederum erzeugt über selbigen Widerstand eine Anhebung des Potentials am - Eingang des OPs. Auf diese Weise versucht der OP seine beiden Eingänge auf das gleiche Potential anzuheben. Beim Leistungswiderstand  $R_9$  handelt es sich um einen sehr kleinen Widerstand, weshalb durch die LED, den MOSFET und den Leistungswiderstand ein sehr hoher Strom fließt. Dieser sorgt für eine hell leuchtende LED zur Übertragung des Signals.

### 3.1.1 Simulation in LT-Spice

### 3.1.2 Platinenlayout in Eagle

Beim Platinenlayout des Senders wurde besonders auf die Trennung von Signal und Leistungswege geachtet. Diese ist in Abbildung 14 mit einer Gelben Linie Mittig des Boards markiert. Im unteren Bereich der Platine befinden sich Leistungswiderstand, LED, MosFet, Spannungswandler und der Operationsverstärker für die Stromquelle. Da in der Schaltung Ströme bis 1,41A fließen können und es somit zu EMV Störungen kommen kann, wurde der Signalweg in den oberen Bereich der Platine gelegt. Zudem wurde die Leiterbahn zwischen Leuchtdiode, Transistor und Leistungswiderstand besonders breit ausgelegt, da hier große Ströme fließen. Auf der Platine wurden für die Operationsverstärker und den Digitalpotentiometer DIP Sockel verlötet. Das ermöglicht nachträglich einen einfachen Austausch der Bauelemente. Da die komplette Schaltung des Visible Light Senders auf die Größe einer kleinen Europlatine entworfen wurde, kam es zu Platzproblemen. Deshalb mussten zwei Leitungen auf der Rückseite der Platine verlegt werden. Diese sind in Abbildung 14 durch die beiden Blauen Linien dargestellt. Um die Schaltung zusätzlich noch störfester zu machen, wurden Freiflächen auf beiden Seiten der Platine mit Masse ausgefüllt. Des

weiteren wurden noch zwei Pins an Masse, und ein Pin an 12V für Messungen und Fehlerüberprüfung angebracht. Der im oberen Teil des Boards eingezeichnete AUX Anschluss wurde nicht, wie in Abbildung 14 gezeigt, an das Board gelötet, sondern am Deckel angebracht. Grund dafür ist das einfachere Anschließen des AUX Kabels. Um den Arduino stabil zu verbauen, wurde er an zwei Seiten befestigt.

### 3.1.3 Thermisches Management

Um die LED als Lichtquelle nutzen zu können, werden hohe Leistungen benötigt. Im hellsten Betrieb der LED fließt ein Strom von 1,41A durch Transistor, LED und Leistungswiderstand. Um der dadurch resultierenden Hitzeentwicklung entgegenzuwirken, werden Transistor und Diode auf einem Kühlkörper montiert. Der Leistungswiderstand liegt nahe am Transistorkühlkörper, darf aber nicht mit dem Kühlkörper verbunden werden. Ohne einen vorhandenen Kühlkörper würde sich der Transistor lt. Datenblatt um 62 °CC/W erhitzen. Da durch den geringen Leistungswiderstand und dem dadurch resultierendem hohen Strom eine Leistung von ca. 5W in Hitze umgewandelt wird, kann dies zur Zerstörung des Bauteils führen. Zur Auslegung der Kühlkörper wurden folgende Berechnungen durchgeführt:

Hierbei ist: –j die Maximale Sperrschichttemperatur in °C des Halbleiters (Herstellerangabe) –A die Umgebungstemperatur in °CC Pv die am zu kühlenden Halbleiter maximal anfallende Verlustleistung in Watt – RJC der Wärmewiderstand zwischen Sperrschicht und Gehäuse (Herstellerangabe) – RCH der Wärmewiderstand zwischen Gehäuse und Kühlkörper – RK der Wärmewiderstand des Kühlkörpers (Herstellerangabe)

#### 3.1.3.1 Kühlkörperberechnung des MOSFETs

Für die Berechnung der Kühlkörper wurden Messungen zur Verlustleistung des Transistors und der LED vorgenommen. Die maximalen Verluste werden bei voller Helligkeit erzeugt.

– Leuchtdiode:  $2,77V \cdot 1,41A = 3,91W$  – Transistor:  $3,6V \cdot 1,41A = 5,08W$

Dadurch berechnet sich der Wärmewiderstand des Transistorkühlkörpers bei gewünschter Chiptemperatur von 55°CC und einer Außentemperatur von 25°CC zu: Ausgehend von der Gleichung 3 wurde ein Kühlkörper von 4,3 °CC/W gewählt.

