

BERLINER HOCHSCHULE FÜR TECHNIK

FACHBEREICH ELEKTROTECHNIK

BACHELORARBEIT

Realitätsnahe, technische Nachbildung natürlicher Lichtquellen

am Beispiel einer Wachskerze

Verfasser: Julian Konstantin Kurz
Matrikelnummer: 895983
Studiengang: Elektrotechnik (B.Eng.)

Erstprüfer: Prof. Dr.-Ing. Tobias Merkel
Zweitprüfer: M. Eng. Gordon Dyballa

Abgabedatum: 18.06.2025

Inhaltsverzeichnis

1 Einleitung	4
2 Entwicklung des Kerzenkopfes	6
2.1 Auswahl der Leuchtmittel	6
2.1.1 Übersicht	6
2.1.2 Realisierung mit den LED-Filamenten	8
2.1.3 Realisierung mit SMD-LEDs	12
2.1.4 Wirkungsgrad	13
2.1.5 Lichtqualität	14
2.1.6 Fazit	15
2.2 Platine des Kerzenkopfs	15
3 Das Flachbandkabel	17
4 Energiemanagement	20
4.1 Auswahl der Batterie	20
4.1.1 Lastfälle im Sollbetrieb	21
4.1.2 Lastfälle im Maximalbetrieb	21
4.1.3 Lastfälle im Grenzbetrieb	22
4.1.4 Vergleich mit echten Kerzen	22
4.2 Spannungsversorgung und Laderegler	22
5 Ansteuerung der LEDs	23
5.1 Einführung Ansteuerung	23
5.2 Nutzung des MOSFETS	26
5.2.1 Hardwarerealisierung der Ansteuerung	27
5.2.2 Wahl der PWM-Auflösung und -Frequenz	31
5.3 Auswahl der Betriebsart	31
5.3.1 Übersicht der Betriebsarten	31
5.3.2 Option 1: Schaltbetrieb	32
5.3.3 Option 2: Linearbetrieb	32
5.3.4 Option 3: Soft-Switching-Betrieb	33
5.3.5 Vergleich der Betriebsarten	33
5.3.6 Auslegung des Tiefpassfilters	33
5.3.7 Fazit	34
5.4 Strommessung	35
5.4.1 Einleitung	35
5.4.2 Option 1: Abtastung des gepulsten Stroms	38
5.4.3 Option 2: Mittelwertmessung über einen Tiefpass	39
5.5 Datenaufbereitung	45
5.5.1 Light Source Extractor	45
5.5.2 Videoimport	45
5.5.3 Linearisierung des Farbraums von Videosignalen	46
5.5.4 Pipeline 1: Radiale Flammen	48
5.5.5 Pipeline 2: Gerichtete Flammen	49
5.5.6 Normierung und Segmentierung	51
5.5.7 Export als C-Datei	52

6 Softwareimplementierung auf dem STM32	54
6.1 Unendliche Aneinanderreihung von zufälligen Clips	54
6.2 Crossfading	58
6.3 Simulation	60
6.3.1 Upsampling und Auflösungsvergrößerung	62
6.3.2 Zufall	63
7 Ergebnis	64
8 Ausblick	65

Abkürzungsverzeichnis

ADC Analog-Digital-Converter (Analog-Digital-Wandler). 35

CCR Capture-Compare-Register. 40

COG Chip-on-Glass. 7

CRI Color Rendering Index (Farbwiedergabeindex). 14

EEPROM Electrically Erasable Programmable Read-Only Memory. 65

FFC Flexible Flat Cable (Flexkabel). 17

GbR Gesellschaft bürgerlichen Rechts. 4, 6

INA Instrumentenverstärker. 35

LDO Low Dropout Regulator. 16

LED Light Emitting Diode. 4, 6–8, 10, 12–15, 65

MCU Microcontroller Unit. 40

MOSFET Metal-Oxide-Semiconductor Field-Effect Transistor. 16

MUX Multiplexer. 35

PWM Pulse Width Modulation. 15, 34

ROI Region of Interest. 48, 49, 66

SMD Surface-Mounted Device. 7, 12–15

sRGB Standard Red Green Blue. 47

USB Universal Serial Bus. 25

1 Einleitung

Die Rethink Lights [Gesellschaft bürgerlichen Rechts \(GbR\)](#) realisiert professionelle und innovative Lichtinstallationen für Filmproduktionen, darunter Netflix-Serien wie „Die Kaiserin“ und Projekte der Babelsberg Studios sowie weitere Beleuchtungsprojekte.

Im Jahr 2022 bat mich Mads Gutowski, Gründer der Rethink Lights [GbR](#), eine digitale Kerze zu entwickeln, weil die am Markt verfügbaren Modelle den Anforderungen der Filmbranche nicht genügten.

Die auf dem Markt verfügbaren [Light Emitting Diode \(LED\)](#)-Kerzen schlechter Qualität flackern sehr stark, was besonders beim Film ein Problem ist. Denn eine Kamera filmt üblicherweise zwischen 24 und 120 FPS. Dies erklärt sich aus der Tatsache, dass die Helligkeit solcher Kerzen durch schnelles Ein- und Ausschalten (PWM) realisiert wird. Betrachtet man eine solche [LED](#) in Zeitlupe, ähnelt ihr Leuchtmuster dem einer Lichtquelle, die ständig ein- und ausgeschaltet wird. Erfolgt dieses Schalten schnell genug, nimmt das menschliche Auge es nur noch als kontinuierliche Helligkeitsänderung wahr.

Niederfrequentes Flackern wird bei einer Video- oder Fotoaufnahme sichtbar oder manchmal sogar von einem Menschen wahrgenommen. Weiterhin sind solche [LED](#)-Kerzen in ihrer Helligkeitsauflösung sehr schlecht. Sie haben mitunter nur 16 Helligkeitsstufen, und selbst bei 256 Stufen würde man den Übergang von einer Stufe in die nächste als ruckartig empfinden.

Ein weiterer Nachteil der billigen Modelle ist die schlechte Lichtqualität. Deren Lichtspektrum ist oft sehr unregelmäßig. Zudem haben die meisten [LED](#)-Kerzen nur eine [LED](#), die sich räumlich nicht bewegt und dadurch in alle Richtungen gleichermaßen leuchtet. Die Flamme einer echten Kerze schwankt jedoch hin und her.

Das größte Problem dieser digitalen Kerzen liegt in ihren simplen Leuchtmustern. Diese entstehen entweder durch einfache, sich wiederholende Lichtmuster oder durch Aneinanderreihung zufälliger Helligkeitswerte. Beide Ansätze können das Flackern einer echten Kerze nicht überzeugend wiedergeben. Eine echte Kerze hat ein kontinuierliches Licht und ihr eigenes, sich nicht wiederholendes Lichtmuster. Ein weiteres Problem ist, dass diese [LED](#)-Kerzen nur mit einem Standard-Modus ausgestattet sind.

Das Thema dieser Bachelorarbeit befasst sich mit der Entwicklung und Realisierung einer digitalen Kerze, die die Anforderungen der Filmindustrie erfüllt – nämlich die realitätsnahe Nachbildung der Flamme einer Wachskerze.

Nachfolgend die wichtigsten Kriterien:

- Nachahmung eines 3D-Effekts
- Hochwertiges Lichtspektrum
- Lange Betriebslaufzeit
- Möglichst authentisches Lichtmuster einer Kerzenflamme
- Möglichst hohe Zeit- und Helligkeitsauflösung
- Kompaktes Design

Die Arbeit umfasst die Analyse möglicher Lösungen, die Auswahl geeigneter technischer Komponenten sowie das Hard- und Software-Design des Ansteuersystems. Das folgende Foto zeigt einen Prototypen der im Zuge der Bachelorarbeit entwickelten digitalen Kerze.



Abbildung 1: Prototyp der digitalen Kerze

Für die sprachliche und technische Ausarbeitung dieser Arbeit kamen unterstützend die KI-Dienste Writefull (in Overleaf) sowie GPT-4o und GPT-3.5 (o3) von OpenAI zum Einsatz. Sie wurden genutzt zur Korrektur und Optimierung von Texten, zur Analyse und Erzeugung von L^AT_EX-Code, zur Generierung technischer Diagramme sowie zur Überprüfung von Programmcode.

Zur Nachvollziehbarkeit aller Ergebnisse dieser Arbeit werden sämtliche Roh- und Begleitdaten, darunter Python-Skripte, Premiere-Pro-Projekte, Photoshop-Dateien und weitere, in einem über einen Link zugänglichen Git-Repository gepflegt.

Das Repository wird auch nach Abgabe dieser Bachelorarbeit unabhängig weiter aktualisiert.

Direkter Zugriff: github.com/JulianKKurz/candle

2 Entwicklung des Kerzenkopfes

2.1 Auswahl der Leuchtmittel

2.1.1 Übersicht

Am Anfang steht die Analyse der in Frage kommenden Leuchtmittel. Die Rethink Lights GbR hat zu Beginn des Projekts eine Lösung mit standardisierten LED-Filamenten vorgeschlagen. Im folgenden Bild sieht man solche LED-Filamente.

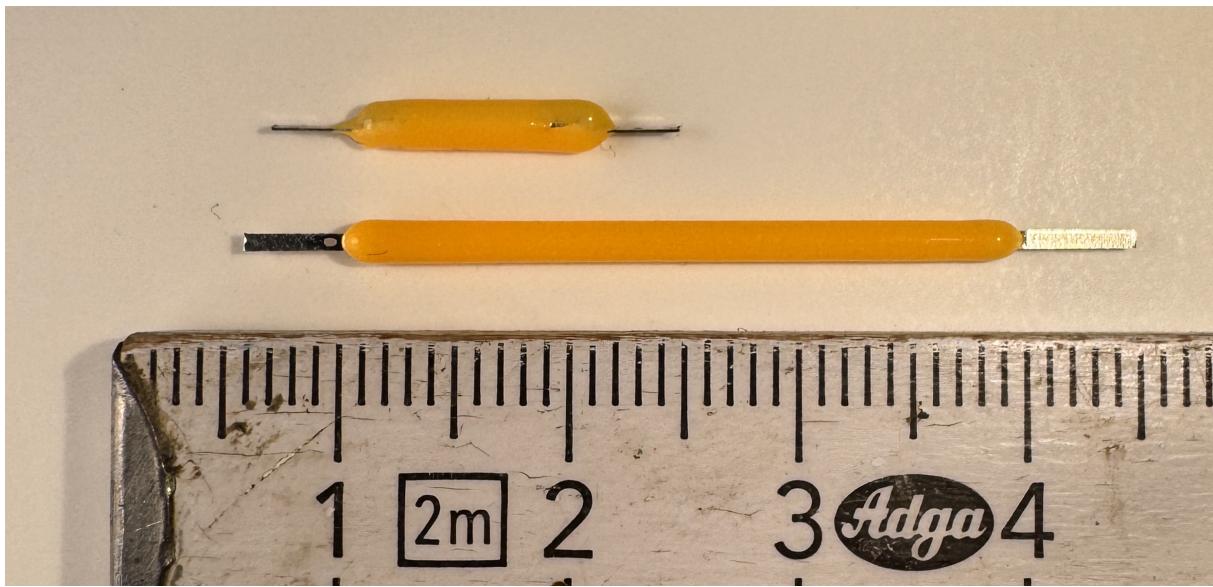


Abbildung 2: LED-Filamente

Nachfolgend ein Kerzenkopf, den die Rethink Lights GbR mit LED-Filamenten und Epoxidharz gebaut hat.

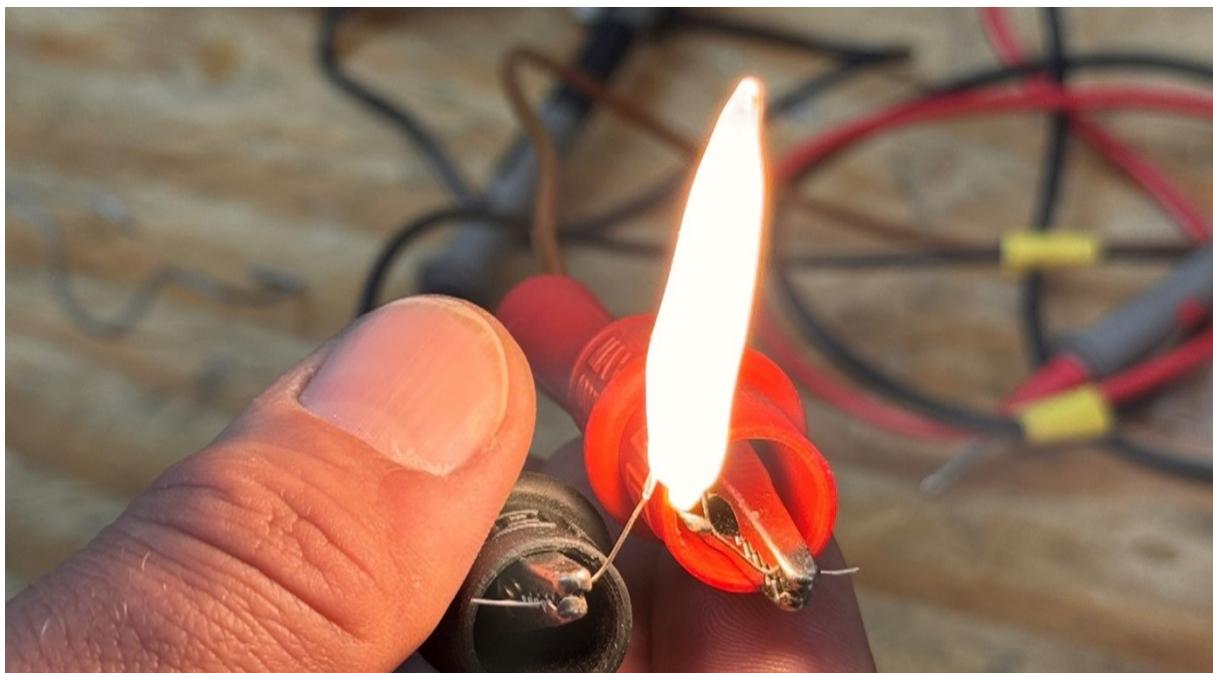


Abbildung 3: Kerzenkopf mit LED-Filamenten, vergossen in Epoxidharz

Dabei sind zwei kurze knapp 2 cm lange **LED**-Filamente an eine lange 4 cm Filament-**LED** gelötet und mit weißem Epoxidharz überzogen worden. Dennoch ist es angebracht, naheliegende Alternativen zu vergleichen. Folgende Kriterien sollten erfüllt werden:

1. Kostengünstig
2. Hohe Lichtwiedergabequalität
3. Die geometrische Form soll der Kerzenflamme ähneln
4. Die Möglichkeit die Bewegung der Flamme nachzuahmen
5. Guter Wirkungsgrad
6. Einfache Verarbeitung
7. 3D-Lichteffekt

Die erste Option wäre die Lösung mit standardisierten **LED**-Filamenten. Die zweite Option wäre eine transparente FPC (flexible Leiterplatte) mit 2×16 **Surface-Mounted Device (SMD)-LEDs** der Bauform 0402 mit einer leichten Phosphorschicht, wie auf dem folgenden Bild zu erkennen ist [1].

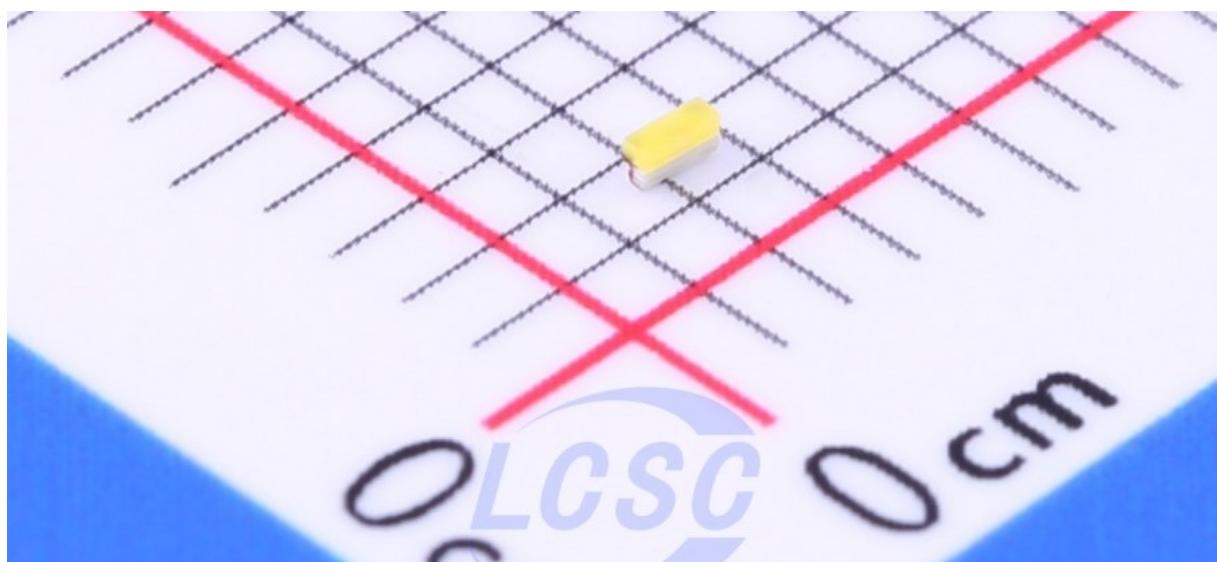


Abbildung 4: SMD-**LED** der Bauform 0402 vom Typ Everlight 16/916/T1D/AP1Q2QY/3T

Als dritte Option wird eine individuell produzierte **Chip-on-Glass (COG)** **LED**-Matrix betrachtet. Man erspart sich einen Großteil der Kosten für die Montage in Handarbeit. Diese Option entfällt sofort wegen der hohen Entwicklungskosten.

2.1.2 Realisierung mit den LED-Filamenten

In der folgenden Grafik ist ein **LED-Filament** mit der entfernten Phosphorschicht zu sehen. Darunter liegen die in Reihe verschalteten **LEDs**, die auf einem Saphirglasträger aufgebracht sind.



Abbildung 5: LED-Filament ohne Phosphorschicht

Folgendes Foto zeigt ein aufgeschnittenes 40 mm langes und 2 mm breites **LED-Filament**, welches im Kerzenkopf verbaut ist.

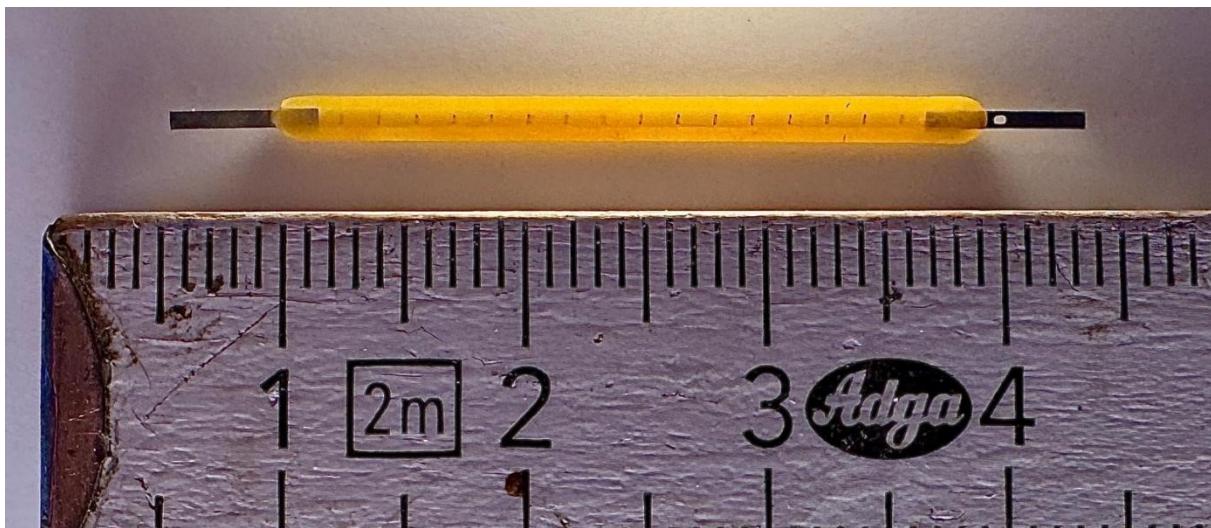


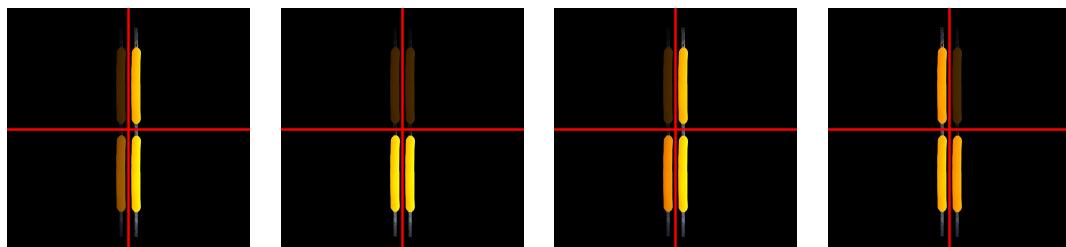
Abbildung 6: Aufgeschnittenes LED-Filament

Man sieht, dass 16 **LED-Chips** auf einer Länge von etwa 24 mm angeordnet sind, was einen Abstand von 1,5 mm bedeutet. Das ist ein typischer Abstand für Filamente. Bei kleiner Stückzahl liegt der Preis bei 0,25 € und bei großer Stückzahl bei etwa 0,09 €.

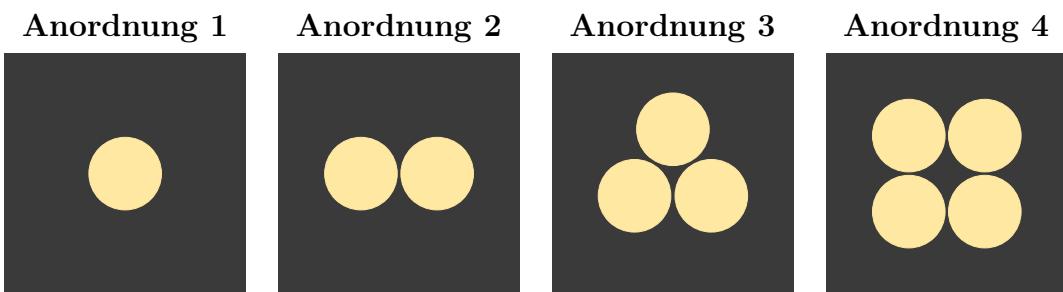
Die Geometrie des Kerzenkopfes wird durch die Anordnung der Filamente bestimmt. Das Ziel ist, die Bewegung der Kerzenflamme durch LED-Filamente zu realisieren. In einem Gedankenexperiment anhand von Bildausschnitten der Kerzenflammen wird versucht, dieses Konzept zu evaluieren.

Tabelle 1: Wie kann sich die Flamme der Kerze bewegen?**Tabelle 2:** Jedes Bild aus der Tabelle wird nun in 4 gleiche Segmente aufgeteilt.

Die Begründung für die Aufteilung des Bildes in vier Segmente ist folgende: Durch die Filamente lässt sich die vertikale und horizontale Bewegung der Kerzenflamme darstellen. Dies ist zunächst eine zweidimensionale Anordnung. Nachfolgend die Überlegungen zur Dreidimensionalität.

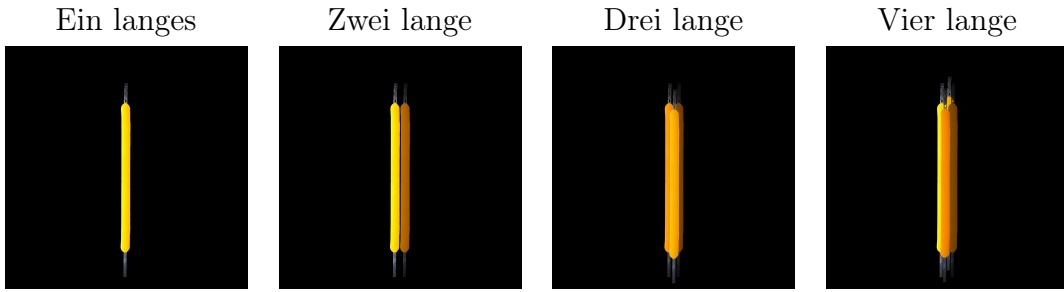
Tabelle 3: Nun wird jedes Segment durch ein kurzes LED-Filament repräsentiert.

Und wie wird daraus eine räumliche Erscheinung? In der folgenden Tabelle sieht man die Anordnungs-Möglichkeiten.

Tabelle 4: Filament-Anordnungen und 3D-Eindruck

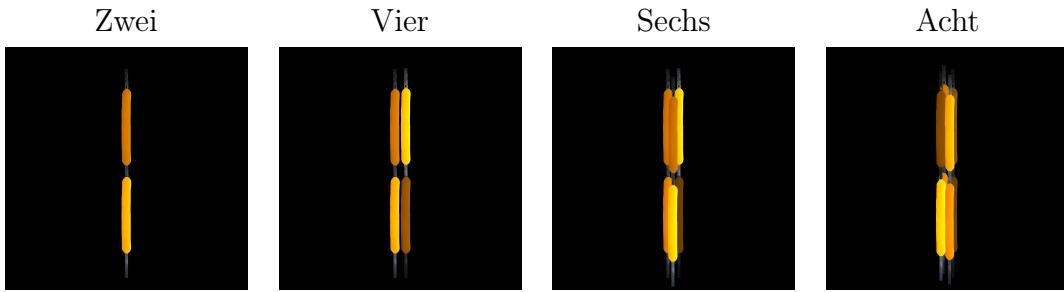
Anordnung 1 stellt nur eine Linie dar. *Anordnung 2* würde von der Seite wie Anordnung 1 aussehen. *Anordnung 3* würde die 3D-Bedingung erfüllen. *Anordnung 4* ermöglicht die Darstellung von vier Pixeln. Ein zuätzliches Argument wäre eine größere Lichtausbeute und die Möglichkeit, es gut auf der Vorder- und Rückseite einer Platine zu verlöten. Realisierung mit langen Filamenten:

Tabelle 5: Lange Filamente – Varianten



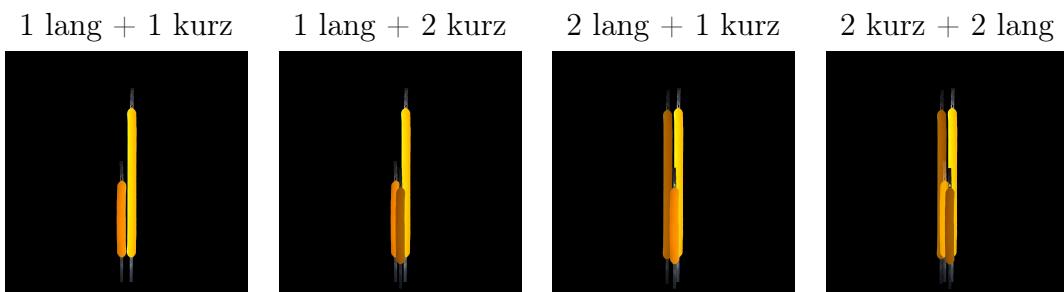
Realisierung mit kurzen Filamenten:

Tabelle 6: Kurze Filamente übereinander – Varianten



Acht kurze Filamente würden die einzelnen Sektoren voll abbilden, weil sich diese komplett ausschalten ließen. Aber der horizontale Spalt zwischen den Filamenten wäre deutlich sichtbar. Auf dem Foto des aufgeschnittenen Filaments auf dem Lineal wird sichtbar, dass es ab dem Fluoreszenzmantel bis zur ersten LED etwa 3 mm sind. Dicht aneinander gereiht ergibt es einen Spalt von mindestens 6 mm. Dieses Problem ist mit einem speziell entwickelten Matrix-Filament zu lösen, was die Kosten des Projekts übersteigen würde. Realisierung mit einer Kombination aus langen und kurzen Filamenten:

Tabelle 7: Kombination lange / kurze Filamente

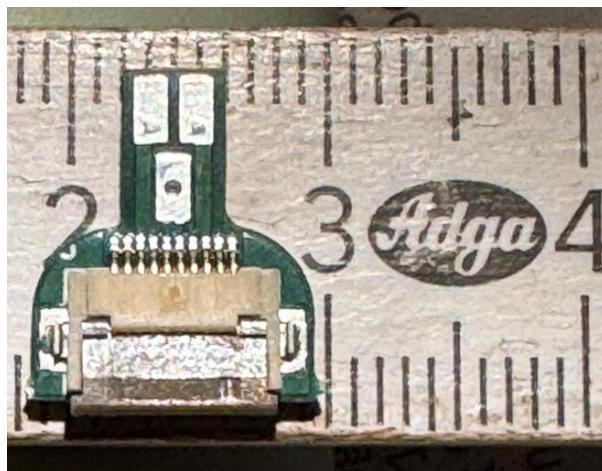


Das Problem mit dem Spalt ist durch die Kombination von kurzen und zwei langen Filamenten lösbar. Das eine Filament befindet sich räumlich mittig vor den beiden langen

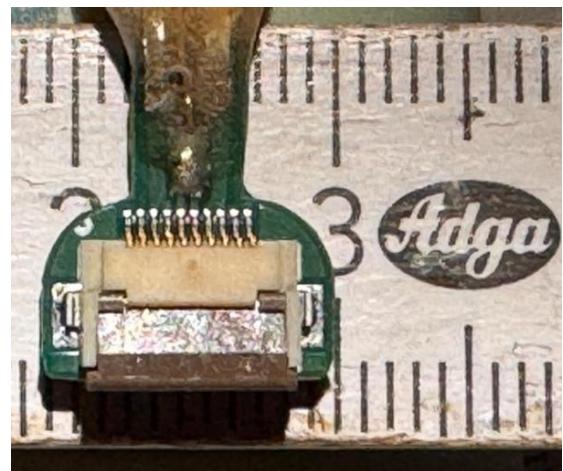
Filamenten, das zweite kurze Filament mittig hinter den beiden langen Filamenten. Auch können alle Kombinationen der simulierten Kerzenflamme mit ihren vier Segmenten durch diese Anordnung dargestellt werden.

Ein zusätzliches Argument für die Begrenzung auf vier Filamente ist der Platzbedarf und der steigende technische Mehraufwand bei mehr Filamenten. In einem der natürlichen Kerzenflamme ähnlichem Kerzenkopf kann man rein räumlich höchstens vier Filamente unterbringen. Der Platzbedarf auf der Platine pro Filament ist 2 mm. Bei vier Filamenten kommen wir auf die Gesamtbreite des angedachten Dochtes von 4 mm.

Die folgenden Grafiken veranschaulichen das Platzproblem.



(a) Kerzenkopfplatine ohne Filamente



(b) Final vergossene Kerzenkopfplatine in Epoxidharz

Tabelle 8: Platzbedarf im Kerzenkopf

Dabei kämen jeweils zwei auf die Vorder- und Rückseite. Diese Lösung hat jedoch den Nachteil, dass sich ein oberes Segment nie alleine ansteuern lässt. Zudem erfüllen vier Filamente das Kriterium der Lichtstärke voll.

Weitere Filamente würden den Aufwand deutlich erhöhen und zudem höhere Kosten und größeren Platzbedarf bedeuten. Ein weiterer Vorteil dieser Anordnung ist, dass oben die zwei langen Filamente herausragen und einen spitzen Kerzenkopf erlauben. Die gewählte Lösung stellt einen ausgewogenen Kompromiss zwischen der maximalen Anzahl steuerbarer Leuchtkörper, einem kompakten Formfaktor, vertretbarem technischem Aufwand und angemessenen Kosten dar.

2.1.3 Realisierung mit SMD-LEDs

Nun wird versucht, die gewählte Filamentlösung mit den **SMD-LEDs** maßgetreu zu realisieren. Dafür müssen die **SMD-LEDs** auf einer Fläche von 4×30 mm untergebracht werden. Es wird angenommen, dass zwei kurze und zwei lange Filament-**LEDs** verbaut werden. Geht man davon aus, dass ein langes Filament 16 **LEDs** hat und ein kurzes 8 **LEDs**, so ergibt es zusammen 48 **LEDs** bei beidseitiger Bestückung. Diese werden auf einer flexiblen transparenten Leiterplatte (FPC) montiert.

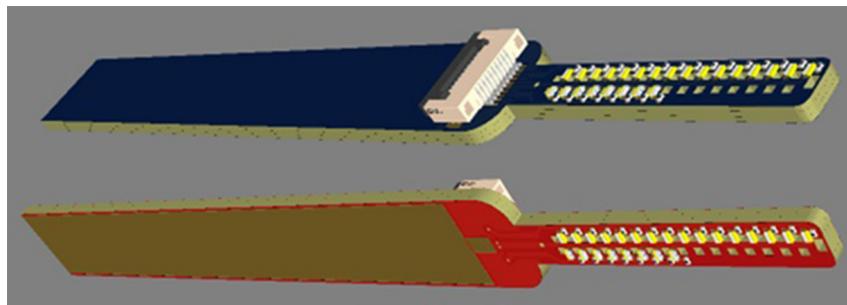


Abbildung 7: Vorder- und Rückseite der Kopfplatine mit SMD-LEDs (3D-Modell)

Bauteile und Leiterbahnen sind nicht lichtdurchlässig. Der Effekt auf die Transparenz der Platine wird nachfolgend analysiert.

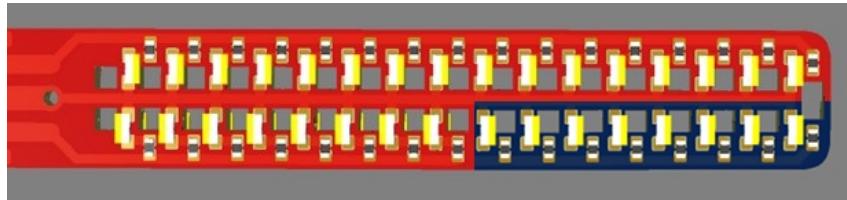


Abbildung 8: Overlay beider Seiten (relevanter Ausschnitt)

Zur Veranschaulichung wird hier eine unrealistische KI-generierte Grafik eingeblendet.



Abbildung 9: KI-generiertes Stimmungsbild

Nun wird der lichtundurchlässige Anteil der Platine ermittelt. Dafür wird dieser in der oberen Hälfte der Platine berechnet. Die Anteile werden berechnet, indem man die Pixel der jeweiligen Farbbereiche zählt. Die Bauteile, Leiterbahnen und Pads beanspruchen ca. 35 % der Platinenfläche und reduzieren die Lichtdurchlässigkeit um etwa 35 %.

Im Vergleich dazu sind **LED**-Filamente nahezu vollständig transparent. Außerdem habe sie einen Abstrahlwinkel von 360 Grad. **SMD-LEDs** hingegen weisen typischerweise einen Abstrahlwinkel von etwa 130 Grad auf.

Wie schon erwähnt werden die **LEDs** in dem **LED**-Filament seriell betrieben. Hier ist ein serieller Betrieb unrealistisch, weil man bei 3 Volt Nennspannung beim längsten Strang

auf 48 Volt kommen müsste. Dadurch erfordert dieses Design eine parallele Ansteuerung mit jeweils einem Vorwiderstand pro **LED**.

2.1.4 Wirkungsgrad

Die Lichtausbeute von LED-Filamenten wird im Datenblatt der Pur-LED GmbH für eine 24-V-COB-Filament-Serie mit 83 lm/W angegeben [2]. Dieser Wert kann herstellerübergreifend als vergleichbar betrachtet werden.

Für die verwendete 0402-**SMD-LED** (Everlight 16-916/T1D-AP1Q2QY/3T) ergibt sich die maximale Lichtstärke zu 112 mcd bei einem Betriebsstrom von 5 mA. Die typische Nennspannung liegt bei 3,0 V, woraus sich eine elektrische Leistung von 15 mW ergibt.

Die Umrechnung von Lichtstärke (cd) und Raumwinkel (sr) in Lichtstrom (lm) erfolgt nach folgender Beziehung [3, 4]:

$$\text{Lumen} = \text{Candela} \cdot \text{Raumwinkel} \quad (1)$$

Der Raumwinkel ergibt sich aus dem Abstrahlwinkel von 130° zu:

$$\Omega = 2\pi \cdot (1 - \cos(65^\circ)) \approx 3,63 \text{ sr} \quad (2)$$

Somit ergibt sich der Lichtstrom zu:

$$\Phi = 0,112 \text{ cd} \cdot 3,63 \text{ sr} \approx 0,41 \text{ lm} \quad (3)$$

Daraus folgt eine Lichtausbeute von:

$$\eta = \frac{0,41 \text{ lm}}{0,015 \text{ W}} \approx 27 \text{ lm/W} \quad (4)$$

Vergleicht man beide Werte, ergibt sich, dass die **LED**-Filamentlösung mindestens den dreifachen Lichtstrom pro Watt liefert. Somit ist die **SMD-LED**-Lösung auch in diesem Fall unbrauchbar.

2.1.5 Lichtqualität

Ein Lichtinstallationsunternehmen stellt auf ihrer Seite das Datenblatt eines flexiblen **LED**-Filaments zur Verfügung, welches wir als Beispiel anführen. Der folgende Plot stellt die Spektren des **LED**-Filaments, einer Kerze und der **SMD-LED** gegenüber. Diese Daten wurden zuvor aus den Datenblättern digitalisiert, aufbereitet und geplottet. Außerdem wurden die Spektren analysiert und der **Color Rendering Index (Farbwiedergabeindex) (CRI)** berechnet. Ein CRI von 100 % bedeutet, dass die Lichtquelle die Farben genauso perfekt wiedergibt wie z. B. das natürliche Sonnenlicht.

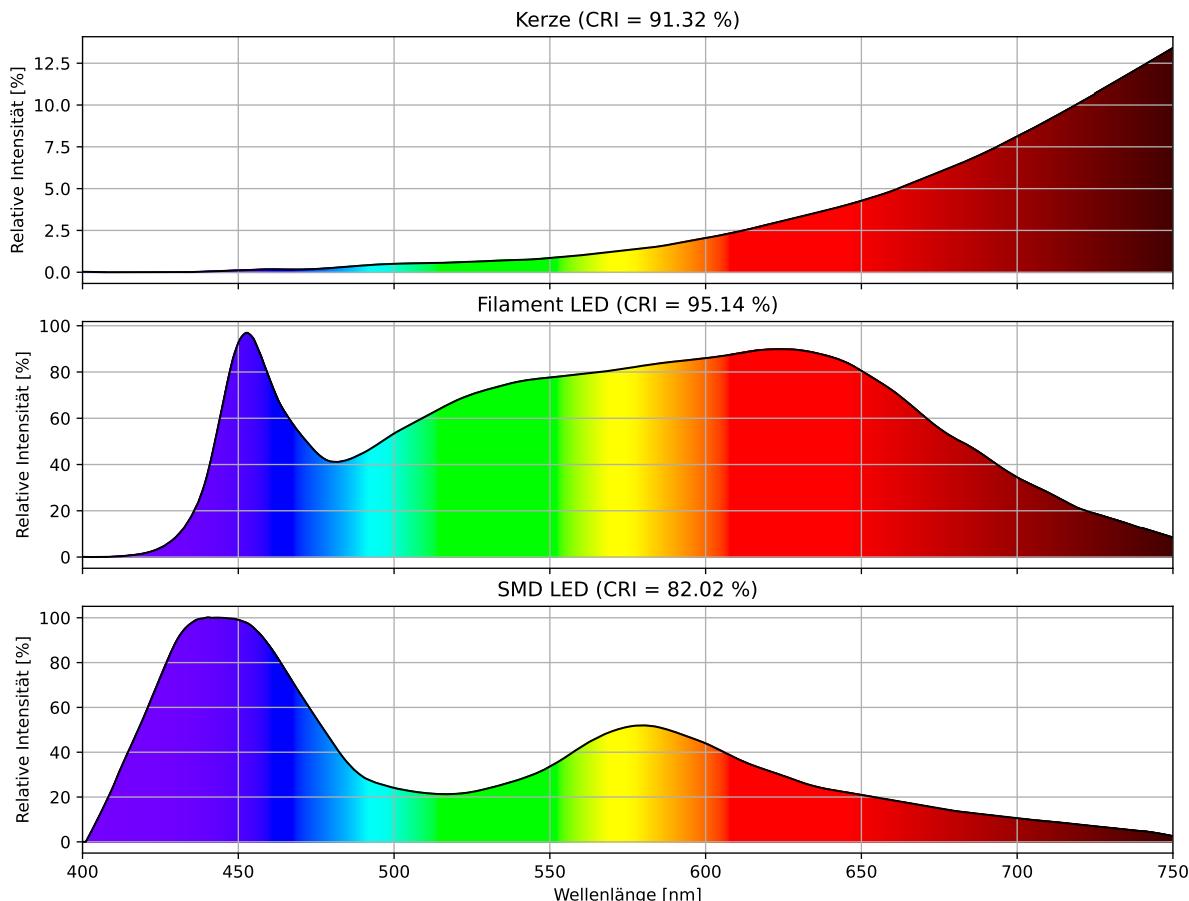


Abbildung 10: Spektrenvergleich und CRI der Wachskerze, Filament und SMD-LED. Den zugehörigen Code findet man auf [GitHub](#).

Man erkennt, dass das **SMD-LED**-Spektrum einen breiten und ausgeprägten Blauanteil hat und einen schwachen Grün- und Rotanteil. Im Vergleich zur Kerze haben das Filament und die **SMD-LED** einen schwachen Infrarotanteil. Das Kerzenspektrum hingegen besteht fast nur aus Infrarotanteil und hat nahezu keinen Blauanteil. Wie aus dem Spektrum zu erkennen ist, kommt das **LED**-Filament dem Kerzenlicht erheblich näher als die **SMD-LED**.

Wie bekommt man bei den **SMD-LEDs** ein Spektrum mit einem hohen CRI hin? Indem blaue **LEDs** verbaut werden, die den Phosphor anregen. Dadurch wird das blaue Licht teilweise in grünes und rotes Licht umgewandelt. Spezieller Phosphor für **LEDs** erzeugt ein hochwertiges Spektrum und ist teuer. Dies erfordert jedoch einen eigenen Entwicklungsschritt.

2.1.6 Fazit

Erst mit LEDs der Baureihe 0603 wäre das Kriterium des Spektrums und des Wirkungsgrades erfüllt, im Gegensatz zu kleineren Bauformen. Aber dagegen spricht der größere Platzbedarf und somit noch weniger Transparenz des Kerzenkopfes. Somit kommt die Realisierung mit SMD-LEDs nicht infrage. Der Nachteil der LED-Filamente ist die nötige Handarbeit.

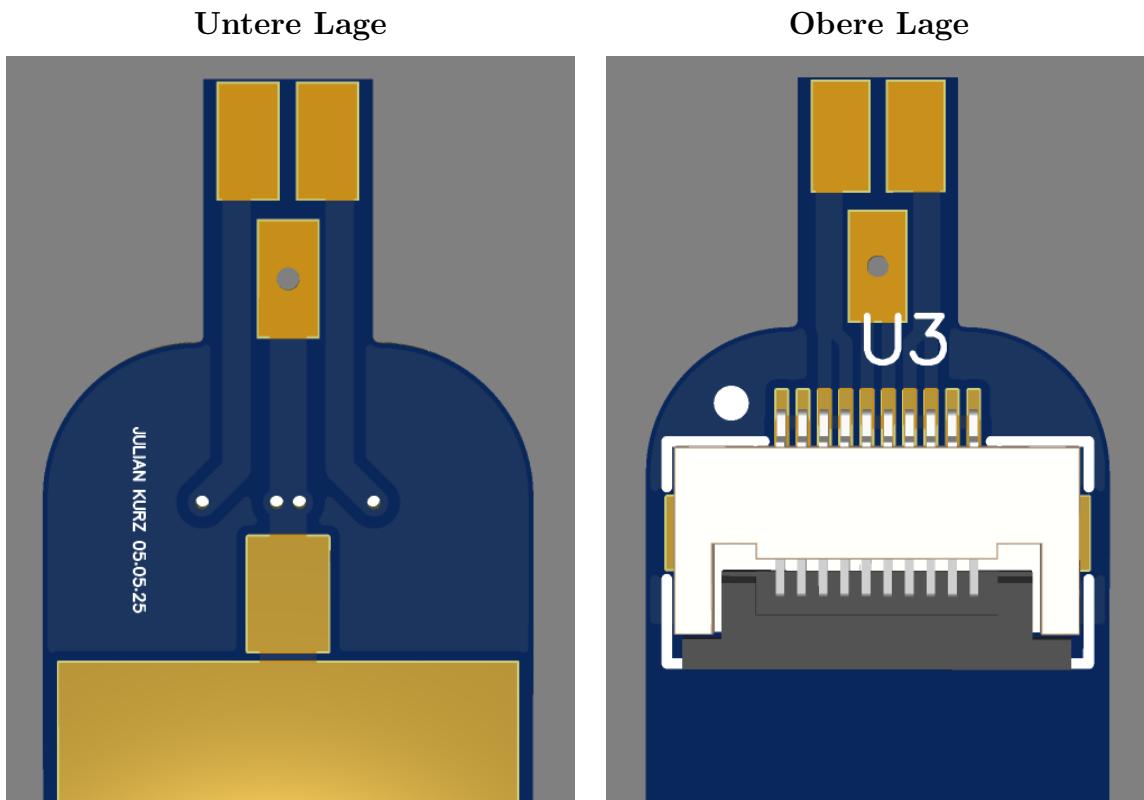
Dennoch fällt die Wahl eindeutig auf die LED-Filamente.