#### 3.1.3.2 Kühlkörperberechnung der LED

Bei der LED wurde eine geringere Chiptemperatur vorgesehen, da sie sich außen an der Kante des Boards befindet und somit leicht berührbar ist. Zudem wurde bei der Berechnung bewusst überdimensioniert. Dazu wird im folgenden mit einem Wirkungsgrad der LED von 0 %, und somit einer Verlustleistung von 100 % gerechnet, was dem Verhalten eines Widerstandes entspricht. LEDs haben jedoch tatsächlich einen Wirkungsgrad von ca. 40 - 50 %. Der Kühlkörper für die LED berechnet sich wie folgt

### 3.1.4 Planung und Aufbau des Gehäuses

Beim Entwurf des Gehäuses wurde so platzsparend wie möglich gearbeitet, weshalb der fest verbaute Arduino zwischen Leiterplatte und Deckel angebracht wurde. Da sich Leistungswiderstand, Transistor und Leuchtdiode im Betrieb stark erhitzen, wurden die zugehörigen Kühlkörper am Boden mithilfe von Abstandsbolzen befestigt. Zusätzlich wurden im Deckel Freischnitte über den Kühlkörpern vorgenommen um die nach oben steigende Luft möglichst gut abzuleiten. Für die Versorgungsspannung der Schaltung gibt es zwei 4mm Anschlüsse, die oben am Deckel angebracht wurden. Für die Datenübertragung wurde eine AUX Buchse im Deckel verbaut.

## 3.2 Software

### 3.2.1 Automatisierte Amplituden-Regelung

### 3.2.2 Übertragung mit Dream

Das Programm verfügt auf zwei verschiedene Möglichkeiten aufgerufen werden. Zum ersten im Sendemodus und zum zweiten im Empfangsmodus. Zudem bietet Dream umfangreiche Einstellungsoptionen um sowohl Rauschen oder aber auch andere Störungen zu minimieren. Die wichtigsten Parameter für die Korrekte Benutzung werden in den folgenden zwei Kapiteln näher erläutert.

#### 3.2.2.1 Dream Transmitter

Zum Start der Software "Dream" im Übertragungsmodus muss das Programm mit dem Parameter "- t" gestartet werden. Die einfachste Möglichkeit für einen Programmstart mit Parameter bietet eine Verknüpfung, welche wie in Abbildung 15 gezeigt, angepasst wird. Das Zielverzeichnis darf dabei nicht verändert werden! Dieser Schritt muss nur einmal gemacht werden und man kann jederzeit den Übertragungsmodus starten indem man die Dream software über diese geänderte Verknüpfung startet.

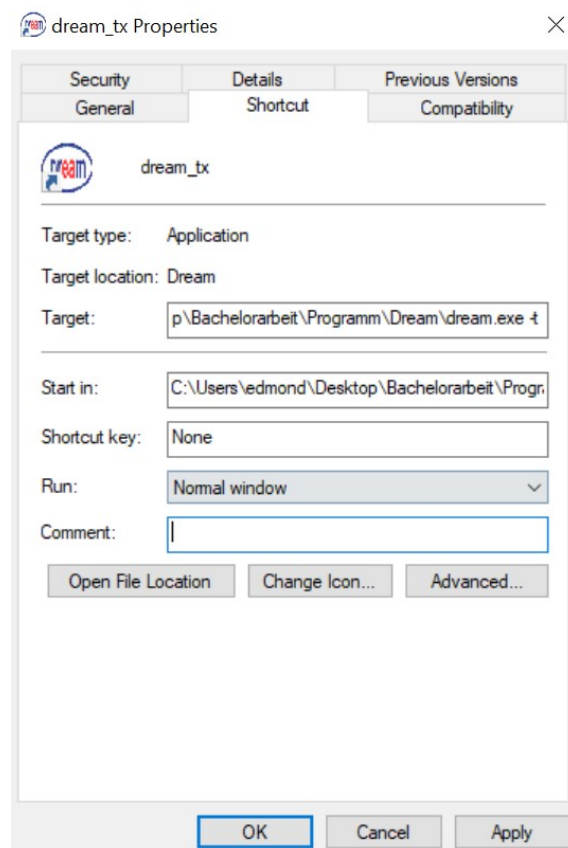


Abbildung 3.5: Modifizierung für den Sendemodus  
[1]



Abbildung 3.6: Internes Audiorouting  
[1]

### 3.2.2.2 Dream Receiver

Der Evaluation Dialog liefert detaillierte Informationen über die empfangenen DRM-Parameter. Hier können Parameter sowie einige Diagramme eingesehen werden. In den folgenden Tabellen, werden für die Übertragung bedeutende Parameter näher erläutert.