2.2 Platine des Kerzenkopfs

Was ist das Hardwarekonzept hinter der Kerze? Die Ansteuerung der LED-Filamente wird mit einem Pulse Width Modulation (PWM)-Signal je LED-Filament realisiert. Fortan repräsentiert jeder Quadrant ein LED-Filament und die dazugehörige Zuleitung wird als Kanal bezeichnet. Es wird also mit vier Quadranten im System und somit vier Kanälen gearbeitet.

Für die Fixierung der LEDs werden alle Kathoden mit einem PAD je Kanal auf einer Trägerplatine verlötet. Die Platine ist so gebaut, dass zwei Filamente auf einer Platinenseite nebeneinander passen und auf der Rückseite das gleiche mit dem minimal möglichen Abstand zueinander.

Tabelle 9: Platine des Kerzenkopfs – Untere und obere Lage



Die Verbindung zwischen Kerzenkopfplatine und Steuerplatine wird durch ein Flachbandkabel realisiert (siehe Kapitel 3).

In der folgenden Zeichnung ist die Ansteuerung des Kanals 1 zu sehen.

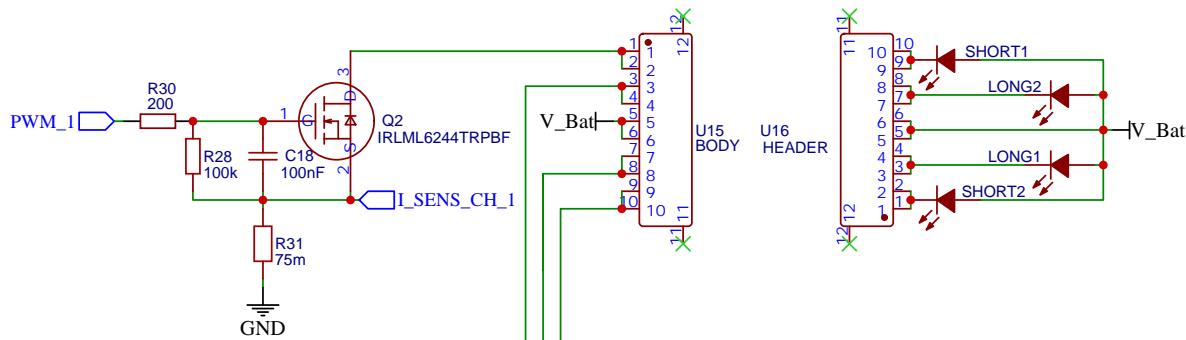


Abbildung 11: Ansteuerung des LED-Filaments über Kanal 1: Verbindung zwischen FPC-Steckverbinder HEADER und der Steuerplatine.

Im Schaltplan sind zwei FPC-Steckverbinder eingezeichnet. Die LEDs werden am Connector HEADER verbunden und über ein Flexkabel an die Steuerplatine angeschlossen. Die zu den LEDs sind hier nur symbolisch dargestellt, weil später im tatsächlichen Schaltplan der HEADER in den Pads endet, und die Filamente nach der Produktion der Kopfplatten manuell an die Pads gelötet werden.

LED-Kathode → FPC-Connector → Drain → SHUNT _ Resistor → GND

Sie werden je Kanal an den Drain eines [Metal-Oxide-Semiconductor Field-Effect Transistor \(MOSFET\)](#)s angeschlossen, welcher jeweils das LED-Filament mit GND kurzschließt. Gesteuert wird jede LED mit einem PWM-Signal vom TIM2 des STM32F103C8T6. Dazu später mehr. Die Anoden aller LEDs werden direkt mit der Lithium-Batterie verbunden. Dafür sehen wir auf den Bildern ein Pad in der Mitte der Platine mit einem Loch. Das Loch ist dafür da, die Anoden mit der Batterie zu verbinden.

LED-Anode → Batterie-Anode

Jeder Kanal ist auf einen Nennstrom von 500 mA ausgelegt. Nehmen wir das mal vier Kanäle, so darf der Strom 2000 mA nicht übersteigen. Ein MOSFET kann problemlos 2,6 A schalten, somit könnte man insgesamt wesentlich mehr Strom schalten. Die Schaltung hat eine Temperatursicherung bei 2000 mA. Es wird ein 10-adriges Flexkabel benutzt.

Die Stromversorgung der Steuerplatine benötigt nicht mehr als 500 mA. Diese wird über einen 3,3 V [Low Dropout Regulator \(LDO\)](#) geregelt. Weil der Strombedarf der Steuerplatine gering ist, wird der Pluspol der Batterie näher an den LEDs angeschlossen. Jedem Kanal werden zwei parallele Adern zugeordnet, dazu später im eigenen Kapitel mehr.

Der Minuspol der Batterie ist über die Lötfläche der Batterie direkt mit dem PAD der Steuerplatine verbunden. Der Pluspol ist mit der Kopfplatine verbunden.

Dadurch wird ein versehentlicher Kurzschluss in der Kopfplatine vermieden. Normalerweise sind die N-MOSFETs im Leerlauf gesperrt und somit kann durch die Kanäle kein Strom fließen. Erst der Microcontroller aktiviert den GND je Kanal.



Abbildung 12: Finaler Kerzenkopf vor dem Verguss mit Epoxidharz und Flachbandkabel

Die Filamente werden schließlich in Epoxidharz vergossen. Dadurch bekommt der Kerzenkopf die richtige Form, Stabilität und Wasserdichtheit. Außerdem wird die Streuung des Lichts erreicht



Abbildung 13: Finaler Kerzenkopf nach dem Verguss mit Epoxidharz und Flachbandkabel

3 Das Flachbandkabel

Die Schaltung ist für den Betrieb von LED-Filamenten mit einem Nennstrom von 120 mA und einer Nennspannung von 3 V konzipiert. Dennoch bestand die Anforderung, die Schaltung so auszulegen, dass auch höhere Ströme möglich sind, um mögliche Erweiterungen oder andere Anwendungen zu berücksichtigen.

Daher war es erforderlich zu überprüfen, ob das eingesetzte [Flexible Flat Cable \(Flexkabel\) \(FFC\)](#) für diese Strombelastung geeignet ist. Konkret wurde das Kabel HC-FFC05-10P-100-A-01 verwendet, ein 10-poliges FFC-Kabel mit einem Leiterabstand (Pitch) von 0,5 mm und einer Gesamtlänge von 100 mm [5]. Zu diesem Produkt liegt nur ein Datenblatt mit den mechanischen Abmessungen vor. Im Folgenden ein Foto vom Kabel.

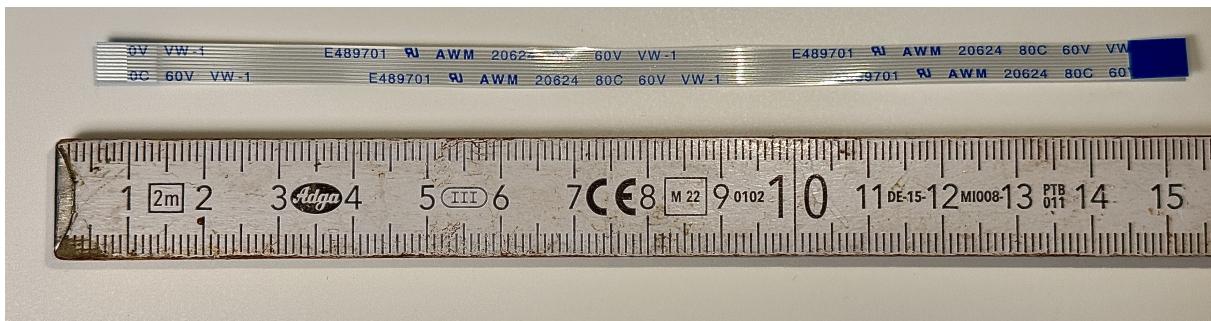


Abbildung 14: FFC-Kabel (HC-FFC05-10P-100-A-01)

Es soll die maximale Stromstärke abgewogen werden, die durch die Anordnung fließen könnte. Zuerst wird das Nutzungskonzept mit folgender Tabelle erklärt. CHx steht dabei für den Kanal und Bat steht dabei für die Batterie Anode.

Tabelle 10: Nutzung des FFC-Kabels

Ader	1	2	3	4	5	6	7	8	9	10
I_max	0,5 A									
Belegung	CH1	CH1	CH2	CH2	BAT	BAT	CH3	CH3	CH4	CH4

Elektrische Spezifikationen wie Stromtragfähigkeit oder Leitermaterial sind nicht angegeben. Deswegen wird auf das Datenblatt eines Produkts des japanischen Herstellers JST mit gleichen Parametern Bezug genommen.

Dabei wurde auf das Datenblatt der FH-Serie von JST [6] zurückgegriffen, das sich auf 0,5 mm-Pitch FPC-Steckverbinder bezieht. Dort wird ein maximaler Strom von 0,5 A pro Kontakt sowie eine maximale Spannung von 50 V angegeben. Somit sind sie mit dem eingesetzten FFC-Kabel vergleichbar.

Wie schon erwähnt, wurde für jeden Kanal der Strom auf zwei Adern verteilt. Dadurch ist der pro Kanal zur Verfügung stehende maximale Strom 1,0 A.

In einem der Versuche kam es zu einem Kurzschluss im Bereich des Flachbandkabels. Dabei verbrannte die Leiterbahn sofort und verfärbte sich schwarz. Zur Vermeidung einer massiven Überhitzung wurde auf der Steuerplatine eine 2 A PTC-Sicherung eingebaut.

Um zu ermitteln, wie sich der Aufbau Kopfplatine und Flachbandkabel bei verschiedenen Strömen verhält, wurde das Netzteil NANKADF WPS3010D als Stromquelle benutzt. Die Maximalspannung wurde auf 5,0 V gesetzt und beginnend bei 0 A schrittweise in 0,5 A Schritten erhöht. Das folgende Bild zeigt den Versuchsaufbau. Dabei wurden alle Adern parallel geschaltet.

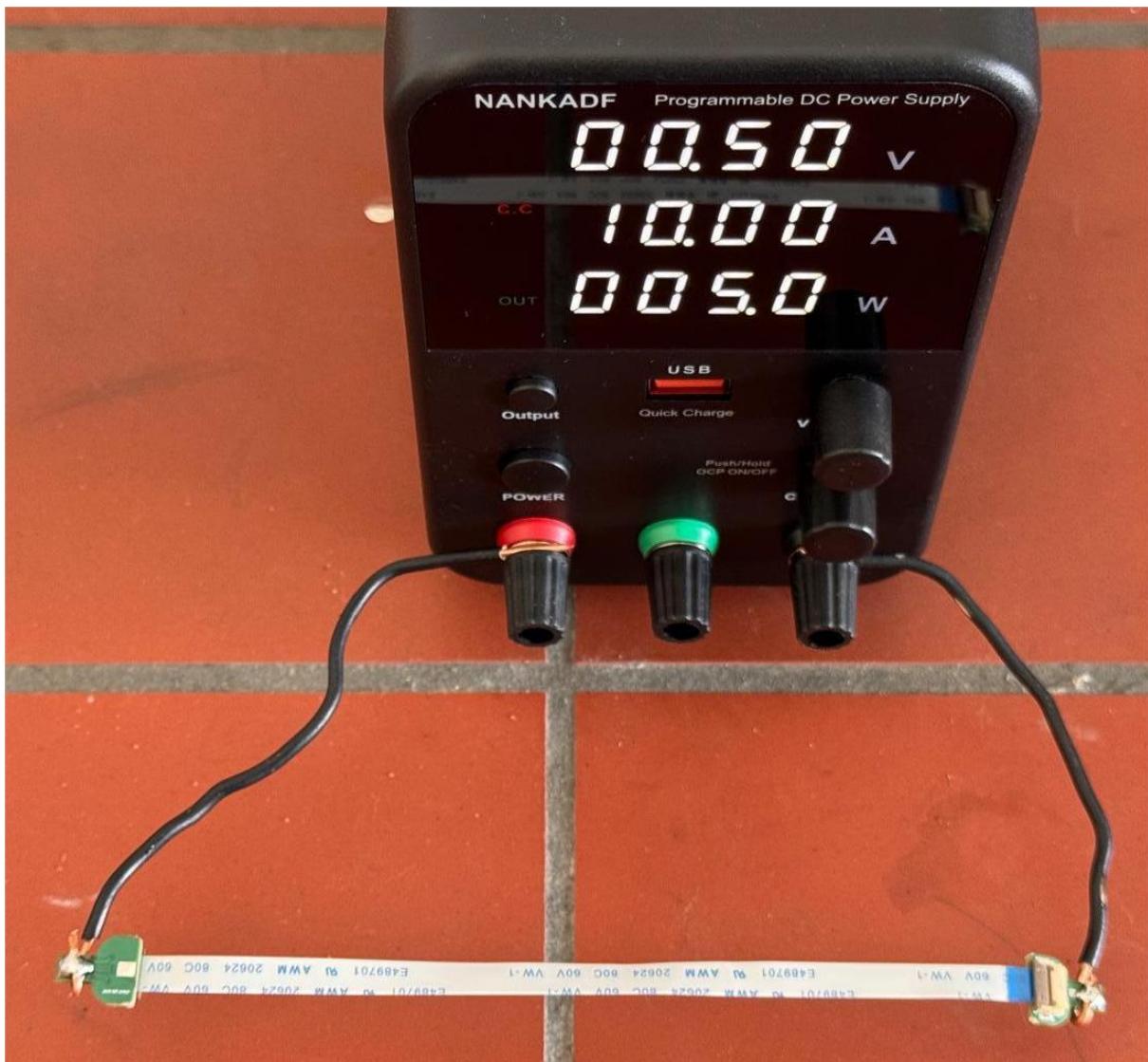


Abbildung 15: Stromfluss von 10 A durch durch alle Kanäle parallel geschaltet

Bei 5 A konnte ein leichtes Erwärmen festgestellt werden. Erst bei 10 A hat sich das Kabel deutlich erwärmt. Da das Netzteil nur 10 A leisten kann, wird hier eine Zwischenbilanz gezogen. Daraus folgt, dass das Flachbandkabel in Kombination mit dem Kerzenkopf pro Kanal ohne Probleme den ausgelegten Strom von 1000 mA verträgt. Dabei sind die LED-Filamente auf 120 mA ausgelegt.

Bei einem weiteren Versuch wurde gemessen, wie viel Strom der einzelne Kanal verträgt. In Abbildung ?? fließt der Gesamtstrom durch einen Kanal.

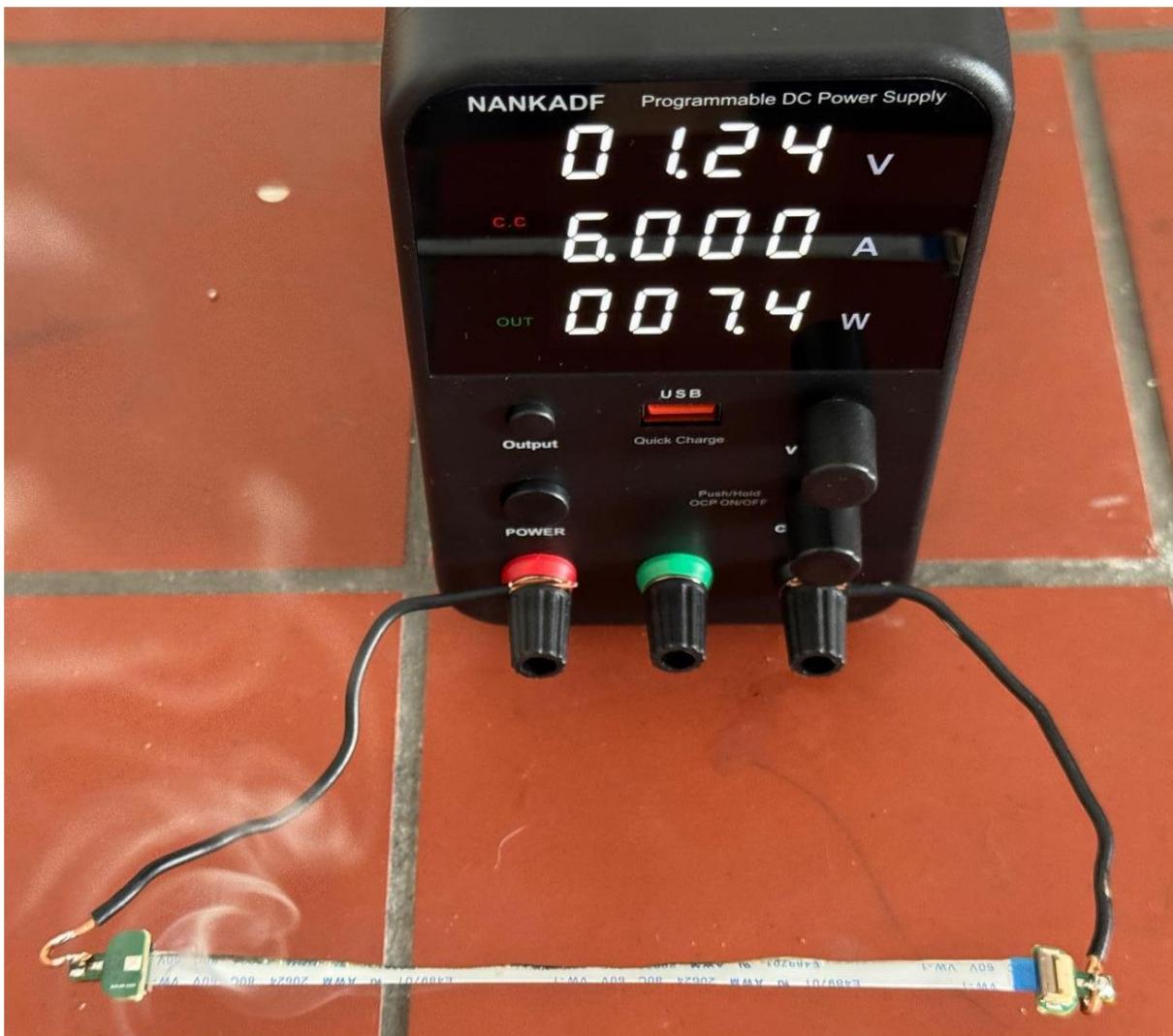


Abbildung 16: 6A fließt durch 2 Adern (einen Kanal). Das Kabel raucht.

Bei 6 A fängt der Draht an zu glühen, was zur Zerstörung der Kopfplatine und des Breitbandkabels führt. Dieser Strom ist um das dreifache größer als der Nennstrom der Sicherung. Fazit: der Betrieb der LEDs ist sicher.

4 Energiemanagement

4.1 Auswahl der Batterie

Für das Kerzenprojekt wurden drei zylindrische Li-Ionen-Zellen in einer engeren Auswahl verglichen, die abhängig vom verfügbaren Bauraum für unterschiedliche Kerzentypen geeignet sind. Alle Angaben beziehen sich auf die Nennspannung 3,6 V – 3,7 V und die typische Kapazität der jeweiligen Zelle. Es gibt noch viele weitere kompatible Exemplare, aber es werden folgende exemplarisch ausgerechnet. Die endgültige Entscheidung wird der Kunde nach persönlichen Anforderungen treffen.

Tabelle 11: Vergleich geeigneter Li-Ionen-Zellen

Zelle	IEC-Format	$\varnothing \times L$ [mm]	Kapazität	Dauerstrom
Panasonic NCR18650B	18650	18 × 65	3,4 Ah*	$\leq 2 C \approx 6,8 A^{**}$
DNK 14650	14650	14 × 65	1,1 Ah*	$\leq 2 C \approx 2,2 A^{**}$
DNK 14250	14250	14 × 25	0,52 Ah*	$\leq 1,0 A^{**}$

* Herstellerangabe (typisch)

** Max. empfohlener Dauerentladestrom

**Abbildung 17:** Mit der Schaltung kompatible Batterien der Typen 14650 und 14430

4.1.1 Lastfälle im Sollbetrieb

Die reale Stromaufnahme ist von der LED-Helligkeit abhängig. Für die Berechnungen werden drei praxisnahe Szenarien unterschieden. Der Strom der Steuerelektronik (≈ 10 mA) ist bereits eingerechnet. Die theoretische Laufzeit wurde nach folgender Formel berechnet:

$$\text{Theoretische Laufzeit} = \frac{\text{Kapazität}}{\text{Stromaufnahme}} \quad (5)$$

Tabelle 12: Lastfälle im Sollbetrieb

Batterie	Kerzentyp	Strom	Kapazität	Laufzeit
14250	Teelicht	250 mA	0,52 Ah	$\approx 2,1$ h
14650	Spitzkerze	250 mA	1,1 Ah	$\approx 4,4$ h
18650	Tafelkerze	250 mA	3,4 Ah	$\approx 13,6$ h

4.1.2 Lastfälle im Maximalbetrieb

Der maximale Strom, der je LED fließen darf, ohne dass diese Schaden nimmt, ist 120 mA. Wir rechnen also für alle Kanäle inklusive Steuerungselektronik mit 500 mA. Echte Kerzen haben von Natur aus auch inkonsistente Helligkeiten und Brennzeiten.

Tabelle 13: Lastfälle im Maximalbetrieb

Batterie	Kerzentyp	Strom	Kapazität	Laufzeit
14250	Teelicht	500 mA	0,52 Ah	≈ 1,0 h
14650	Spitzkerze	500 mA	1,1 Ah	≈ 2,2 h
18650	Tafelkerze	500 mA	3,4 Ah	≈ 6,8 h

4.1.3 Lastfälle im Grenzbetrieb

Nun zum Fall der maximalen Belastung der Schaltung, knapp unterhalb der Auslöseschwelle der Sicherung. Dieser Zustand dient der theoretischen Betrachtung. Die LEDs würde man bei solch einem Versuch durch Heizwiderstände ersetzen, da sie bei diesem Strom sofort verbrennen.

Tabelle 14: Lastfälle im Grenzbetrieb

Batterie	Kerzentyp	Strom	Kapazität	Laufzeit
14250	Teelicht	1000 mA	0,52 Ah	≈ 0,52 h
14650	Spitzkerze	2000 mA	1,1 Ah	≈ 0,55 h
18650	Tafelkerze	2000 mA	3,4 Ah	≈ 1,7 h

4.1.4 Vergleich mit echten Kerzen

In diesem Kapitel wird die elektronische Kerze mit einer echten Kerze hinsichtlich ihrer Leuchtdauer verglichen. Es wird davon ausgegangen, dass beide Kerzen unter normalen Bedingungen leuchten.