Measurements	
<b>DC Frequency Offset</b>	Dieser Offset entspricht der resultierenden Soundkarten-Zwischenfrequenz des Frontends. Diese Frequenz ist nicht auf einen bestimmten Wert beschränkt, sondern nur darauf, dass das DRM-Spektrum vollständig innerhalb der Bandbreite der Soundkarte liegen muss.
<b>Sample Frequency Offset</b>	Offset der Abtastrate des lokalen Computers zum DA-Wandler im Sender.
<b>Doppler / Delay</b>	The Doppler frequency of the channel is estimated for the Wiener filter design of channel estimation in time direction. If linear interpolation is set for channel estimation in time direction, this estimation is not updated. The Doppler frequency is an indication of how fast the channel varies with time. The higher the frequency, the faster the channel changes are. The total delay time of the Power Delay Spectrum (PDS) is estimated from the impulse response estimation derived from the channel estimation. This delay corresponds to the range between the two vertical dashed black lines in the Impulse Response (IR) plot.
<b>I/O Interface LED</b>	This LED shows the current status of the sound card interface. The yellow light shows that the audio output was corrected. Since the sample rate of the transmitter and local computer are different, from time to time the audio buffers will overflow or under run and a correction is necessary. When a correction occurs, a "click" sound can be heard. The red light shows that a buffer was lost in the sound card input stream. This can happen if a thread with a higher priority is at 100 and the Dream software cannot read the provided blocks fast enough. In this case the Dream software will instantly lose the synchronization and has to re-synchronize. Another reason for red light is that the processor is too slow for running the Dream software.
<b>Time Sync Acq LED</b>	This LED shows the state of the timing acquisition (search for the beginning of an OFDM symbol). If the acquisition is done, this LED will stay green.
<b>Frame Sync LED</b>	The DRM frame synchronization status is shown with this LED. This LED is also only active during acquisition state of the Dream receiver. In tracking mode this LED is always green.
<b>FAC CRC LED</b>	This LED shows the Cyclic Redundancy Check (CRC) of the Fast Access Channel (FAC) of DRM. FAC is one of the three logical channels and is always modulated with a 4-QAM. If the FAC CRC check was successful, the receiver changes to tracking mode. The FAC LED is the indication whether the receiver is synchronized to a DRM transmission or not.
<b>SRC CRC LED</b>	This LED shows the CRC check result of the Service Description Channel (SDC) which is one logical channel of the DRM stream. This data is transmitted in approx. 1 second intervals and contains information about station label, audio and data format etc. The error protection is normally lower than the protection of the FAC. Therefore this LED will turn to red earlier than the FAC LED in general.
<b>MSC CRC LED</b>	This LED shows the status of the Main Service Channel (MSC). This channel contains the actual audio and data bits. The LED shows the CRC check of the AAC core decoder. The SBR has a separate CRC, but this status is not shown with this LED. If SBR CRC is wrong but the AAC CRC is ok one can still hear something (of course, the high frequencies are not there in this case). If this LED turns red, interruptions of the audio are heard. The yellow light shows that only one 40 ms audio frame CRC was wrong. This can happen when the audio is not perfectly synchronized.

Parameters	
<b>DRM mode/ bandwidth</b>	<p>In a DRM system, four possible robustness modes are defined to adapt the system to different channel conditions. According to the DRM standard:</p> <p><b>Mode A:</b> Gaussian channels, with minor fading</p> <p><b>Mode B:</b> Time and frequency selective channels, with longer delay spread</p> <p><b>Mode C:</b> As robustness mode B, but with higher Doppler spread</p> <p><b>Mode D:</b> As robustness mode B, but with severe delay and Doppler spread</p> <p>The bandwidth is the gross bandwidth of the current DRM signal.</p>
<b>Interleaver Depth</b>	<p>The symbol interleaver depth can be either short (approx. 400 ms) or long (approx. 2 s). The longer the interleaver the better the channel decoder can correct errors from slow fading signals. But the longer the interleaver length the longer the delay until audio can be heard (after a re-synchronization).</p>
<b>SDC / MSC Mode</b>	<p>Shows the modulation type of the SDC and MSC channel. For the MSC channel, some hierarchical modes are defined which can provide a very strong protected service channel.</p>
<b>Prot. Level (B/A)</b>	<p>The error protection level of the channel coder. For 64-QAM, there are four protection levels defined in the DRM standard. Protection level 0 has the highest protection whereas level 3 has the lowest protection. The letters A and B are the names of the higher and lower protected parts of a DRM block when Unequal Error Protection (UEP) is used. If Equal Error Protection (EEP) is used, only the protection level of part B is valid.</p>
<b>Number of Services</b>	<p>This shows the number of audio and data services transmitted in the DRM stream. The maximum number of streams is four.</p>
<b>Received time - date</b>	<p>This label shows the received time and date in UTC. This information is carried in the SDC channel.</p>

Tabelle 3.2: Unterschrift der Tabelle

Chart	
<b>SNR</b>	<p>Signal to Noise Ratio (SNR) estimation is plotted as a bar and as a value.</p>
<b>Main Plot</b>	<p>Graphical display of different vectors of the DRM decoder.</p>

Tabelle 3.3: Unterschrift der Tabelle

Advanced Settings	
<b>Frequency Interpolation</b>	<p>With these settings the channel estimation method in frequency direction can be selected. The default value uses the most powerful algorithm. Wiener (default) - Wiener interpolation uses estimation of the statistics of the channel to design an optimal filter for noise reduction. Linear - Simple linear interpolation method to get the channel estimate. The real and imaginary parts of the estimated channel at the pilot positions are linearly interpolated. This algorithm causes the lowest CPU load but performs much worse than the Wiener interpolation at low SNR's. DFT Zero Pad: - Channel estimation method for the frequency direction using Discrete Fourier Transformation (DFT) to transform the channel estimation at the pilot positions to the time domain. A zero padding is applied to get a higher resolution in the frequency domain -&gt; estimates at the data cells. This algorithm is very speed efficient but has problems at the edges of the OFDM spectrum due to the leakage effect.</p>
<b>Time Interpolation</b>	<p>With these settings the channel estimation method in time direction can be selected. The default value uses the most powerful algorithm. Wiener (default) - Wiener interpolation uses estimation of the statistics of the channel to design an optimal filter for noise reduction. Linear - Simple linear interpolation method to get the channel estimate. The real and imaginary parts of the estimated channel at the pilot positions are interpolated linearly. This algorithm causes the lowest CPU load and the audio is decoded more quickly, but in general it performs worse than the Wiener interpolation especially at low SNR's.</p>
<b>Time Sync Tracking</b>	<p>With these settings the time synchronization tracking methods can be selected. Guard Energy (default) - This algorithm utilizes the estimation of the impulse response and tries to maximize the energy in the guard-interval to set the correct timing. First Peak - This algorithm searches for the first peak in the estimated impulse response and moves this peak to the beginning of the guard-interval (timing tracking algorithm)</p>
<b>Flip Input Spectrum</b>	<p>Checking this box will flip or invert the input spectrum. This is necessary if the mixer in the front-end uses the lower side band.</p>
<b>Mute Audio</b>	<p>The audio can be muted by checking this box. The reaction of checking or unchecking this box is delayed by approx. 1 second due to the audio buffers.</p>
<b>MLC, Number of Iterations</b>	<p>In DRM a multilevel channel coder is used. With this code it is possible to iterate the decoding process in the decoder to improve the decoding result. The more iterations are used the better the result will be. But switching to more iterations will increase the CPU load. Simulations showed that the first iteration (number of iterations = 1) gives the most improvement (approx. 1.5 dB at a BER of 10<sup>-4</sup> on a Gaussian channel, Mode A, 10 kHz bandwidth). The improvement of the second iteration (number of iterations = 2) will be as small as 0.3 dB. The recommended number of iterations given in the DRM standard is one iteration (default value: number of iterations = 1). The selection is saved in the Dream.ini file.</p>
<b>Log File</b>	<p>Checking this box causes Dream to write two kinds of log files about the current reception of an audio service using AAC source coding, a standard and a long log file. Both files are written to the directory where the Dream application is located. The standard log file Dream-Log.txt is compatible to the log file created by the "DRM Software Radio". Each minute information about the average SNR, number of correct decoded FAC and number of correct decoded MSC blocks is recorded. The header of each log section also contains frequency, modulation, bit rate, sample rate, bandwidth. This file format is</p>





## **4. Modell**

### **4.1 Testaufbau zur Funktionsprüfung**

### **4.2 Vorzeige Modell**



## **5. Ergebnis**



## 6. Zusammenfassung

Hier sind einige Beispiele.