Tabelle 15: Vergleich mit echten Kerzen

Typ	Maße	Brennzeit	Zelle	Laufzeit
Teelicht	38 mm × 16 mm	3,5–4,5 h	14250	≈ 2 h
Spitzkerze	22 mm × 240 mm	8–9 h	14650	≈ 4,2 h
Tafelkerze	25 mm × 300 mm	8–10 h	18650	≈ 13 h

Eine pauschale Aussage zur Brenndauer echter Kerzen wird an dieser Stelle nicht getroffen, da diese stark von Faktoren wie Bauform, Wachsqualität, Umgebungsbedingungen (z.B. Temperatur, Luftfeuchtigkeit und Wind) und der Verarbeitung abhängt. Dennoch lässt sich festhalten, dass die elektronische Kerze, insbesondere bei kleinen Bauformen, eine deutlich kürzere Laufzeit aufweist, typischerweise etwa 50 % im Vergleich zur Wachskerze gleicher Größe. Mit zunehmendem Bauraum für die Batterie verschiebt sich dieses Verhältnis jedoch deutlich. Bei größeren Kerzenformen kann die elektronische Variante die Brenndauer einer echten Kerze sogar übertreffen. Die zentrale technische Herausforderung besteht darin, die Batterie möglichst kompakt und effizient in das begrenzte Volumen der Kerze zu integrieren.

4.2 Spannungsversorgung und Laderegler

Die E-Kerze ist nicht ohne Batterie betreibbar. Deshalb erklärt die folgende Grafik das Powermanagement der Kerze.

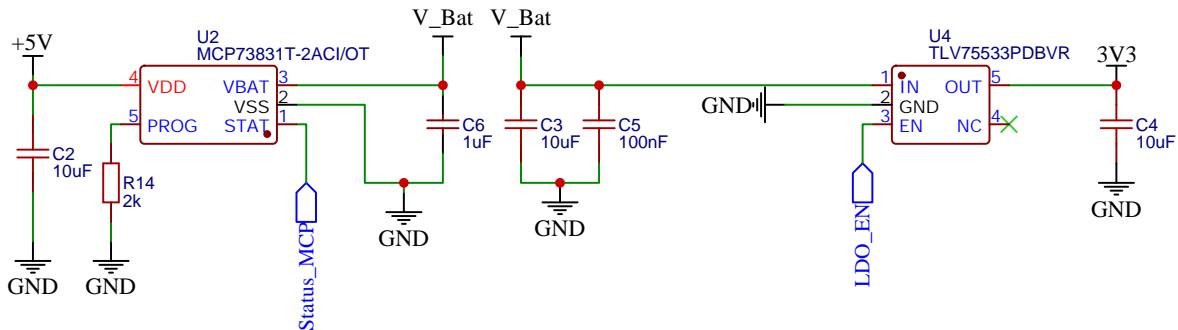


Abbildung 18: Powermanagement der Steuerplatine. Links der Batterieladeregler, rechts der Linearregler

Das Gerät lässt sich ohne angeschlossene Batterie programmieren, bricht jedoch bereits bei geringsten Lasten zusammen. Das liegt daran, dass die 5V-USB-Versorgung immer über den Laderegler geführt wird, während der Akku erst hinter dem Lade-IC in die Schaltung eingebunden ist.

Ein sogenanntes Power-ORing (z. B. über Dioden) wäre grundsätzlich möglich, würde jedoch selbst bei Schottky-Dioden zu einem Spannungsabfall von etwa 0,3 V führen. Zudem würde ein permanenter Leckstrom über die Diode entstehen. Da ein Betrieb ohne Akku nur im Entwicklungsfall oder bei einem Batteriedefekt vorgesehen ist, wird auf ein solches ORing bewusst verzichtet.

5 Ansteuerung der LEDs

5.1 Einführung Ansteuerung

Das im vorherigen Kapitel besprochene Flexkabel führt in die zentrale Steuerplatine der Kerze. Diese ist für die Steuerung der Energieversorgung und LEDs zuständig. Ferner läuft darauf der Kerzenalgorhythmus. Sie ist auch für die Messung der Ströme und Spannungen zuständig. Bei der Entwicklung standen vor allem ein kompaktes Layout, eine zuverlässige Stromversorgung und präzise Signalfilterung im Fokus. Außerdem sollte der Mikrocontroller große Datenmengen schnell verarbeiten können. Da diese Platine für die Rethink Lights GbR entwickelt wurde, wurden noch viele weitere Funktionen integriert. Zum besseren Verständnis wird die Platine vorgestellt, aber es wird nur auf die wesentlichen Funktionen eingegangen.

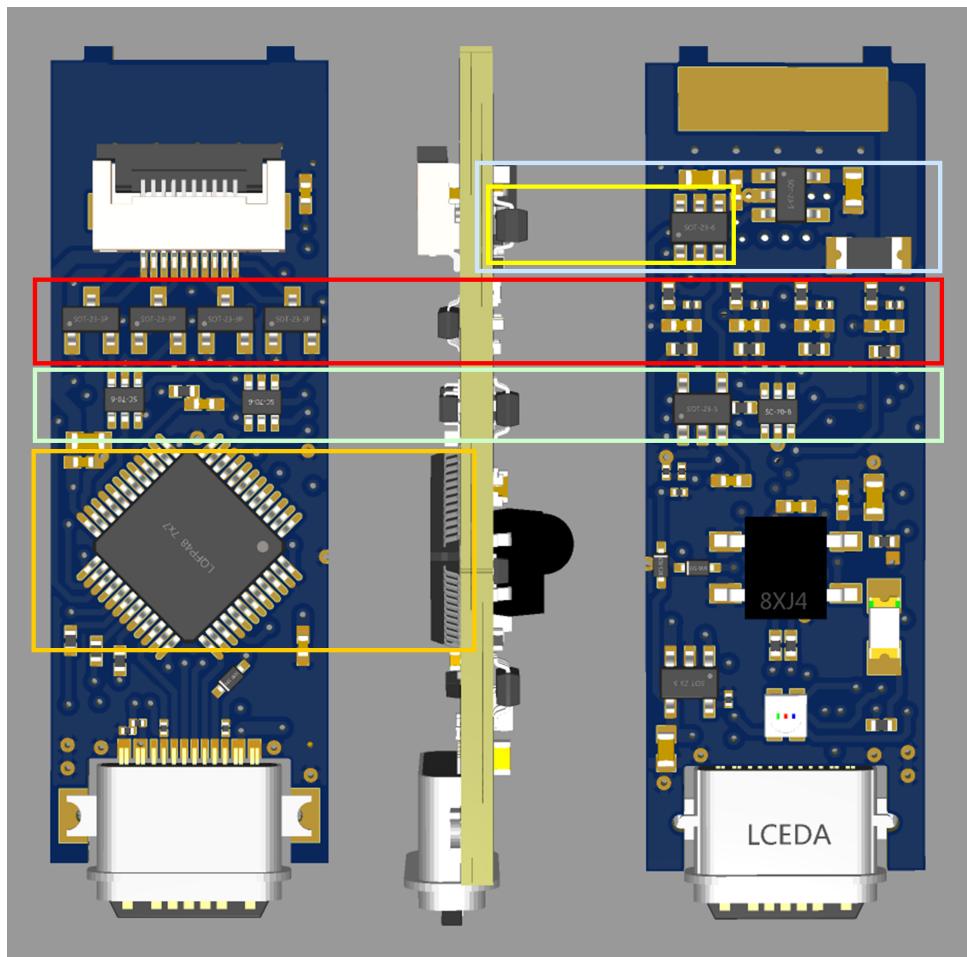


Abbildung 19: Triptychon der Steuerplatine. Die farblich markierten Bereiche werden in der folgende Tabelle 15 erläutert.

Die farblich eingerahmten Bereiche in Abbildung 19 markieren die signifikanten Einheiten, die für die Funktion relevant sind. Diese werden in Tabelle 16 erläutert.

Tabelle 16: Signifikante Funktionseinheiten der Steuerplatine

	Linearregler mit 2 A PTC-Sicherung
	Vier N-MOSFETs mit jeweils zwei Tiefpassfiltern
	Drei analoge Schalter mit Tiefpass und ein OPV als Impedanzwandler
	STM32F103C8T6 Mikrocontroller betrieben mit 64 MHz
	Batterielade-IC verbunden mit dem 5 V-Pin des USB-C-Steckers
	Instrumentenverstärker für die Strommessung mit Tiefpassfilter

Bevor die wesentlichen Aspekte der Platine beleuchtet werden, hier ein kurzer Überblick ihrer Funktionalitäten: Die Platine ist knapp $4\text{ cm} \times 1,5\text{ cm}$ groß und besteht aus vier Lagen. Herzstück ist ein STM32F103C8T6 mit bis zu 74 MHz Taktfrequenz, welche im Projekt jedoch nur mit 64 MHz betrieben wird.

Die Schaltung verfügt über:

- einen Batterieladeregler
- einen Linearregler
- eine Status-LED
- einen Strommessverstärker für die MOSFET-Kanäle
- einen NEC-Infrarotprotokoll-Empfänger
- einen Infrarotsensor
- eine RGB-Status-LED
- einen USB-C-Anschluss, über den das Gerät per RS232-Protokoll kommunizieren kann und über den sich die Platine programmieren, resetten und in den Boot-Modus versetzen lässt

Zusätzlich misst die Platine die Spannung der Batterie, die 5 V-Versorgung über [Universal Serial Bus \(USB\)](#) sowie die Temperatur. Jede einzelne Peripherie lässt sich vom Mikrocontroller abschalten, um im STOP-Modus einen minimalen Stromverbrauch zu erreichen. Es gibt noch eine weitere Methode, die LEDs anzusteuern, indem man den Ausgang der PWM über eine Filtereinheit direkt an jeder LED verbaut mit einem Kondensator, einer Diode und einer Drossel. Diese Methode ist jedoch bei kompakten Bauweisen ungeeignet. Die Drosseln sind große Bauteile; außerdem würden die Bauteile mehr Geld kosten. Der Vorteil der Drossel-Lösung wäre ein glatterer Ausgangstrom. Außerdem würde der Wirkungsgrad steigen und man könnte den Tiefpassfilter am Gate minimieren oder ganz weglassen.

Die Lösung mit der Drossel wurde jedoch vom Kunden abgelehnt, weil beim Film viele empfindliche Geräte zum Einsatz kommen. Besonders wenn mehrere digitale Kerzen im Betrieb sind, können sich die elektromagnetischen Auswirkungen potenzieren. Deshalb kommt diese Lösung für die Rethink Lights GbR nicht in Frage.

Ein weiteres Problem ist, dass die benötigte Größe für die Drossel umgekehrt proportional zur PWM-Frequenz ist. Das heißt, mit höherer PWM-Frequenz wird die benötigte Größe der Drossel kleiner. Aber dadurch sinkt auch die Auflösung des Tastgrads und somit die Auflösung der Helligkeit. Es würde dem Kriterium der realitätsnahen Nachbildung der Flamme im Wege stehen.

Bei diversen Simulationsversuchen kam heraus, dass eine PWM von 60 kHz benötigt wird, um einen geeigneten Kompromiss zwischen Stromripple, Bauteilgröße und Tastgrad-Auflösung zu bekommen. Somit würde man einen Tastgrad von 1000 Stufen realisieren können. Diese Lösung würde jedoch eine Drossel von etwa 500 μH benötigen, welche groß und teuer ist.

In vielen LED-Schaltungen werden LED-Treiber verbaut. Um die maximale Flexibilität bei Gestaltung und Nutzung der Platine zu haben, wurde auf diese Bauteile verzichtet – auch weil man sich nicht von der Verfügbarkeit dieser Bauteile abhängig machen möchte. Es kommt regelmäßig vor, dass ein Bauteil zum Zeitpunkt der Entwicklung verfügbar ist, jedoch bei der Auftragserteilung an JLCPCB bereits nicht mehr lieferbar und die Schaltung kurzfristig angepasst werden muss.

Damit der Betrieb nach Wunsch später mit einer Drossel und einer höheren als die gewählte Frequenz möglich ist, wurde die Steuerplatine so designt, dass die Kondensatoren der Tiefpässe an den Gates mit einer Entlötpinzette schnell entfernt werden können, um die starke Filterung abzuschwächen.

5.2 Nutzung des MOSFETS

Die Stromversorgung erfolgt durch eine 3,7 V Lithiumbatterie. Vom LED-Modul führt das Flachbandkabel entlang der Batterie zur Steuerplatine, welche unter anderem für die Ansteuerung der einzelnen Kanäle verantwortlich ist.

Die MOSFETs übernehmen die zentrale Rolle bei der Leistungssteuerung. Für die Kerze wurde der IRLML6244TRPBF von Texas Instruments verwendet. Es handelt sich um einen kompakten N-Kanal-MOSFET im SOT-23-Gehäuse, hergestellt von Infineon Technologies. Er ist speziell für den Einsatz mit Logikpegeln (z. B. 3,3 V oder 5 V) ausgelegt und eignet sich ideal für die Ansteuerung kleiner Lasten per PWM.

Wichtige Kennwerte:

- Drain-Source-Spannung (V_{DS}): 30 V (max.)
- Dauerstrom (I_D): bis 2,6 A bei 4,5 V Gate-Spannung
- Durchlasswiderstand ($R_{DS(on)}$): 35 mΩ bei $V_{GS} = 4,5$ V
- Gate-Schwellspannung ($V_{GS(th)}$): typ. 1 V
- Maximum Power Dissipation (@ 25 °C): 1,3 W

Die MOSFETs genügen somit den Erfordernissen des Kerzenprojekts.

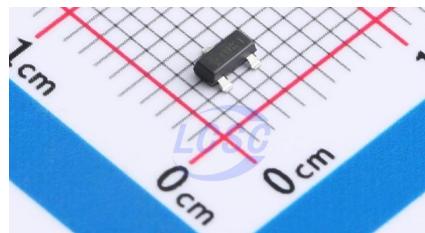


Abbildung 20: Abbildung des eingesetzten MOSFETs [7]

Besonders hervorzuheben ist die hohe Strombelastbarkeit der eingesetzten MOSFETs von bis zu 2,6 A. Dieser Wert liegt deutlich über der gewählten Sicherungsgrenze von 2 A. Ebenfalls erwähnenswert ist die hohe zulässige Drain-Source-Spannung des Bauteils. Da bei Rethink Lights GbR häufig 24 V-Dimmer und entsprechende LED-Systeme zum Einsatz kommen, ist die verwendete Platine durch diese MOSFET-Auswahl auch für höhere Betriebsspannungen gut geeignet.

5.2.1 Hardwarerealisierung der Ansteuerung

Der folgende Auszug zeigt die gesamte Steuer-Elektronik für die PWMs. Die **PWM_1** bis **PWM_4** gehen zum Mikrocontroller und werden vom **TIM2** angesteuert.

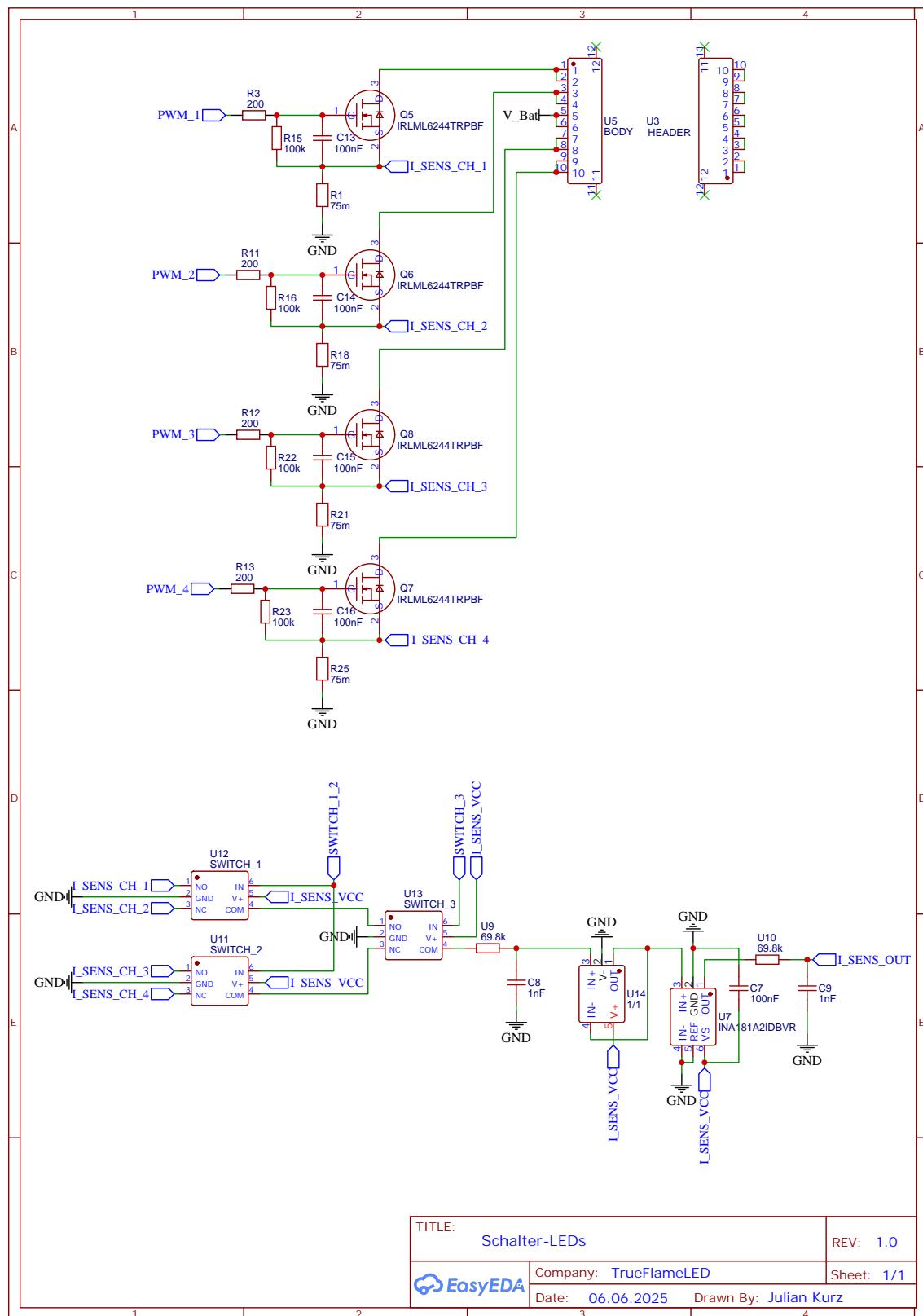


Abbildung 21: Schaltplan-Ausschnitt der PWM-Ansteuerung

Vor jedem Gate befindet sich ein Tiefpassfilter. Ursprünglich sind PWM-Spitzen sehr stark ausgeprägt. Das menschliche Auge ist bei sehr schwachem Licht sehr empfindlich. Zudem könnten steile Flanken EMV-Störungen verursachen. Der Tiefpassfilter sorgt für einen sanften Übergang von dunkel zu hell. Ohne Tiefpassfilter würde der MOSFET den Strom sofort voll durchschalten. Durch den Tiefpass wird der maximale Durchschaltmoment verzögert. Das ist ein klassisches Problem bei Dimmern. Qualitativ minderwertige Dimmer springen schon auf den ersten Stufen sichtbar an, hochwertige erst nach vielen feinen Abstufungen.

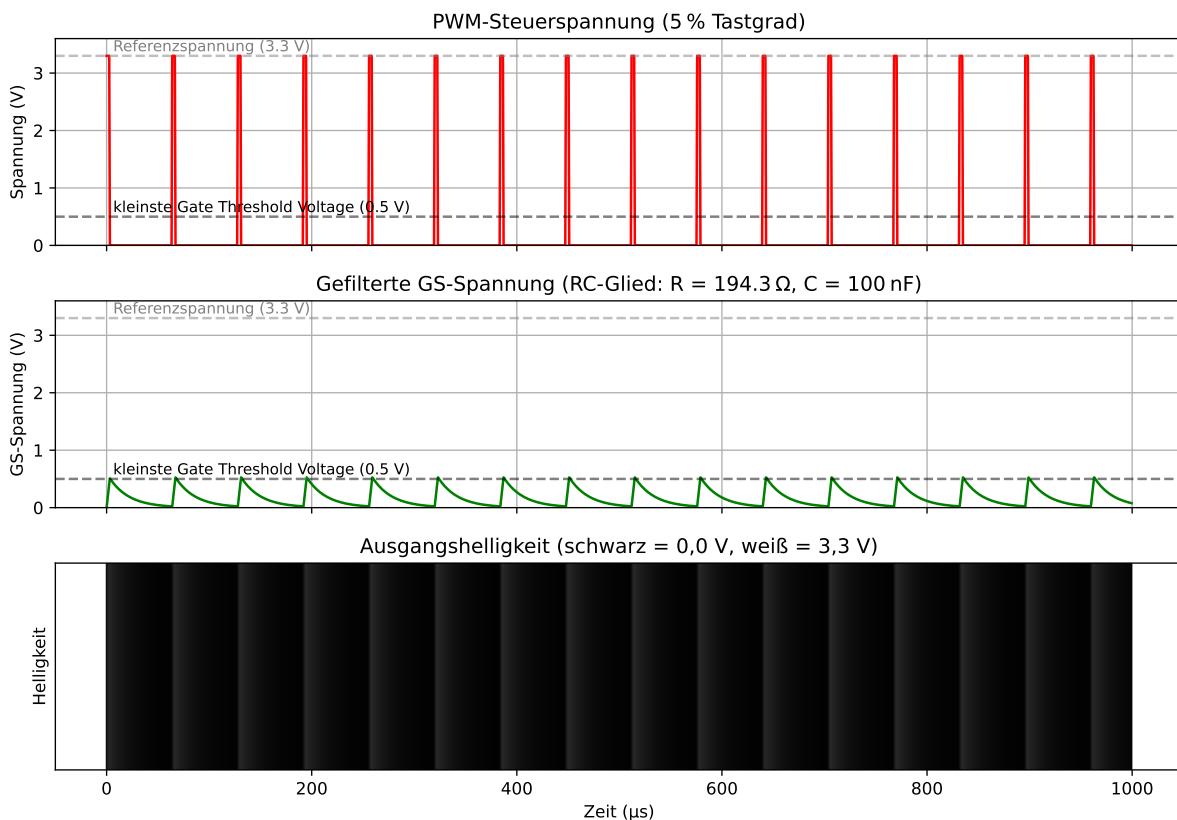


Abbildung 22: Plots zum visuellen Effekt des Tiefpassfilters am Gate. Tastrate liegt bei 5 %. Den zugehörigen Code findet man auf [GitHub](#).

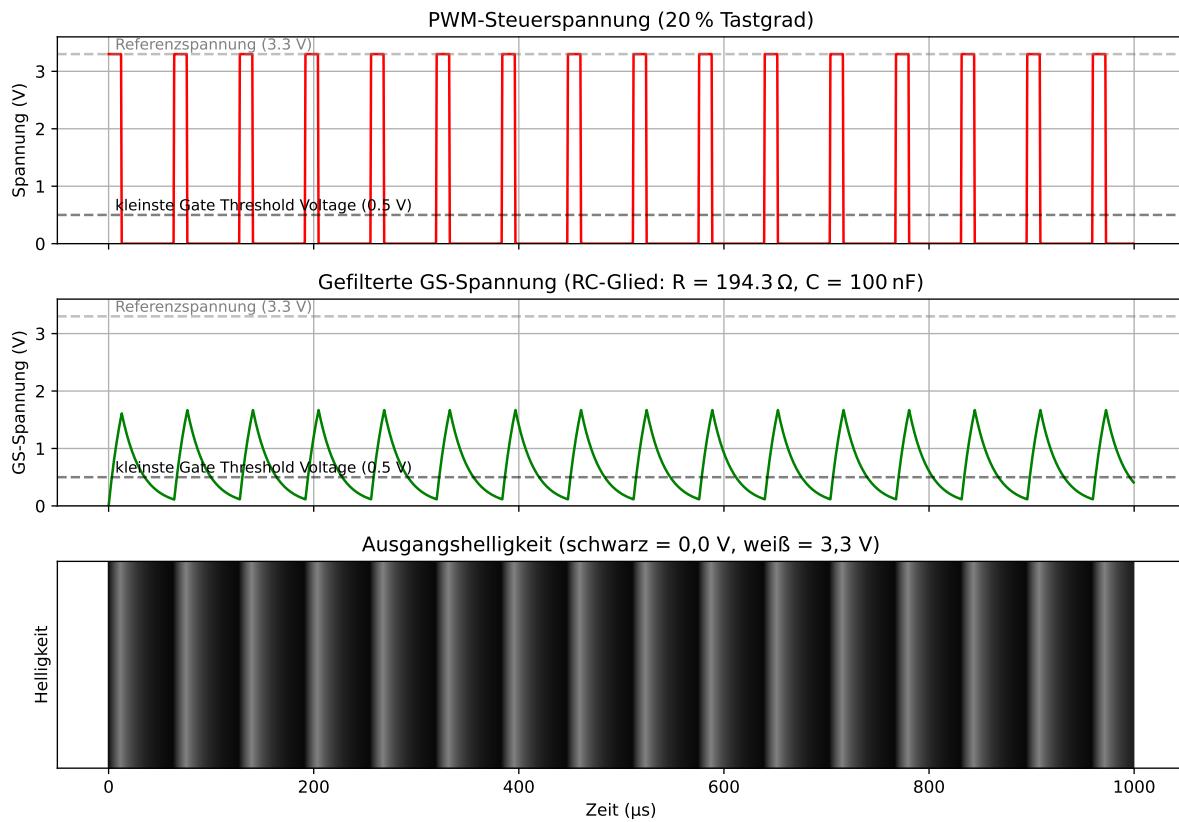


Abbildung 23: Zur besseren Veranschaulichung wird ein weiterer Plot als Helligkeitsverlauf visualisiert.

Bei einer normalen, rein PWM-gesteuerten LED erkennt man den Tastgrad, wenn man die Lichtquelle in Slow-Motion mit einer Kamera aufnimmt. Man erkennt bei einer offenen LED-Birne, wie auf dem Foto unten, auf dem Board einen Trafo, eine Spule und zwei große Kondensatoren. Diese dienen der Glättung der Spannung.



Abbildung 24: Der rote Pfeil zeigt auf die internen Bauteile einer LED-Birne (Kondensatoren, Trafo, Spule)

Es wurde bei eigenen Versuchen festgestellt, dass bei günstigen LED-Birnen das starke Flackern nach dem Einbau eines Elektrolytkondensators für den Betrachter verschwand. Das Flackern hing mit der Netzfrequenz von 50 Hz zusammen. Theoretisch wäre das Problem durch eine ausreichend hohe Frequenz lösbar. Am Anfang der Überlegungen, noch vor der Simulation, stand die Frage, wie schnell die PWM überhaupt sein kann und wie viele Helligkeitsstufen realisierbar wären.

Vorab einige Überlegungen. Der STM32 kann mit 72 MHz takten, aber nur mit externem Quarz. Alternativ kann man den internen RC-Taktgeber einschalten, welcher einen Takt von 64 MHz liefert.

Nun zum Timer. Auf dem ausgewählten Mikrocontroller gibt es vier Timer, welche hardwaremäßig verbaut sind und unterschiedliche Eigenschaften haben. Es wurde der **Timer 2** ausgewählt, weil die Pins an einer für das Layout geeigneten Position sind. Der Timer kann theoretisch bis `0xffff` hochzählen.

Je höher die PWM-Frequenz, desto höher die zeitliche Auflösung, vergleichbar mit FPS beim Film. Aber dadurch wird das **ARR** (Auto-Reload-Register) kürzer, die Tastgradauflösung und somit auch die Helligkeitsauflösung geringer. Hier musste man einen Kompromiss finden.

5.2.2 Wahl der PWM-Auflösung und -Frequenz

In einigen Fällen können PWM-Signale auf elektronischen Schaltungen Geräusche verursachen. Bei einem Selbstversuch wurde das hörbare Spektrum abgehört und dabei festgestellt, dass die Hörgrenze bei etwa knapp unter 16 kHz liegt. Dieser Wert liegt deutlich über der Hörgrenze eines Erwachsenen. Wenn die PWM-Frequenz also auf 16 kHz gesetzt wird, sind störende Geräusche nicht mehr zu hören.

Ein weiterer Aspekt ist die Helligkeitsauflösung. Da übliche Bildformate und LED-PWMs eine Auflösung von 8 bit haben, galt es dies zu übertreffen.

Der Zusammenhang zwischen den relevanten Größen ergibt sich aus:

$$f_{\text{PWM}} = \frac{f_{\text{Timer}}}{(PSC + 1)(ARR + 1)} \quad (6)$$

Umgestellt nach ARR:

$$ARR = \frac{f_{\text{Timer}}}{(PSC + 1) f_{\text{PWM}}} - 1 \quad (7)$$

Mit $f_{\text{PWM}} = 16 \text{ kHz}$, $PSC = 0$ und $f_{\text{Timer}} = 64 \text{ MHz}$ ergibt sich:

$$ARR = \frac{64 \text{ MHz}}{16 \text{ kHz}} - 1 = 3999 \quad (8)$$

Somit ist die Wahl von 4096 sowohl rechnerisch sinnvoll als auch aus technischer Sicht optimal. Da es sich um eine Zweierpotenz handelt (2^{12}), ist die Implementierung auf einem digitalen System besonders effizient.

Damit ergibt sich für die Ansteuerung der LEDs eine Helligkeitsauflösung von 12 bit. Diese Auflösung ermöglicht eine angemessen feine Abstufung der Leuchtintensität.

5.3 Auswahl der Betriebsart

5.3.1 Übersicht der Betriebsarten

Es gibt einige Methoden, die LEDs anzusteuern. Eine dieser Methode wird beim Abwärtswandler verwendet. Der PWM-Strom lässt sich mit einem Kondensator, Diode und Drossel direkt an jeder LED glätten. Die dafür nötigen Drosseln sind jedoch groß und teuer, sodass das Verfahren ohne stark erhöhte PWM-Frequenzen nicht für kompakte Designs taugt. Obwohl es einen gleichmäßigeren LED-Strom und höheren Wirkungsgrad bietet, kommt diese Lösung für den Kunden nicht in Frage, auch wegen möglicher kumulierter elektromagnetischer Störungen am Filmset.

Um sich nicht von der Verfügbarkeit einzelner Bauteile abhängig zu machen, muss sicher sein, dass alle Komponenten zum Zeitpunkt der Produktion verfügbar sind. Gelegentlich waren zu Beginn der Entwicklung die noch bei JLCPCB verfügbaren Bauteile zum Produktionszeitpunkt nicht mehr verfügbar, und die Schaltung musste angepasst werden. Damit der Betrieb nach Wunsch später mit einer Drossel und einer höheren als die gewählte Frequenz möglich wäre, wird die Steuerplatine so designt, dass die Kondensatoren der Tiefpässe an den Gates mit einer Entlötpinzette schnell entfernt werden können. Dadurch würden die MOSFETs im Schaltbetrieb arbeiten.

Im nachfolgenden werden der Schalt- und Linear- und Soft-Switch-Betrieb näher betrachtet. Zur Abschätzung der elektrischen Verluste wird ein vereinfachter Verbraucher angenommen, der sich als ohmscher Widerstand modellieren lässt. Dabei wird auf eine reale

LED-Struktur bewusst verzichtet, um den Vergleich der beiden Ansteuerungsarten (linear vs. PWM) unabhängig von nichtlinearen Effekten zu ermöglichen.

Annahmen:

- Maximale Betriebsspannung der Batterie: $U_{\text{Bat},\text{max}} = 4,2 \text{ V}$
- Nennspannung der Last (repräsentativ): $U_{\text{Nenn}} = 3,0 \text{ V}$
- Nennstrom: $I_{\text{Nenn}} = 120 \text{ mA}$
- Repräsentativer Verbraucher: $R_{\text{Last}} = \frac{U_{\text{Nenn}}}{I_{\text{Nenn}}} = 25 \Omega$
- Innenwiderstand des MOSFETs im Durchlassbetrieb: $R_{\text{DS(on)}} = 21 \text{ m}\Omega$

5.3.2 Option 1: Schaltbetrieb

Im Schaltbetrieb bleibt die mittlere Stromaufnahme der Last identisch (120 mA). Der Strom wird jedoch durch einen MOSFET mit sehr niedrigem Innenwiderstand geschaltet, wodurch nur geringe Verluste entstehen.

$$P_{\text{MOSFET}} = I^2 \cdot R_{\text{DS(on)}} = (0,12 \text{ A})^2 \cdot 21 \text{ m}\Omega \approx 0,302 \text{ mW} \quad (9)$$

$$P_{\text{gesamt}} = P_{\text{Last}} + P_{\text{MOSFET}} = \underbrace{U_{\text{Nenn}} \cdot I_{\text{Nenn}}}_{360 \text{ mW}} + 0,302 \text{ mW} \approx 360,3 \text{ mW} \quad (10)$$

$$\eta_{\text{PWM}} = \frac{P_{\text{Last}}}{P_{\text{gesamt}}} \approx 99,92 \% \quad (11)$$

5.3.3 Option 2: Linearbetrieb

Im linearen Betrieb fällt ein Teil der Batteriespannung an einem in Serie geschalteten Vorwiderstand ab. Die Last erhält dabei konstant ihre Nennspannung von 3,0 V bei 120 mA.

$$P_{\text{gesamt}} = 4,2 \text{ V} \cdot 0,12 \text{ A} = 504 \text{ mW} \quad (12)$$

$$P_{\text{Last}} = 3,0 \text{ V} \cdot 0,12 \text{ A} = 360 \text{ mW} \quad (13)$$

$$P_{\text{Verlust}} = 504 \text{ mW} - 360 \text{ mW} = 144 \text{ mW} \quad (14)$$

$$\eta_{\text{linear}} = \frac{360 \text{ mW}}{504 \text{ mW}} \approx 71,43 \% \quad (15)$$

Da die maximale Verlustleistung des MOSFETs bei 1,3 W liegt, wäre dieser Betrieb zwar möglich, sehr ineffizient und das Bauteil würde sich aufheizen.

5.3.4 Option 3: Soft-Switching-Betrieb

Die Kerze wird mit einer „hybriden PWM“ gesteuert. Was heißt das? Würde man einen sehr starken Tiefpassfilter nehmen, dann würde das PWM-Steuersignal zu einer Gleichspannung werden. Dies würde aber den MOSFET im ohmschen Bereich betreiben. Das heißt, der MOSFET verhält sich wie ein regelbarer Widerstand. Dadurch würde der Strom eine erhebliche Verlustleistung am MOSFET verursachen und ihn erhitzten. Wie im MOSFETs beschrieben, würde es ohne Filter stark ausgeprägte Flanken geben. Es gilt hier einen Kompromiss zu finden.

5.3.5 Vergleich der Betriebsarten

Tabelle 17: Vergleich der Betriebsarten bezüglich des Wirkungsgrads

Betriebsart	Nutzleistung	Verlustleistung	Gesamtleistung	Wirkungsgrad
Linearbetrieb	360 mW	144 mW	504 mW	71,43 %
Schaltbetrieb	360 mW	0,302 mW	360,3 mW	99,92 %

Das bedeutet, dass mit zunehmendem Effekt des Tiefpassfilters der Wirkungsgrad sinkt, jedoch die Signalflanken glättet und die EMV-Störunterdrückung verbessert werden. Die Entscheidung fällt zugunsten der Soft-Switch-Betriebsart, weil diese die Vorteile der beiden anderen Betriebsarten vereint.

5.3.6 Auslegung des Tiefpassfilters

Der Widerstand des Tiefpasses wurde so dimensioniert, dass in den Kondensator ein maximaler Strom des Mikrocontrollers nicht überschritten wird. Bei 3,3 V ist der maximale Strom 16,5 mA, bei 200Ω . Der maximale Nennstrom des Pins ist 25 mA.

Es wird die Vorgabe getroffen, dass erst bei einer Tastrate von 5 % die Spitzenspannung der Steuer-PWM den Durchlassbereich erreicht. Damit ist ein weiches Einblenden garantiert. Das Problem ist, dass die Gate Threshold Voltage eine große Toleranz von 0,5 V bis 1,1 V hat. Sicherheitshalber wird mit dem Minimum von 0,5 V gerechnet, um den Durchlassbereich eher später zu erreichen als zu früh.

Im Folgenden wird der Tiefpassfilter ausgelegt.

Tabelle 18: Parameter zur Auslegung des Tiefpassfilters

Größe	Wert
PWM-Frequenz f	$\frac{64 \text{ MHz}}{0xffff} \approx 15,63 \text{ kHz}$
PWM-Periode T	$64 \mu\text{s}$
Duty Cycle D	5 %
High-Zeit t_{high}	$3,2 \mu\text{s}$
Eingangsspannung V_{PWM}	3,3 V
Zielspannung U_{GS}	0,5 V
Kapazität C	100 nF

$$\text{Formel: } U_{\text{GS}} = V_{\text{PWM}} \left(1 - e^{-\frac{t_{\text{high}}}{R \cdot C}} \right) \quad (16)$$

$$\text{Umgestellt: } R = -\frac{t_{\text{high}}}{C \cdot \ln\left(1 - \frac{U_{\text{GS}}}{V_{\text{PWM}}}\right)} \approx 194,3 \Omega \quad (17)$$

Damit ist der Tiefpass so dimensioniert, dass erst bei 5 % Tastgrad der Durchlassbereich erreicht wird und ein sanftes Einschalten gewährleistet ist.

5.3.7 Fazit

Das nachfolgende Bild wurde mit einer Canon 750D Spiegelreflexkamera aufgenommen und zeigt den Kerzenkopf im Betrieb bei einem konstanten Tastgrad von 50 % auf allen Kanälen . Das Foto wurde aufgenommen bei einer Verschlusszeit von 1/4000 und ISO 12800. Zum Vergleich, die Verschlusszeit wird benötigt, um Rotorblätter eines Hubschraubers oder den Flügelschlag eines Kolibris aufnehmen zu können.

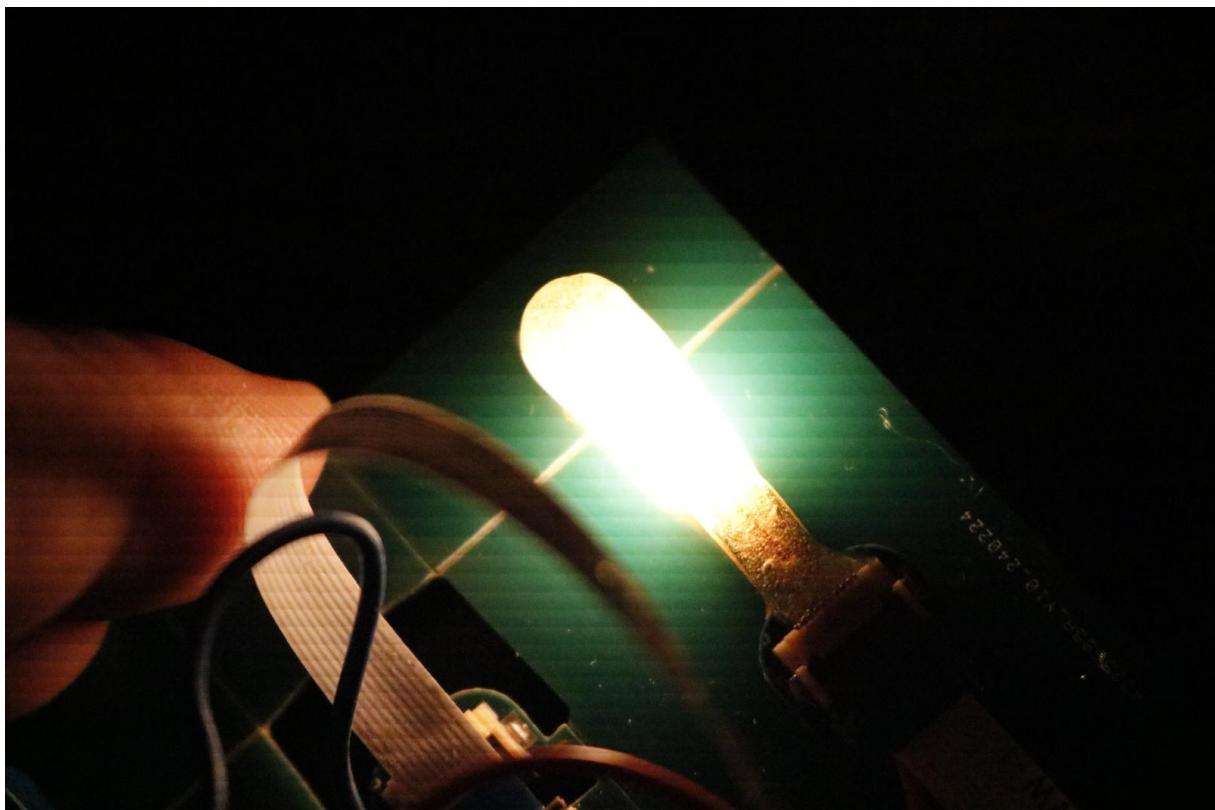


Abbildung 25: Foto des Kerzenkopfes aufgenommen mit einer Canon 750D bei einer Verschlusszeit von 1/4000 und ISO 12800. Tastrate für alle Kanäle bei 50 %.

Daraus folgt, dass die LED-Kerze die Anforderungen der Filmindustrie bezüglich Flackern und **PWM** erfüllt.

5.4 Strommessung

5.4.1 Einleitung

In diesem Kapitel wird erklärt, wie man eine Schaltung designt, die den mittleren Strom im ms-Takt messen kann. Bei gleicher mittlerer Spannung kann je nach unterschiedlicher PWM-Peak-Spannung ein anderer mittlerer Strom entstehen. Auch die Temperaturänderung hat einen Einfluss auf die Kennlinie. Um den tatsächlichen mittleren Stromwert zu messen, wird in die LED-Kerze ein Strommessverstärker verbaut. Dieses Bauteil dient der präzisen Überwachung des durch jeden einzelnen Kanal fließenden Stroms.

Folgende Anforderungen werden an einen Strommessmechanismus gestellt:

1. Die Schaltung soll alle Kanäle messen können.
2. Die Schaltung soll einen möglichst großen Strombereich erfassen.
3. Der **Analog-Digital-Converter (Analog-Digital-Wandler) (ADC)** soll in der Lage sein, das Ausgangssignal des Instrumentenverstärker (**INA**)s zu messen.
4. Der Ripple am **ADC**-Eingang soll möglichst klein sein.
5. Die Schaltung soll möglichst genau messen.

Es gibt viele Möglichkeiten, diese Schaltung zu realisieren. Folgende technische Bausteine kommen infrage: Ein Instrumentenverstärker, ein Shunt-Widerstand, ein **Multiplexer (MUX)**, ein Tiefpass und ein Analog-Digital-Wandler. Die in der untenstehenden Tabelle rot untermalten Regionen sind die unbrauchbaren Optionen. Die grün markierte Variante kommt in Frage.

Tabelle 19: Bewertung möglicher Signalpfade für die Strommessung

Variante	Signalpfad	Bemerkung
V1	4× Shunt → 4× ina → MUX → TP → ADC	Zu großer Platzbedarf
V2	4× Shunt → MUX → ADC	Ungenau
V3	4× Shunt → MUX → INA → ADC	Slew-Rate des INAs kann begrenzend wirken
V4	4× Shunt → MUX → INA → TP → ADC	Slew-Rate des INAs kann begrenzend wirken
V5	4× Shunt → MUX → TP → INA → ADC	Großer Platzbedarf, Messfehler durch Bias-Strom des INAs

Zum besseren Verständnis wird vorweg die realisierte Strommessschaltung gezeigt und anschließend der Lösungsweg erklärt.

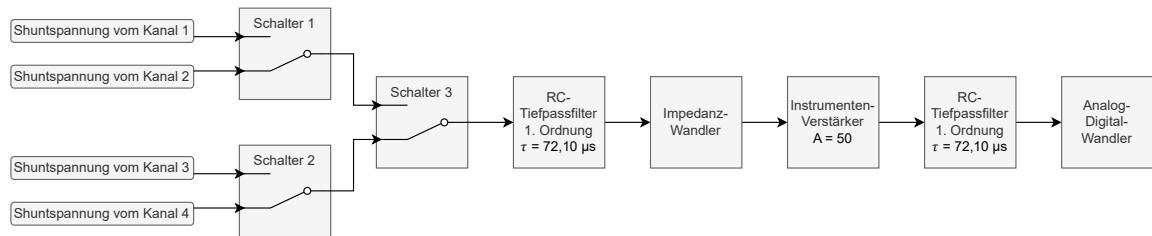


Abbildung 26: Übersicht der realisierten Strommessschaltung

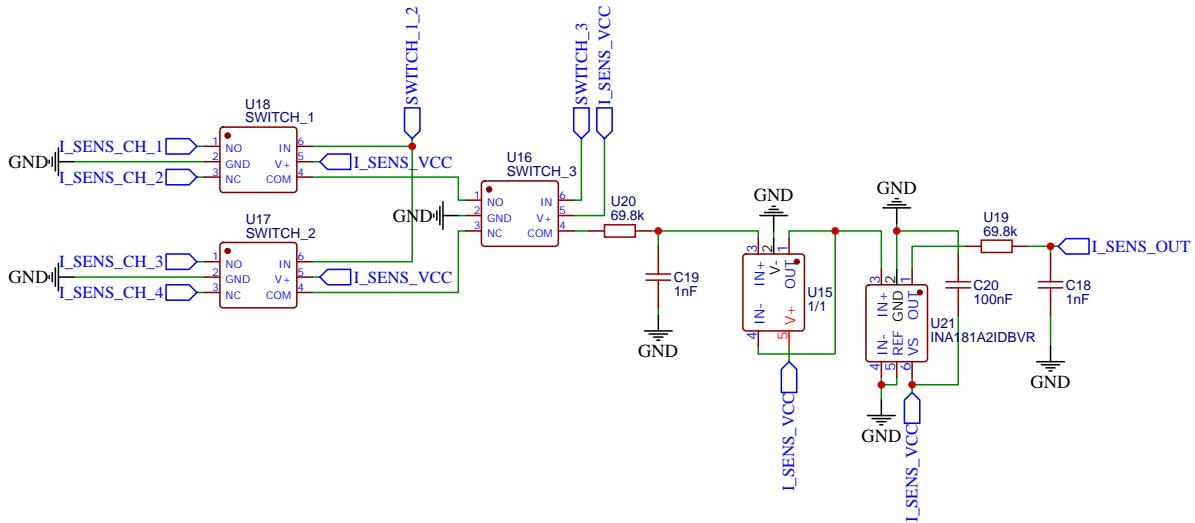


Abbildung 27: Schaltplan der Strommessung

Die folgende Tabelle gibt eine Übersicht über die Signale und Leitungen, die im Rahmen der Strommessarchitektur verwendet werden.