### 6.0.1 Abkürzung

So verwende ich eine Abkürzung Donald John Trump (DJT) und so erneut DJT.

### 6.0.2 Symbol

So füge ich ein Symbol ein  $h_s$ .

### 6.0.3 Bild

So füge ich ein Bild ein:



Abbildung 6.1: Bild Unterschrift  
[?]



Abbildung 6.2: Bild Unterschrift1 [?]

So beziehe ich mich auf das Bild 6.1

6.0.4 Formel

$$F_A \geq 2 \cdot F_S$$

(6.1)

So beziehe ich mich auf die Formel 6.1

6.0.5 Tabelle

Stimmen für SleepyJoe	davon gefaked	Sieger
80000000	80000000	Donald

Tabelle 6.1: Unterschrift der Tabelle

So beziehe ich mich auf die Tabelle 6.1

6.0.6 Referenz und Zitat

Referenzierung auf ein Kapitel ?? 5 Zitat einfügen

[?]. [?] So verwende ich eine Online Quelle [1]

# Online-Referenzen

- [1] Eigene Darstellung
  
- [2] What is Arduino?  
<https://www.arduino.cc/en/Guide/Introduction>  
Abgerufen 2021-02-09
  
- [3] LT-Spice  
<https://en.wikipedia.org/wiki/LTspice>  
Abgerufen 2021-02-13
  
- [4] DREAM  
<http://www.mynetcologne.de/~nc-keilje/drm/dream/index.htm>  
Abgerufen 2021-03-10
  
- [5] Was ist Eagle?  
<https://www.autodesk.de/products/eagle/overview>  
Abgerufen 2021-03-10
  
- [6] Eagle  
[https://de.wikipedia.org/wiki/Eagle\\_\(Software\)](https://de.wikipedia.org/wiki/Eagle_(Software))  
Abgerufen 2021-03-10





# Abbildungsverzeichnis

2.1	Hörbereich des menschlichen Ohres . . . . .	4
2.2	Abtastung eines Signals . . . . .	5
2.3	Unterabtastung eines Signals . . . . .	5
2.4	Strahlungserzeugung in der LED am pn-Übergang . . . . .	10
2.5	Diodenkennlinie . . . . .	11
2.6	Flussspannungen von LEDs verschiedener Farben . . . . .	12
2.7	Kennlinie einer Diode – Lichtleistung zu fließendem Strom . . . . .	12
2.8	Operationsverstärker Anschlusschema . . . . .	13
2.9	Invertierender Operationsverstärker . . . . .	14
2.10	Nichtinvertierender Operationsverstärker . . . . .	15
2.11	Operationsverstärker . . . . .	15
2.12	Aufbau des MCP42010 . . . . .	17
2.13	Beschaltung und erzeugte Spannung des LT1054 . . . . .	18
2.14	Arduino IDE . . . . .	20
3.1	Signalverarbeitungsschritte . . . . .	21
3.2	Eingangssignalverarbeitungsstufe . . . . .	22
3.3	Signalverarbeitungsstufe 2 . . . . .	22
3.4	LED- Treiber als Endstufe . . . . .	23
3.5	Modifizierung für den Sendemodus . . . . .	25
3.6	Internes Audiorouting . . . . .	26
6.1	Titel für das Abbildungsverzeichnis . . . . .	35
6.2	Titel für das Abbildungsverzeichnis1 . . . . .	36



# Tabellenverzeichnis

2.1	Anschlussinformation und Pinbelegung des MCP42010 . . . . .	18
3.1	Unterschrift der Tabelle . . . . .	27
3.2	Unterschrift der Tabelle . . . . .	28
3.3	Unterschrift der Tabelle . . . . .	28
3.4	Unterschrift der Tabelle . . . . .	29
6.1	Unterschrift der Tabelle . . . . .	36



# Symbolverzeichnis

$h_s$     Test [s]



# Abkürzungsverzeichnis

DJT	Donald John Trump
DRM	Digital Radio Mondiale
IC	Integrated Circuit
IDE	Integrated Developer Environment
LED	Light Emitting Diode
MOSFET	Metall-Oxid-Halbleiter-Feldeffekttransistor
SPI	Serial Peripheral Interface
VLC	Visible light communication