Tabelle 20: Erläuterung der Leitungen in der Strommessschaltung

Leitung	Beschreibung
I_SENS_CH_x	Spannung am Shunt-Widerstand des Kanals x, dient als Eingangssignal für den Multiplexer.
I_SENS_VCC	Versorgungsspannung für die Strommesseinheit (INA, MUX, etc.).
SWITCH_1_2	Steuersignal für die parallele Umschaltung der Kanäle 1 und 2.
SWITCH_3	Steuersignal für die Umschaltung zwischen Kanalgruppe 1/2 und 3/4.
I_SENS_OUT	Ausgang des Instrumentenverstärkers, wird direkt vom ADC des Mikrocontrollers erfasst.

Zunächst wird der Shunt-Widerstand dimensioniert. Als Instrumentenverstärker wird der Typ INA181A2IDBVR ausgewählt. Dieser ist bei JLCPCB in großer Stückzahl verfügbar und hat ein ausreichend kleines Gehäuse. Zudem stammt der Baustein vom US-amerikanischen Halbleiterhersteller Texas Instruments, was dem Ziel entgegenkommt,

möglichst auf chinesische Komponenten zu verzichten. Die Serie INA181 ist mit Verstärkungen $A = 20, 50, 100$ oder 200 erhältlich. Die Slew-Rate beträgt laut Datenblatt [8] unabhängig von der Verstärkung $2\text{ V}/\mu\text{s}$.

Tabelle 21: Auswirkungen unterschiedlicher Verstärkungsfaktoren auf die Strommessung

Verstärkung	Auswirkung auf die Schaltung
20	Weniger Rauschen \rightarrow größerer Shunt-Widerstand \rightarrow höhere Verlustleistung
50	Gute Verfügbarkeit, günstiger Preis \rightarrow sinnvoller Kompromiss
200	Mehr Rauschen \rightarrow kleinerer Shunt-Widerstand \rightarrow geringere Verlustleistung

Aus der Festlegung des maximalen Stroms im Kapitel Flachbandkabel folgt, dass ein Strom von maximal einem Ampere durch einen Shunt fließen darf. Dabei soll der Verstärker die maximale Referenzspannung von $3,3\text{ V}$ erreichen.

Zur Bestimmung des benötigten Shunt-Werts wird rückwärts gerechnet:

$$U_{\text{shunt}} = \frac{U_{\text{out}}}{A} = \frac{3,3\text{ V}}{50} = 66\text{ mV} \quad (18)$$

$$R_{\text{shunt}} = \frac{U_{\text{shunt}}}{I_{\text{max}}} = \frac{66\text{ mV}}{1\text{ A}} = 66\text{ m}\Omega \quad (19)$$

$$U_{\text{LSB}} = \frac{3,3\text{ V}}{2^{12}} = 0,806\text{ mV}, \quad U_{\text{shunt,LSB}} = \frac{U_{\text{LSB}}}{A} \approx 16,1\text{ }\mu\text{V} \quad (20)$$

$$I_{\text{LSB}} = \frac{U_{\text{shunt,LSB}}}{R_{\text{shunt}}} \approx 215\text{ }\mu\text{A} \quad (21)$$

$$P_{\text{max}} = I_{\text{max}}^2 \cdot R_{\text{shunt}} = (1\text{ A})^2 \cdot 75\text{ m}\Omega = 75\text{ mW} \quad (22)$$

$$I_{\text{max,theor}} = \sqrt{\frac{250\text{ mW}}{75\text{ m}\Omega}} \approx 1,826\text{ A} \quad (23)$$

Nun richtet sich der Blick nun auf die eigentliche Messarchitektur. Im nächsten Abschnitt werden daher zwei Messarten gegenübergestellt:

- **Option 1: Abtastung des gepulsten Stroms**

Der Strom wird während des PWM-Signals direkt gemessen. Dabei muss die Signalfalte schnell genug ansteigen, um eine präzise Abtastung innerhalb der PWM-Periode zu ermöglichen.

- **Option 2: Mittelwertmessung über einen Tiefpass**

Der PWM-Strom wird durch einen Tiefpass geglättet, sodass ein analoger Mittelwert entsteht. Dieser wird vom ADC abgetastet und ermöglicht eine hohe Auflösung ohne auf schnelle Signalflanken angewiesen zu sein.

5.4.2 Option 1: Abtastung des gepulsten Stroms

Die Schaltung wird mit einer internen Systemuhr von 64 MHz betrieben. Der PWM-Timer ist auf 0xFFFF = 4095 Schritte konfiguriert. Damit ergibt sich die PWM-Frequenz zu:

$$f_{\text{PWM}} = \frac{64 \text{ MHz}}{4096} = 15,625 \text{ kHz} \quad (24)$$

Der Strom durch jeden Kanal wird über den Tastgrad gesteuert. Bei einem Tastgrad von 100 % fließt ein maximaler Strom von 1 A durch den jeweiligen Shunt.

Die maximale Ausgangsspannung von 3,3 V führt bei einer Slew-Rate von 2 V/µs zu:

$$t_{\text{slew}} = \frac{3,3 \text{ V}}{2 \text{ V}/\mu\text{s}} = 1,65 \mu\text{s} \quad (25)$$

Zusätzlich muss das Signal für mindestens 1,5 ADC-Takte stabil bleiben:

$$t_{\text{sample}} = 1.5 \cdot \frac{1}{10,67 \text{ MHz}} = 140,6 \text{ ns} \quad (26)$$

Die gesamte benötigte Signaldauer ist somit:

$$t_{\text{signal}} = t_{\text{slew}} + t_{\text{sample}} = 1,65 \mu\text{s} + 0,14 \mu\text{s} = 1,79 \mu\text{s} \quad (27)$$

Die PWM-Periodendauer ergibt sich zu:

$$T_{\text{PWM}} = \frac{1}{f_{\text{PWM}}} = \frac{1}{15,625 \text{ kHz}} = 64 \mu\text{s} \quad (28)$$

Daraus folgt ein minimaler Duty-Cycle:

$$D_{\min} = \frac{t_{\text{signal}}}{T_{\text{PWM}}} = \frac{1,79 \mu\text{s}}{64 \mu\text{s}} \approx 2,8 \% \quad (29)$$

Der minimal sicher messbare Strom beträgt:

$$I_{\min} = 1 \text{ A} \cdot 2,8 \% = 28 \text{ mA} \quad (30)$$

Die kleinste vom ADC detektierbare Spannung:

$$\Delta U_{\text{ADC}} = \frac{3,3 \text{ V}}{4096} = 0,806 \text{ mV} \quad (31)$$

Bei Verstärkung $A = 50$:

$$U_{\text{shunt,min}} = \frac{\Delta U_{\text{ADC}}}{A} = \frac{0,806 \text{ mV}}{50} = 16,13 \mu\text{V} \quad (32)$$

Mit $R_{\text{shunt}} = 66 \text{ m}\Omega$:

$$I_{\min} = \frac{U_{\text{shunt,min}}}{R_{\text{shunt}}} = \frac{16,13 \mu\text{V}}{66 \text{ m}\Omega} \approx 244 \mu\text{A} \quad (33)$$

Die Konversionszeit des ADC:

$$t_{\text{conversion}} = 14 \cdot \frac{1}{10,67 \text{ MHz}} = 1,3125 \mu\text{s} \quad (34)$$

Anzahl der ADC-Messungen pro PWM-Periode:

$$N_{\text{samples}} = \frac{64 \mu\text{s}}{1,3125 \mu\text{s}} \approx 48 \quad (35)$$

Effektive Auflösung durch Oversampling:

$$\text{Effektive Bits} = 12 + \log_2(48) \approx 12 + 5.6 = 17.6 \text{ Bit} \quad (36)$$

Dies ist praktisch gesehen unwahrscheinlich, weil man dafür ein geeignetes Rauschen auf dem Signal bräuchte. Außerdem ist durch die *Slew Rate* die minimale t_{signal} -Dauer der Variante ohne Tiefpassfilter 1,79 μs . Das bedeutet, bei der T_{PWM} könnte man nur 35 Messungen durchführen. Weil die Schaltung zur Überwachung und Kalibrierung benötigt wird, liegt die Priorität auf der Messwertauflösung.

5.4.3 Option 2: Mittelwertmessung über einen Tiefpass

Welche Anforderungen gibt es an den Tiefpass?

- Das *Nyquist*-Kriterium muss erfüllt sein.
- Optimierter Ripple für Oversampling (≤ 1 LSB).
- Möglichst hohe Grenzfrequenz.
- Geringste Verzögerung.
- Das Signal darf nicht schneller steigen als die Slew-Rate zulässt.
- Bauteile dürfen nicht zu groß werden.

Wie soll der Filter ausgelegt werden? Abbildung 28 zeigt das Konzept bereits mit den berechneten Ergebnissen.

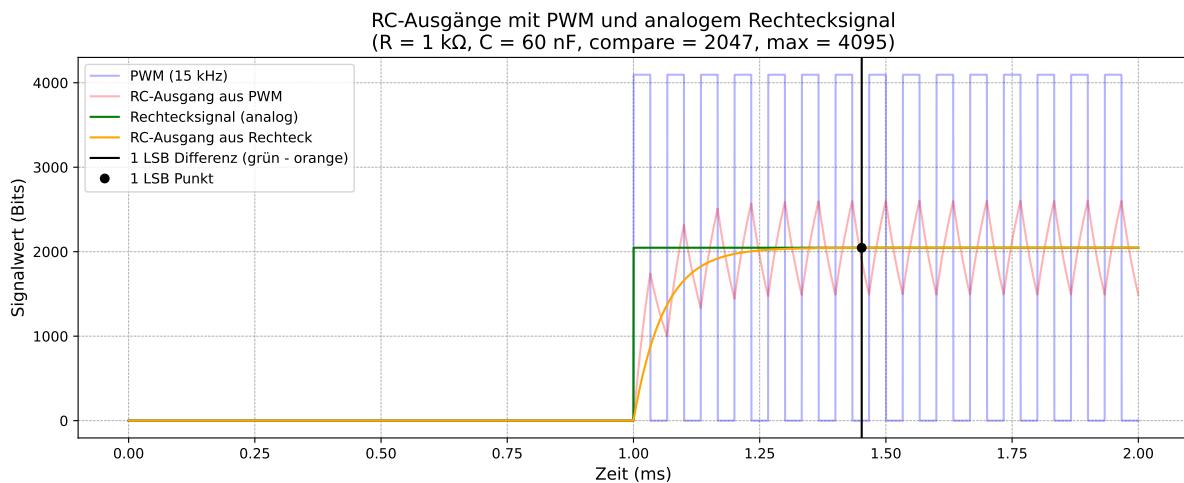


Abbildung 28: Konzeptioneller Verlauf von PWM, Sprungsignal und Tiefpass-Ausgang. Den zugehörigen Code findet man auf [GitHub](#).

Wie schon erwähnt, arbeiten die Kanäle der Platine mit 15,625 kHz, welche in den Plots blau dargestellt sind. Außerdem findet alle 1000 µs eine Aktualisierung des **Capture-Compare-Register (CCR)**s statt. Somit ergibt sich der größtmögliche Wechsel, wenn die Tastrate (CCR) von 0 auf 0xffff gesetzt wird. Dieser Sprung wird durch die grüne Kurve dargestellt. Das rote Sägezahnsignal repräsentiert die echte analoge Spannung, die man nach dem Filter messen würde. Die gelbe Kurve stellt die Reaktion des Filters dar, wenn der Eingang eine perfekte analoge Spannung und kein PWM-Signal wäre.

Ein wesentlicher Nachteil eines Tiefpass-Filters besteht in seiner Signalverzögerung. Der Ausgang kann erst dann den Mittelwert des gepulsten Eingangsstroms abbilden, wenn der Kondensator ausreichend „aufgeladen“ ist. Die Schaltung soll jede Millisekunde einen gültigen Messwert liefern. Deshalb wird der Filter so dimensioniert, dass er sich innerhalb von 500 µs (= $\frac{1}{2}$ PWM-Periode) bis auf 1 LSB an seinen Endwert annähert. Eine vollständige Aufladung ist physikalisch nicht erreichbar. Sie ist jedoch nicht erforderlich, da die zweite Hälfte oszilliert das PWM-Signal um den Mittelwert. [Abbildung 29](#) visualisiert die relevanten Zeitbereiche und zeigt, wie der Tiefpass das hochfrequente Tastsignal in eine nahezu gleichförmige Spannung überführt.

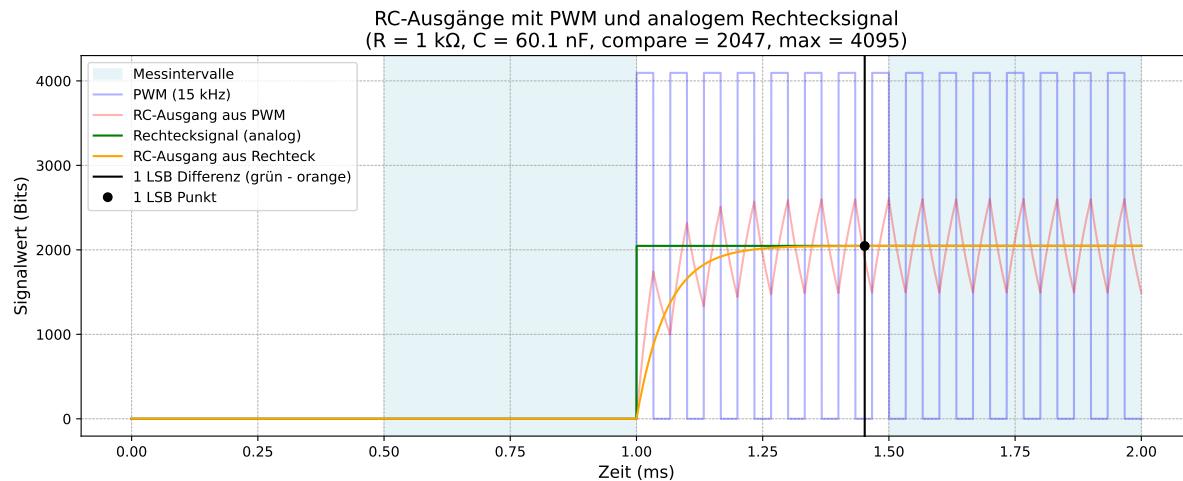


Abbildung 29: Visualisierung der Messintervalle eines gefilterten PWM-Signals

Es sind so viele Punkte, dass sie auf dem Plot nicht mehr darstellbar sind und daher als farbliche Bereiche dargestellt werden. Je nach Genauigkeitsbedarf, kann man die Anzahl der Messungen anpassen.

Der ADC hat einen eigenen Arbeitstakt, welcher ein verlangsamter Prozessortakt ist. Der ADC kann alle 14 ADC-Takte eine Messung durchführen. Bei einer ADC-Taktfrequenz von 10,67 MHz ergibt sich eine minimale Konversionszeit von:

$$t_{\text{conversion}} = 14 \cdot \left(\frac{1}{10,67 \text{ MHz}} \right) = 14 \cdot 93,75 \text{ ns} = 1,3125 \mu\text{s} \quad (37)$$

Somit kann man innerhalb der Messintervalls 500 us / 1,3125 us Messungen = 380 Messungen durchführen.

Zu Beginn wurde geschrieben, dass der Ripple möglichst klein sein soll. In diesem Modell ist er jedoch sehr groß. Es gibt folgende Möglichkeiten diesen zu verkleinern: Option Auswirkung Tau vergrößern Signal kann nicht mehr rechtzeitig aufgeladen werden Ein weiterer Tiefpass vor dem ADC Weitere Bauteile Oversampling Microcontroller Unit (MCU)

muss mehr arbeiten. Da innerhalb jedes Messintervalls viele Werte aufgenommen werden, eignet sich Oversampling besonders gut. Anschließend kann das Signal digital über einen inkrementellen Mittelwert geglättet werden. Dieses Verfahren ist speicher- und recheneffizient, da keine Zwischenspeicherung aller Messwerte erforderlich ist. Die folgende Formel beschreibt das Verfahren: Der aktuelle Mittelwert mean_n wird aus dem vorherigen Mittelwert mean_{n-1} und dem neuen Messwert x_n berechnet:

$$\text{mean}_n = \text{mean}_{n-1} + \frac{x_n - \text{mean}_{n-1}}{n} \quad (38)$$

Im folgenden Plot wird der Prozess der Mittelwerberechnung dargestellt.

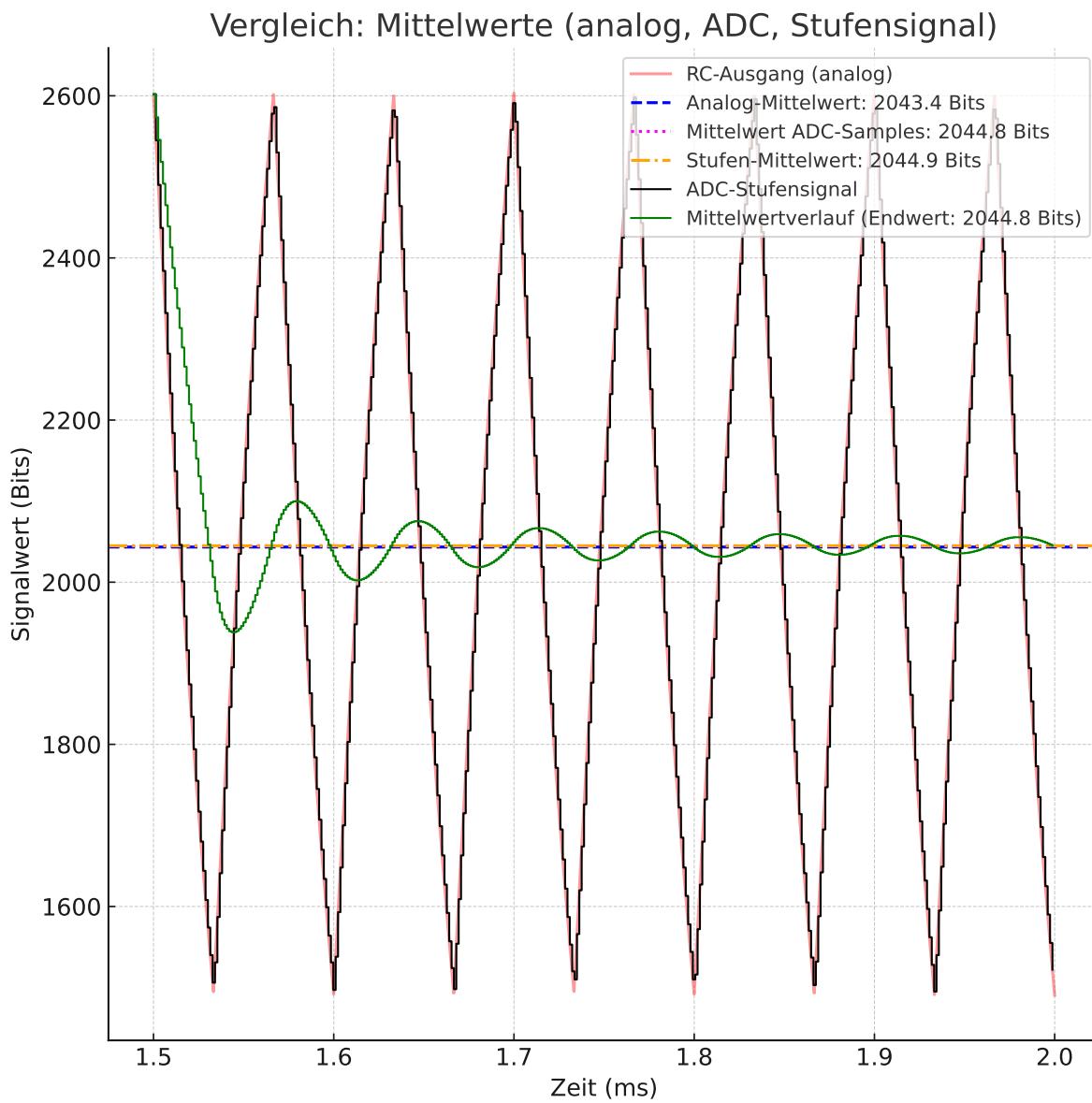


Abbildung 30: Darstellung des digitalen Mittelwertverfahrens mit Oversampling

Das Verfahren hat eine Schwäche: Der durch das PWM-Signal verursachte Ripple erzeugt steile Flanken im RC-gefilterten Signal. Dadurch kommt es bei der Quantisierung zu einem Fehler von bis zu etwa 1,5 Bits in dieser Simulation.

- Analog-Mittelwert: 2043,4 Bits
- Stufenmittelwert: 2044,9 Bits
- Fehler: Echter Mittelwert – Stufenmittelwert = 1,5 Bits

Dieser Effekt ist abhängig vom PWM-Tastgrad. Der Ripple ist typischerweise am größten bei etwa 50 % Tastgrad und nimmt bei einem sehr kleinen oder sehr großen Tastgrad ab. Das bedeutet, im unteren Messbereich, also bei kleinem Tastgrad, ist der Ripple geringer und damit auch der Quantisierungsfehler entsprechend kleiner.

Im Folgenden wird τ berechnet. Dabei soll τ so ausgewählt werden, dass die Aufladung ein LSB vom Sollwert entfernt ist nach 500 μs . Dafür wird die Kondensator-Auflade-Funktion benutzt und nach τ umgeformt:

$$U_{\text{out}} = U_{\text{in}} \cdot \left(1 - \exp\left(-\frac{t}{\tau}\right)\right) \quad (39)$$

$$\tau = -\frac{t}{\ln\left(1 - \frac{U_{\text{out}}}{U_{\text{in}}}\right)} \quad (40)$$

Es wird definiert, dass die Aufladung $0xFFFF - 1$ LSB erreichen soll. Es wird festgelegt:

- Auflösung: 12 Bit \rightarrow Maximalwert: $0xFFFF = 4095$
- 1 LSB darunter: 4094

$$\frac{U_{\text{out}}}{U_{\text{in}}} = \frac{4094}{4095} \quad (41)$$

$$t = 500 \mu\text{s} \quad (42)$$

Eingesetzt ergibt das:

$$\tau = -\frac{500 \cdot 10^{-6}}{\ln\left(1 - \frac{4094}{4095}\right)} = 60,10 \mu\text{s} \quad (43)$$

Für die Funktion genügt dieser τ -Wert bei einem Tiefpass erster Ordnung. Aber um das Ripple zu verbessern, wird in der tatsächlichen Schaltung ein zweiter Tiefpass am ADC-Eingang eingebaut. Das τ gilt für beide Filter und wird berechnet aus der Formel für den Filter zweiten Grades:

$$V(t) = V_{\text{in}} \cdot \left[1 - \left(1 + \frac{t}{\tau}\right) \cdot \exp\left(-\frac{t}{\tau}\right)\right] \quad (44)$$

Die Lösung wurde numerisch mit Python berechnet und ergibt $\tau = 72,10 \mu\text{s}$. Das zugehörige Skript findet man auf [GitHub](#).

Der folgende Plot stellt die Auswirkungen dar. Der Fehler ist nicht mehr erkennbar. Betrachtet man die Y-Achse, ist erkennbar, dass der Ripple etwa um den Faktor zehn kleiner geworden ist im Vergleich zum vorherigen Bild ([Abbildung 30](#)).

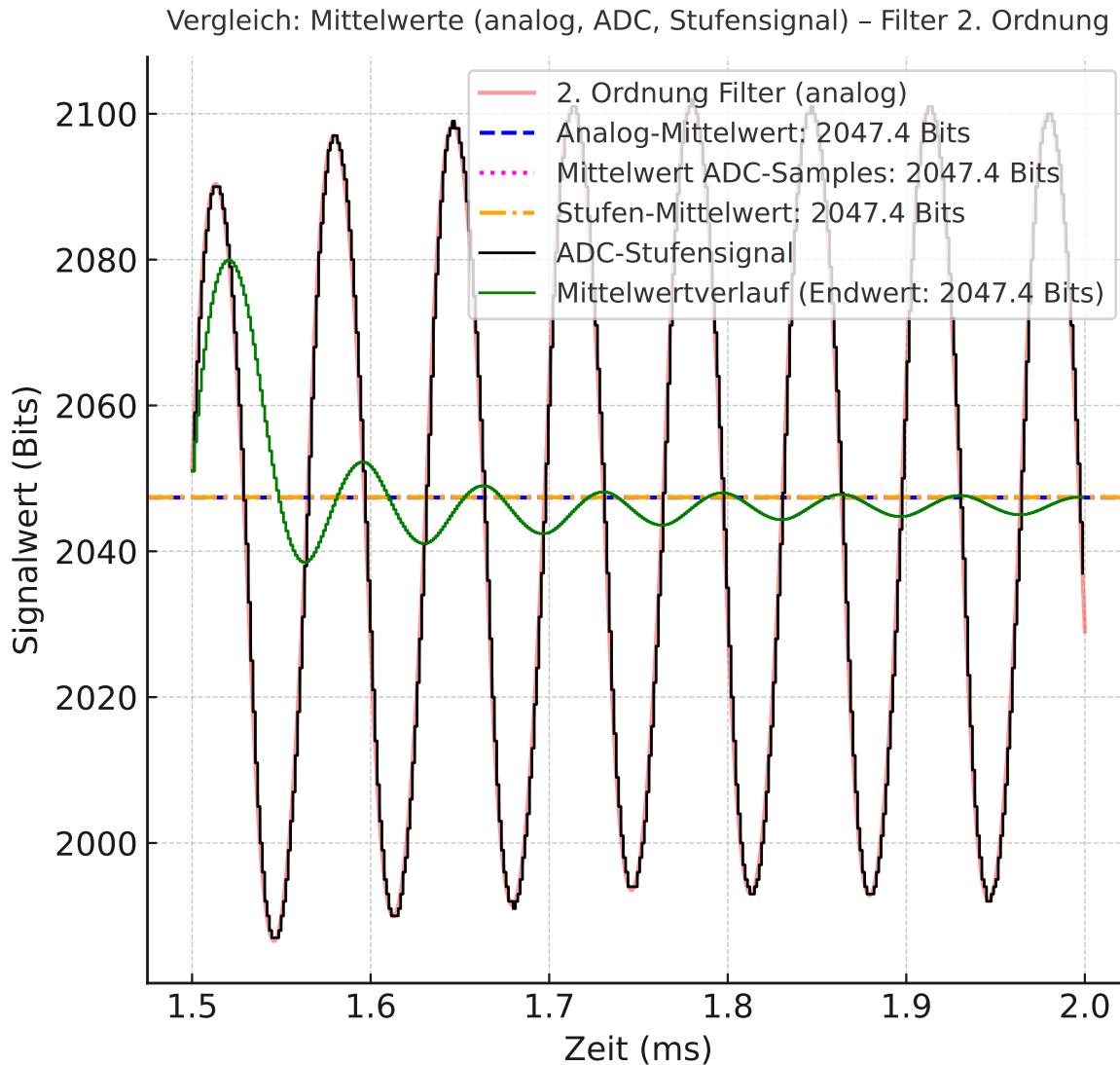


Abbildung 31: Verbesserte Glättung des PWM-Signals durch zweite Filterstufe

Nachfolgend wird gezeigt, dass die Slew-Rate in beiden Filtervarianten nicht verletzt wird. Diese Überlegungen erfolgten vor der Dimensionierung von τ .

Gegeben sind [9]:

$$\text{SR} = \frac{dv}{dt} = \frac{i}{C}, \quad \Delta v = 3,3 \text{ V}, \quad \text{SR} = 2 \text{ V}/\mu\text{s} \quad (45)$$

Berechnung der minimal benötigten Anstiegszeit:

$$\Delta t = \frac{\Delta v}{\text{SR}} = \frac{3,3 \text{ V}}{2 \text{ V}/\mu\text{s}} = 1,65 \mu\text{s} \quad (46)$$

Daraus ergibt sich ein minimal erforderliches Zeitkonstante:

$$\tau_{\min} = \Delta t = 1,65 \mu\text{s} \quad (47)$$

Wie im folgenden Plot zu sehen ist, liegt das verwendete $\tau = 60,10 \mu\text{s}$ weit entfernt von der kritischen Slew-Rate-Grenze. Zusätzlich wurde das spätere $\tau = 72,10 \mu\text{s}$ mit dargestellt.

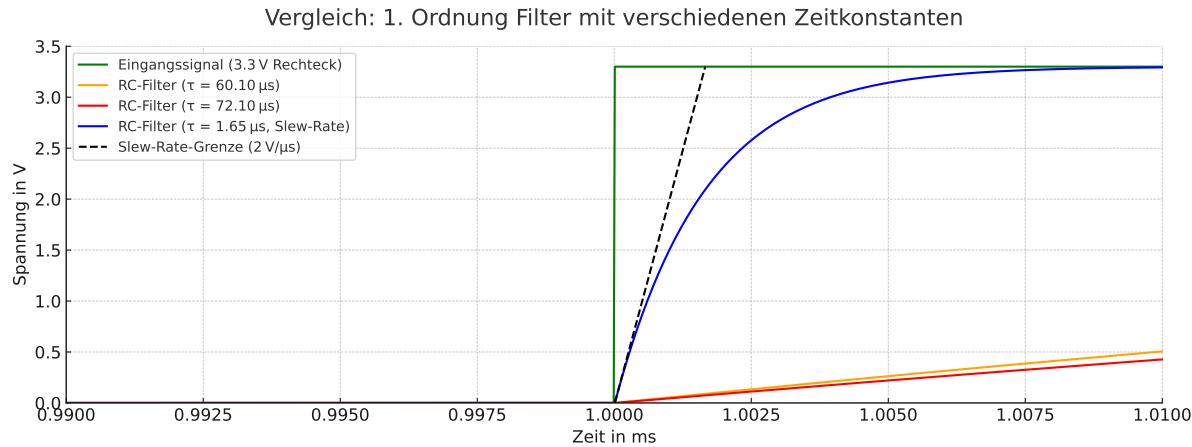


Abbildung 32: Nachweis, dass die Slew-Rate in beiden Filtervarianten nicht verletzt wird

Auswahl der Komponenten

Zum Schluss müssen die Komponenten ausgewählt werden. Die parasitären Kapazitäten des ADC und des INA liegen nur im Pikofarad-Bereich. Daher wird die Kapazität des RC-Filters größer gewählt (zwischen 1 nF und 10 nF), um die parasitären Kapazitäten der Bauteile zu überlagern. Der Filterwiderstand wird im Bereich von 1 kΩ bis 10 kΩ gewählt. Aufgrund des hohen Eingangsbiasstroms des INA würde über diesen Widerstand ein Spannungsabfall entstehen, der durch die Verstärkung des INA erheblich vergrößert und so die Messung verfälschen würde. Dieses Problem wird gelöst, indem hinter dem RC-Glied ein Impedanzwandler in Unity-Gain-Konfiguration (TLV9061IDBVR) eingesetzt wird.

Es wurde ein 10 nF-Kondensator mit einer Toleranz von 1% gewählt. Dazu kommt ein Widerstand mit 72,1 kΩ. Tatsächlich wurde ein günstiger 69,8 kΩ-Widerstand mit 0,1%-Toleranz verwendet. Dieser wurde bewusst etwas kleiner gewählt, um im Zweifelsfall früher aufgeladen zu werden als zu spät, wobei sich der Ripple geringfügig erhöht. Im Folgenden wird überprüft, ob die Zeitkonstante τ durch die Toleranzen weiterhin im gewünschten Bereich bleibt.

Tabelle 22: Berechnungsgrundlage zur Komponentenwahl und Toleranzbetrachtung

Parameter	Wert	Bereich durch Toleranz
Ziel- τ	72,10 μs	69,03 μs – 70,57 μs
Kondensator C	1 nF ±1 %	0,99 nF – 1,01 nF
Widerstand R	69,8 kΩ ±0,1 %	69,7302 kΩ – 69,8698 kΩ

$$\tau_{\min} = R_{\min} \cdot C_{\min} = 69,7302 \cdot 10^3 \Omega \cdot 0,99 \cdot 10^{-9} F = 69,03 \mu s \quad (48)$$

$$\tau_{\max} = R_{\max} \cdot C_{\max} = 69,8698 \cdot 10^3 \Omega \cdot 1,01 \cdot 10^{-9} F = 70,57 \mu s \quad (49)$$

Wie aus Tabelle 22 hervorgeht, bleibt die tatsächliche Zeitkonstante in jedem Fall kleiner als das Ziel- τ von 72,10 μs und erfüllt somit die Anforderungen an die Ladezeit des Tiefpassfilters.

5.5 Datenaufbereitung

5.5.1 Light Source Extractor

Der *Light Source Extractor* ist ein Programm, das im Sommersemester 2024 im Rahmen des Moduls *Digitale Bildverarbeitung* unter der Betreuung von Prof. Dr.-Ing. Marcus Purat an der BHT Berlin entwickelt wurde. Die folgende Grafik zeigt das Programm nach dem Start.

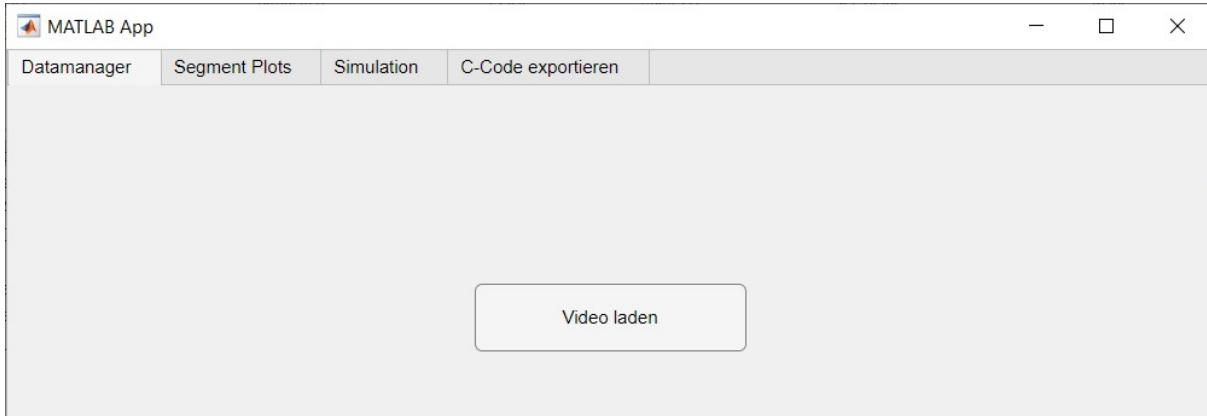


Abbildung 33: Benutzeroberfläche des Light Source Extractors beim Start

Ziel der Software ist es, aus Videomaterial die zeitliche Helligkeitsverteilung zu extrahieren und als Grundlage für die weitere algorithmische Verarbeitung zu nutzen. Diese Signale werden als Matrix in einer .c-Datei gespeichert, sodass sie auf einem Mikrocontroller mithilfe eines Algorithmus aufbereitet und anschließend per PWM wiedergegeben werden können.

Im Rahmen dieser Bachelorarbeit wurde die bestehende Software weiterentwickelt mit dem Ziel, die Funktionalität zu erweitern, um die Signalverarbeitung auf dem Mikrocontroller zu optimieren. Dazu gehört unter anderem die Integration einer glssrgb-Dekodierung des Signals.

Ferner wurde die Exportfunktion überarbeitet. In diesem Zusammenhang wurde die Datenstruktur der .c-Datei neu definiert. Zudem werden zusätzliche Metadaten (u. a. `fps` und `fade_frame`) eingebunden.

5.5.2 Videoimport

Der Benutzer lädt ein Video hoch, das eine Lichtquelle zeigt, z. B. eine Kerzenflamme, einen Feuerwerkskörper oder eine Wunderkerze. Je nach Brennverhalten der Flamme – radial oder gerichtet – reagiert der Algorithmus unterschiedlich.

Zunächst wird der Kerzendochter getrackt, um eine stabile Referenzposition zu erhalten. Gleichzeitig wird für jeden Frame die Flamme ausgeschnitten, um unnötige Hintergrundinformationen zu entfernen und die Bewegung der Lichtquelle präzise zu erfassen.

Es werden zwei verschiedene Lichtquellenarten unterschieden:

- ein Licht mit stark streuender Helligkeit um einen zentralen Punkt
- eine Lichtquelle mit klaren Konturen und definierter Brennrichtung

Je nach Lichtquelle werden zwei verschiedene Bildverarbeitungspipelines angewendet.

5.5.3 Linearisierung des Farbraums von Videosignalen

Zu Beginn wird die Szene mit einer Videokamera aufgenommen. Das Video wird in einem 8-Bit-Format (z. B. .mp4) gespeichert. Dabei wird die reale, physikalisch lineare Helligkeit in einen sogenannten Gamma-korrigierten Bereich überführt, welcher der nichtlinearen Wahrnehmung des menschlichen Auges angepasst ist.

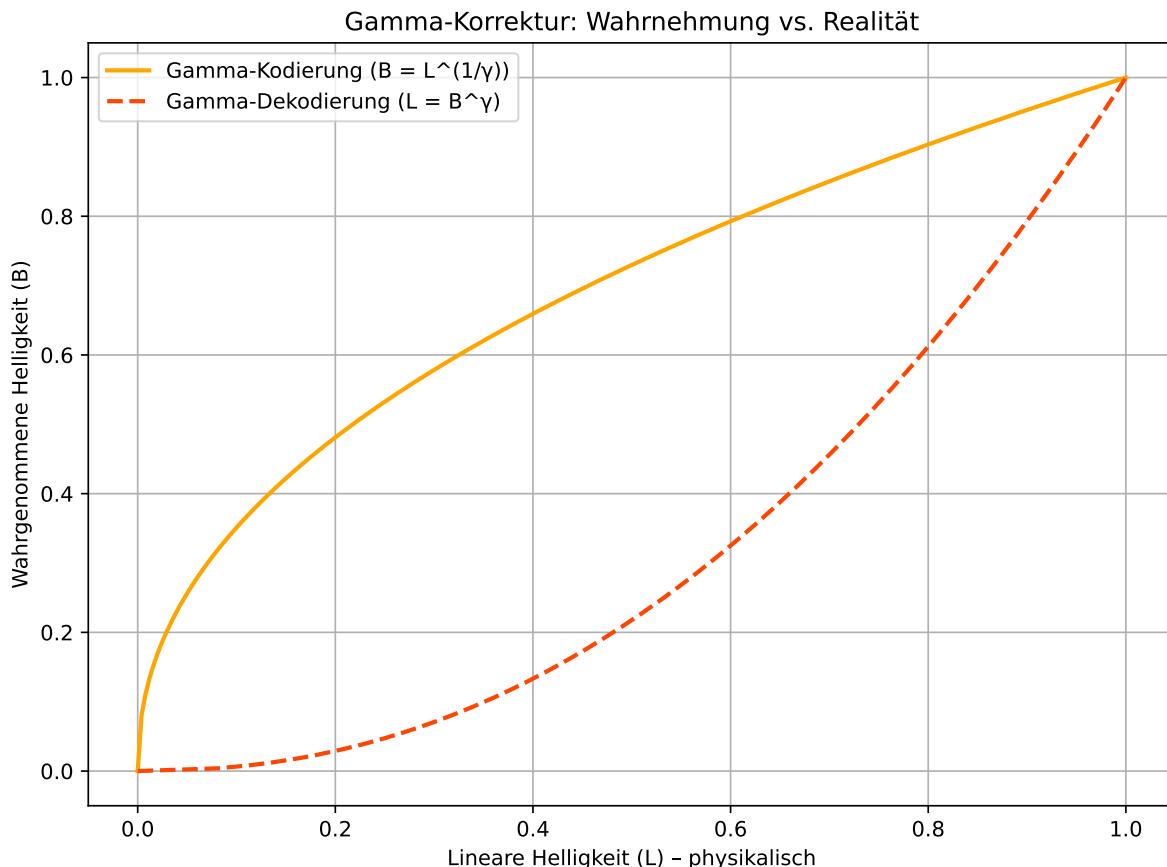


Abbildung 34: Zusammenhang zwischen linearer und wahrgenommener Helligkeit

Das menschliche Auge nimmt bei geringen Lichtniveaus Unterschiede besonders stark wahr, während es bei hoher Helligkeit weniger empfindlich ist. Auch bei digitalen Bildern, etwa JPEGs, sind die Pixelwerte im Bereich 0–255 gamma-korrigiert. Der Wert 128 (die Hälfte von 255) wird vom Auge etwa als 50% Helligkeit wahrgenommen.

Obwohl der sRGB-Standard (IEC 61966-2-1) eine präzisere Transferfunktion vorsieht, genügt im Rahmen dieser Arbeit die Verwendung eines vereinfachten Gamma-Werts von $\gamma = 2.2$. Die wahrgenommene Helligkeit V ergibt sich aus der physikalischen Helligkeit L durch:

$$V = L^\gamma \quad \Leftrightarrow \quad L = V^{1/\gamma} \quad (50)$$

Gamma-Bereich**Linearer Bereich****Tabelle 23:** Gegenüberstellung von **Gamma-Bereich** und **linearer Bereich**

Das **Standard Red Green Blue (sRGB)**-Bild wirkt in dunklen Bereichen dunkler, da diese gestreckt und helle Bereiche komprimiert werden – das entspricht der menschlichen Wahrnehmung. „Any LED responds (almost) linearly to pulse-width modulation, but the human eye is dramatically nonlinear.“[10] Ohne Linearisierung würden Operationen wie Mittelwertbildung, Skalierung oder Interpolation zwischen Frames verzerrt werden.

5.5.4 Pipeline 1: Radiale Flammen

In der folgenden Grafik wird die Funktionsweise der ersten Pipeline erklärt. Man sieht ein Frame aus einem Video einer Wunderkerze. Die Markierungen veranschaulichen die Arbeitsweise der Pipeline.

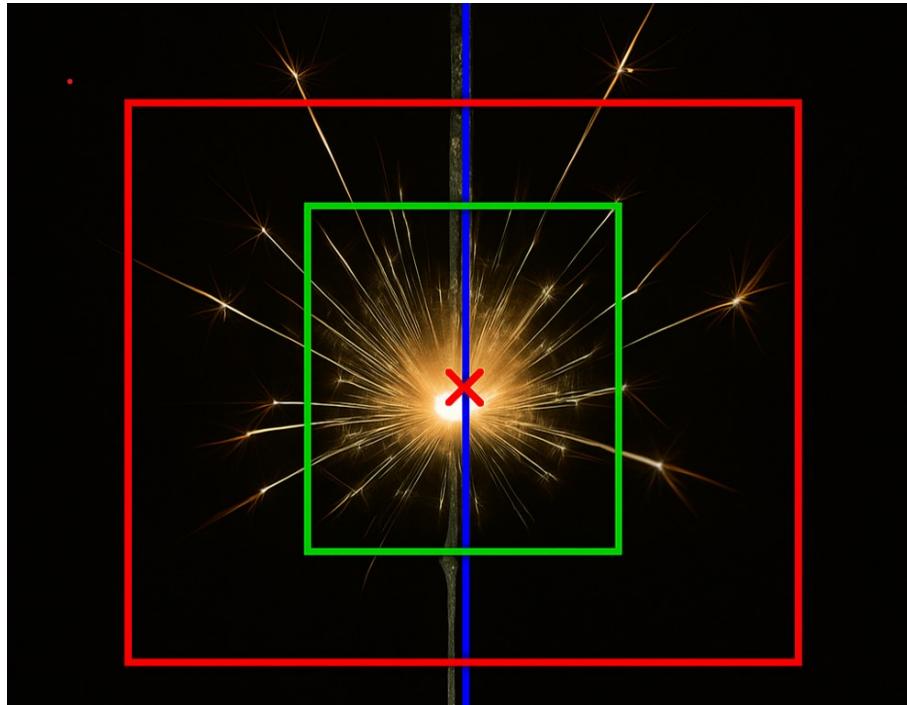


Abbildung 35: Frame mit grünem Ausschnitt (relevante Lichtinformation) und blauer Tracking-Bahn

Der grüne Bereich in diesem Frame ist der Ausschnitt, der die relevanten Lichtinformationen enthält, etwa 99 % je nach Konfiguration. Weitere Lichtpunkte außerhalb des grünen Bereichs werden ignoriert, damit man den grünen Rahmen nicht ins Unendliche erweitert. Der grüne Rahmen verändert sich in jedem Frame in Bezug auf die Position zum Trackingpunkt und die geometrische Form. Dabei wird der Trackingpunkt in jedem Frame neu bestimmt und markiert den Ursprung der Lichtquelle. Die blaue Linie stellt dabei die Laufbahn des Trackingpunkts über das gesamte Video dar.

Nachdem für jedes Frame der grüne Rahmen und alle Trackingpunkte berechnet wurden, wird der rote Rahmen – auch *Region-of-interest* (ROI) genannt – bestimmt. Der grüne Bereich verlässt im gesamten Videoverlauf niemals die *Region of Interest* (ROI), dargestellt durch den roten Rahmen.

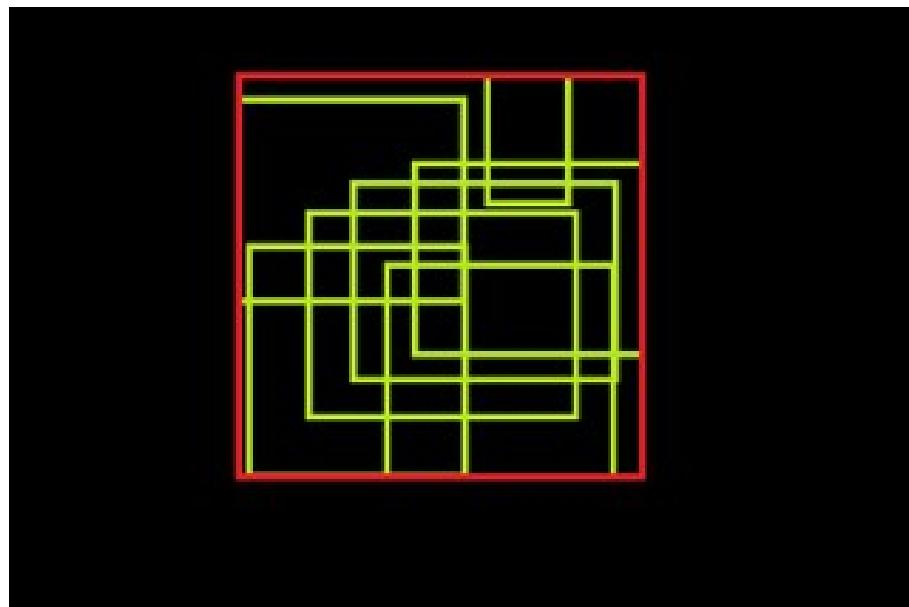


Abbildung 36: Bestimmung der **ROI** (roter Rahmen) aus allen grünen Ausschnitten

Er stellt den Bereich dar, der später in jedem Frame ausgeschnitten wird.

Je nach ausgewählter Pipeline liegt der Trackingpunkt auf einer anderen Position in der ROI.

Tabelle 24: Position des Trackingpunkts in den beiden Pipelines

Pipeline 1	Pipeline 2
A black square with a red rectangular frame around it. A red star-shaped marker is located near the center of the frame.	A black square with a red rectangular frame around it. A red star-shaped marker is located near the bottom edge of the frame.

Der rote Punkt markiert die Flammenquelle. Die blaue Linie beschreibt die Laufbahn der Trackingpunkte.

5.5.5 Pipeline 2: Gerichtete Flammen

Die folgende Grafik zeigt die Arbeitsweise des Programms mit einer Tafelkerze.

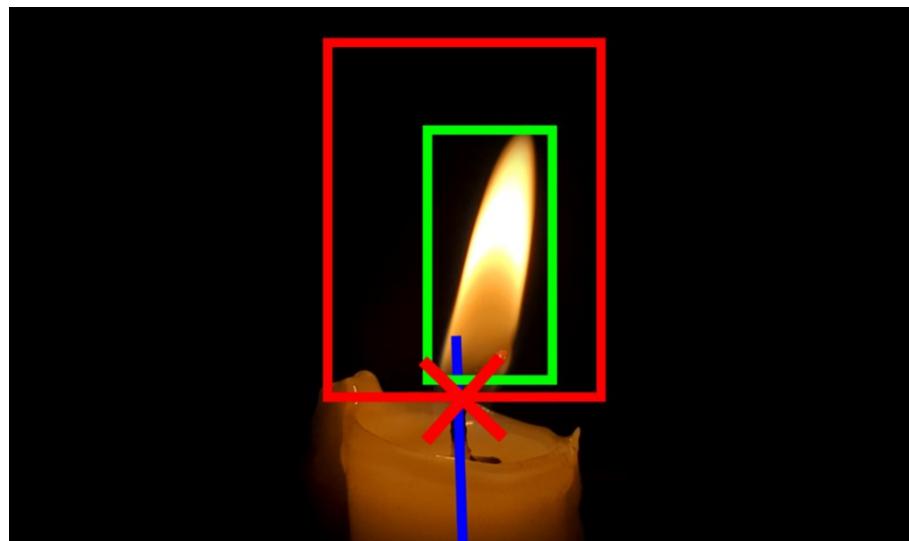


Abbildung 37: Frameanalyse einer Tafelkerze (Pipeline 2)

Zuerst wird die Bewegung der Kerze analysiert. Das gesamte Frame wird mit Hilfe eines Thresholdfilters in Lichtregionen unterteilt.



Abbildung 38: Segmentierung in Lichtregionen

Bei einer brennenden Kerze gibt es die Flamme und den Kerzenschaft. Es wird davon ausgegangen, dass sich der Ursprung der Lichtquelle in der hellsten Region befindet. Der grüne Rahmen umfasst die hellste Region. Im Gegensatz zu Pipeline 1 wird hier nicht der hellste Punkt der hellsten Region gesucht, sondern der über mehrere Frames ruhigste Punkt.

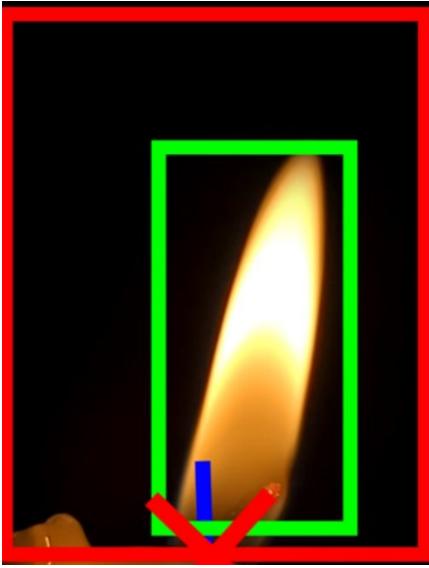
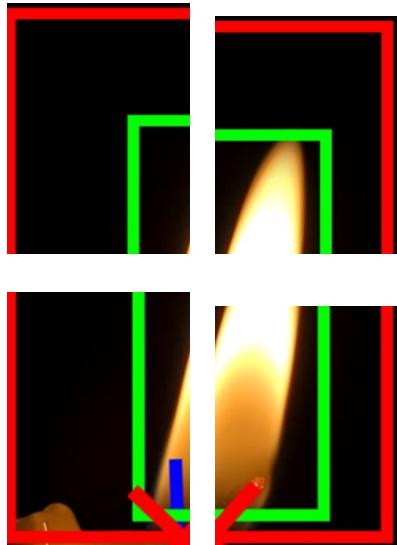
Originalbild (Pipeline 2)	Segmentierung in vier Quadranten
	

Tabelle 25: Strukturierte Zerlegung des ROI-Bereichs in vier gleich große Bildsegmente (Quadranten). Links: Original-Frame mit markiertem ROI; rechts: 2×2 -Raster zur Identifikation der ruhigsten Lichtregion.

Wie in der Abbildung zu sehen, sind die beiden linken Segmente fast dunkel. Das liegt entweder daran, dass das Video schief oder bei kontinuierlichem Wind aufgenommen wurde. Deshalb wird eine Rotationskorrektur gemacht. Dabei wird der hellste Punkt der grünen Region berechnet. Dann wird ein Vektor vom Trackingpunkt zum Mittelpunkt der Flamme gebildet. Für jedes Frame wird dieser Vektor berechnet. Am Ende wird ein Durchschnittsvektor aller Vektoren und deren Winkel berechnet. Das Video wird um diesen Winkel so korrigiert, dass der Durchschnittsvektor nach oben zeigt.

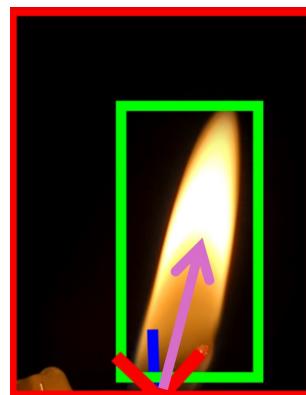


Abbildung 39: Rotationskorrektur mittels Durchschnittsvektor

5.5.6 Normierung und Segmentierung

Nach der Linearisierung des Videomaterials wird jedes einzelne Frame in vier gleich große Segmente unterteilt. Für jedes dieser Segmente wird anschließend der gemittelte Helligkeitswert berechnet.

Die so gewonnenen vier Helligkeitskanäle werden anschließend normiert: Dabei wird der global hellste Pixelwert über alle Segmente und Frames hinweg als Referenzwert (255) definiert. Alle anderen Werte werden proportional dazu skaliert.

Diese Normierung stellt sicher, dass alle Kanäle relativ zur maximal beobachteten Helligkeit vergleichbar sind.

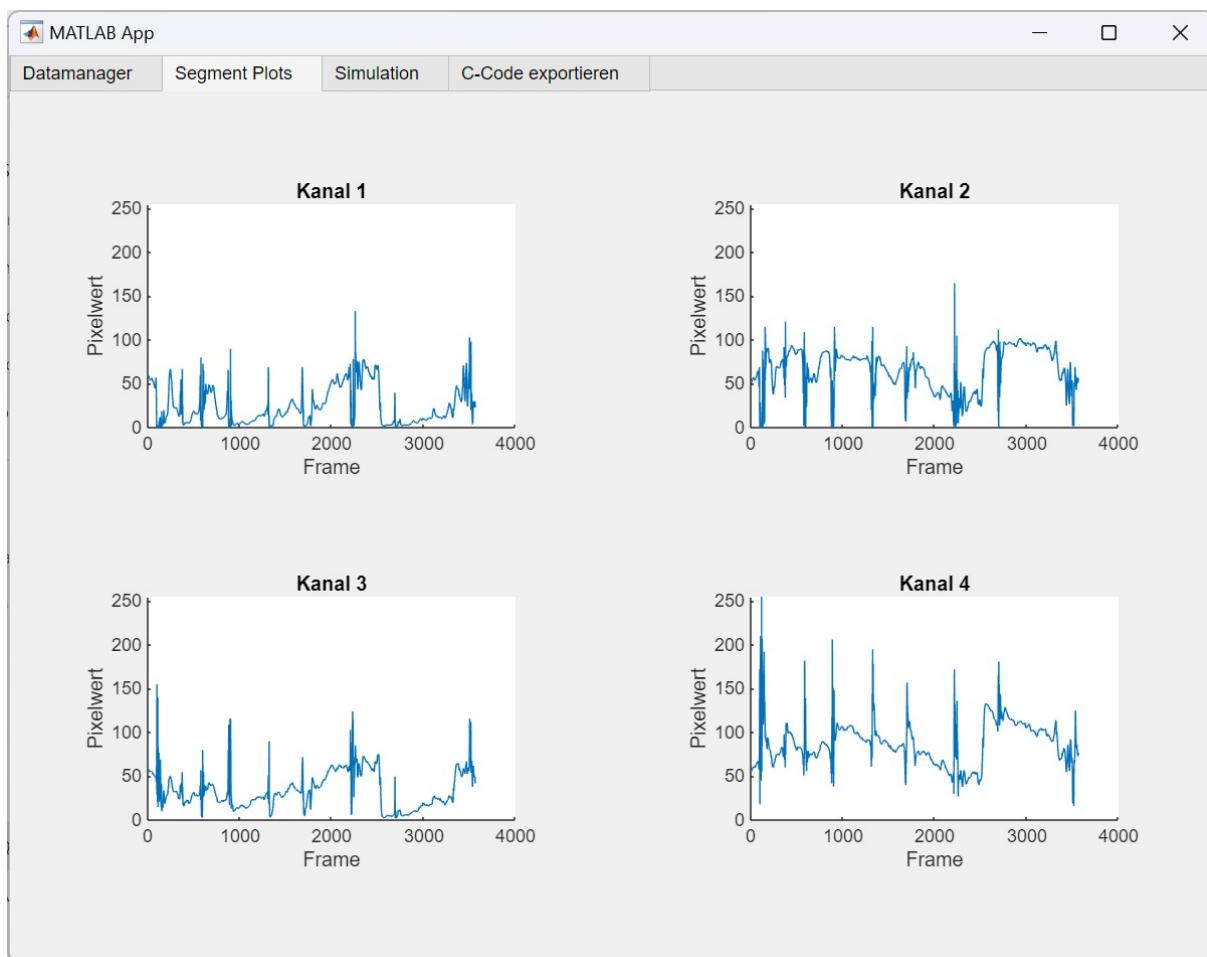


Abbildung 40: Visualisierung der normierten Helligkeitsverläufe der vier Bildsegmente über die Zeit. Die Daten dienen als Grundlage für die spätere PWM-Signalrekonstruktion.

5.5.7 Export als C-Datei

Nach der erfolgreichen Extraktion der Kanäle kann im Tab [C-Code exportieren] der Export durchgeführt werden. Dabei werden die Werte in einer Matrix gespeichert und anschließend in eine C-Datei überführt.

Im Folgenden ist die Struktur des Exports beispielhaft dargestellt:

Codeauszug	Beschreibung
<pre>static const uint8_t ch0[N] = {0, 54, 255, ...}; static const uint8_t ch1[N] = {65, 98, 65, ...}; static const uint8_t ch2[N] = {9, 25, 255, ...}; static const uint8_t ch3[N] = {127, 54, 4, ...};</pre>	Vier Arrays mit Helligkeitswerten der Kanäle.
<pre>static const LightChannels data = { .ch = { ch0, ch1, ch2, ch3 } };</pre>	Struktur, die alle vier Lichtkanäle in einem Container zusammenfasst.
<pre>const LightProgram candle = { .name = "Kerze", .length = N, .fps = 30, .min_value = 48, .max_value = 197, .fade_frames = 100, .data = &data };</pre>	Struktur für ein Lichtprogramm mit Metadaten und Referenz auf die Daten.

Tabelle 26: Struktur des C-Exports der extrahierten Lichtkanäle mit farblicher Hervorhebung

Die Daten können anschließend direkt auf den Mikrocontroller geladen werden. Dort werden sie vom Algorithmus verarbeitet und mittels Pulsweitenmodulation (PWM) als Helligkeitssignal wiedergegeben.

6 Softwareimplementierung auf dem STM32

6.1 Unendliche Aneinanderreihung von zufälligen Clips

Im Prinzip kann jeder Kanal als ein kleiner Film betrachtet werden – daher wird in diesem Abschnitt die Terminologie aus der Filmproduktion übernommen. Die Schnittarbeit mit Adobe Premiere Pro an eigenen Kurzfilmen diente dabei als Inspiration für den entwickelten Algorithmus.

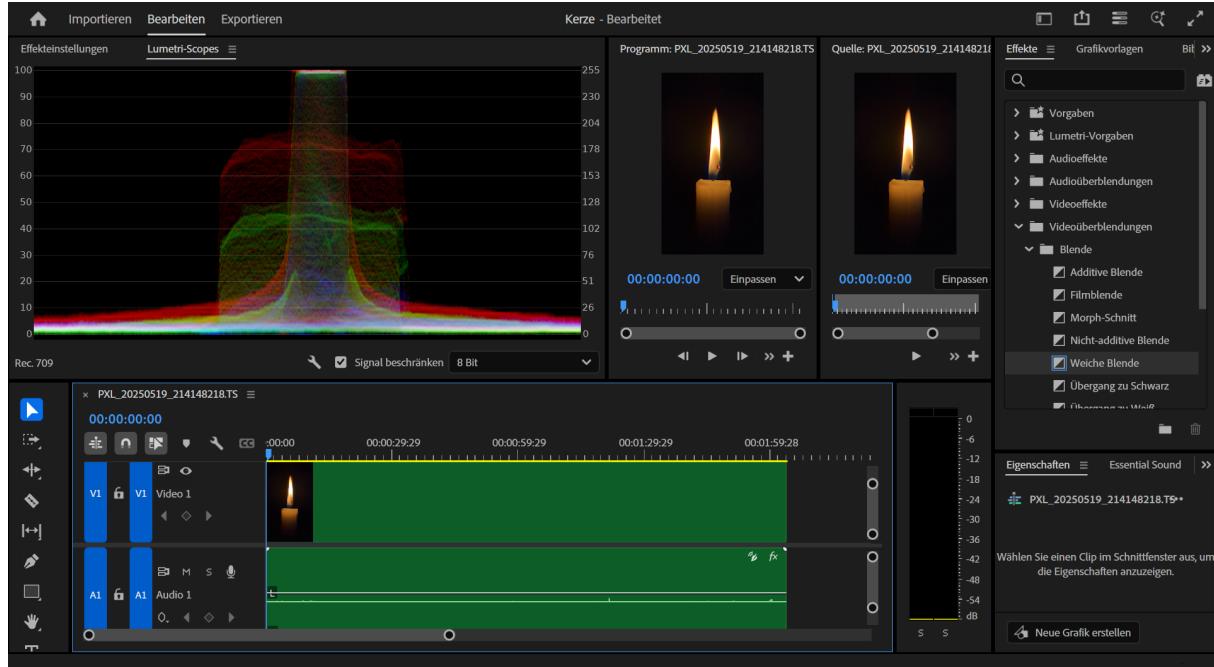


Abbildung 41: Schnittfenster in Adobe Premiere Pro mit importiertem Videomaterial einer Kerze. Das zugehörige Beispielprojekt findet man auf [GitHub](#).

Das importierte Video ist knapp zwei Minuten lang. Der Startpunkt links heißt In-Point, der Endpunkt rechts Out-Point. In den weiteren Beispielen wird lediglich das Schnittfenster gezeigt. Der grüne Streifen im Schnittfenster entspricht dem Video.

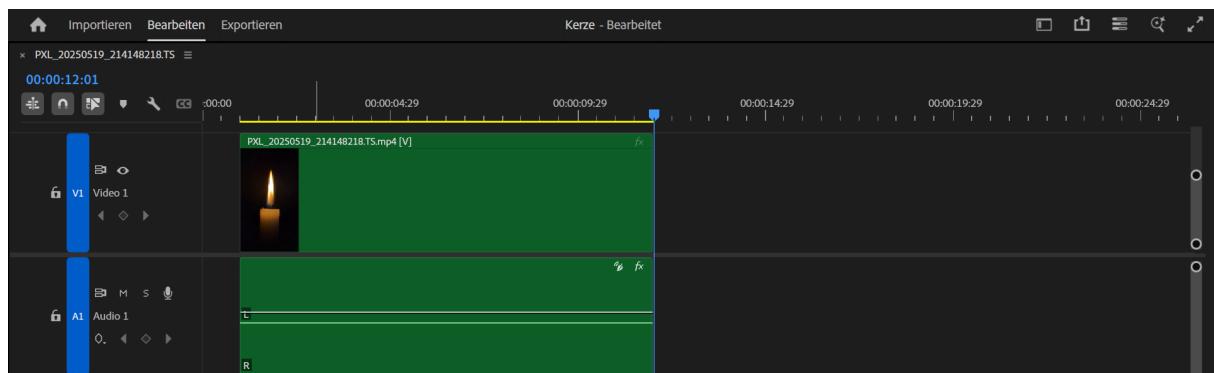


Abbildung 42: Zufälliger Ausschnitt aus dem Ursprungsmaterial (Clip 1)

Das Video wurde an einer zufälligen Stelle abgeschnitten.

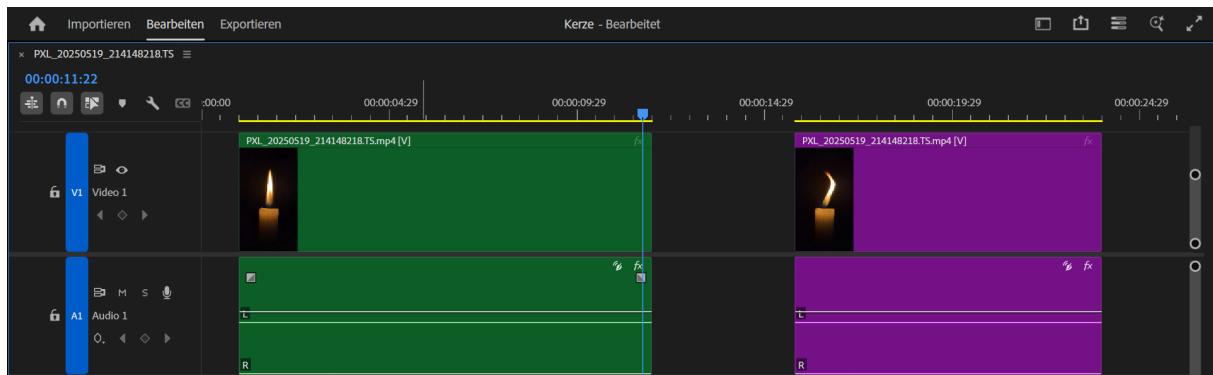


Abbildung 43: Zweiter Zufallsclip (auseinandergeschoben)

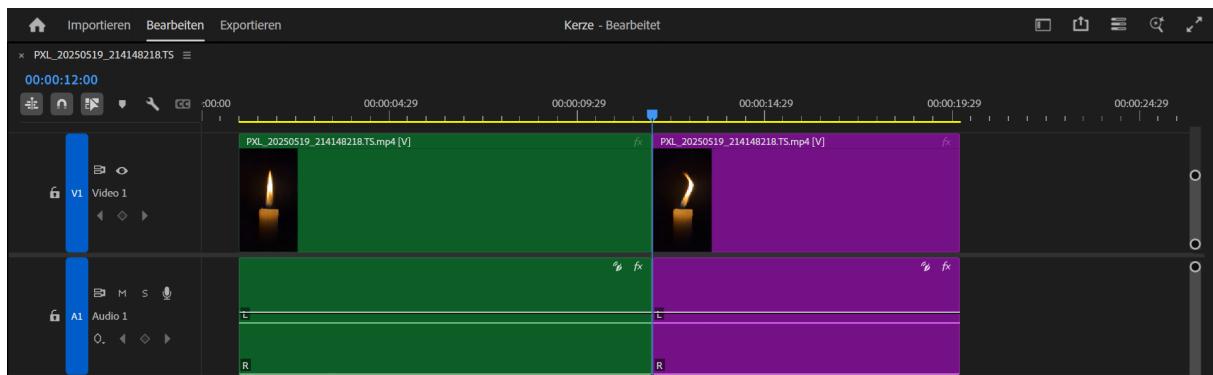


Abbildung 44: Zweiter Zufallsclip (*zusammengezogen*)

Das nochmal kopierte Original wird wieder zu einem Clip geschnitten, aber diesmal an einer anderen zufälligen Stelle. Jetzt gibt es zwei komplett unterschiedliche Clips aus der gleichen Quelle. Wenn nun die Clips nebeneinandergelegt werden, gibt es einen harten Sprung. Nach 12 Sekunden endet der erste Clip. Ein Frame später beginnt der zweite Clip.

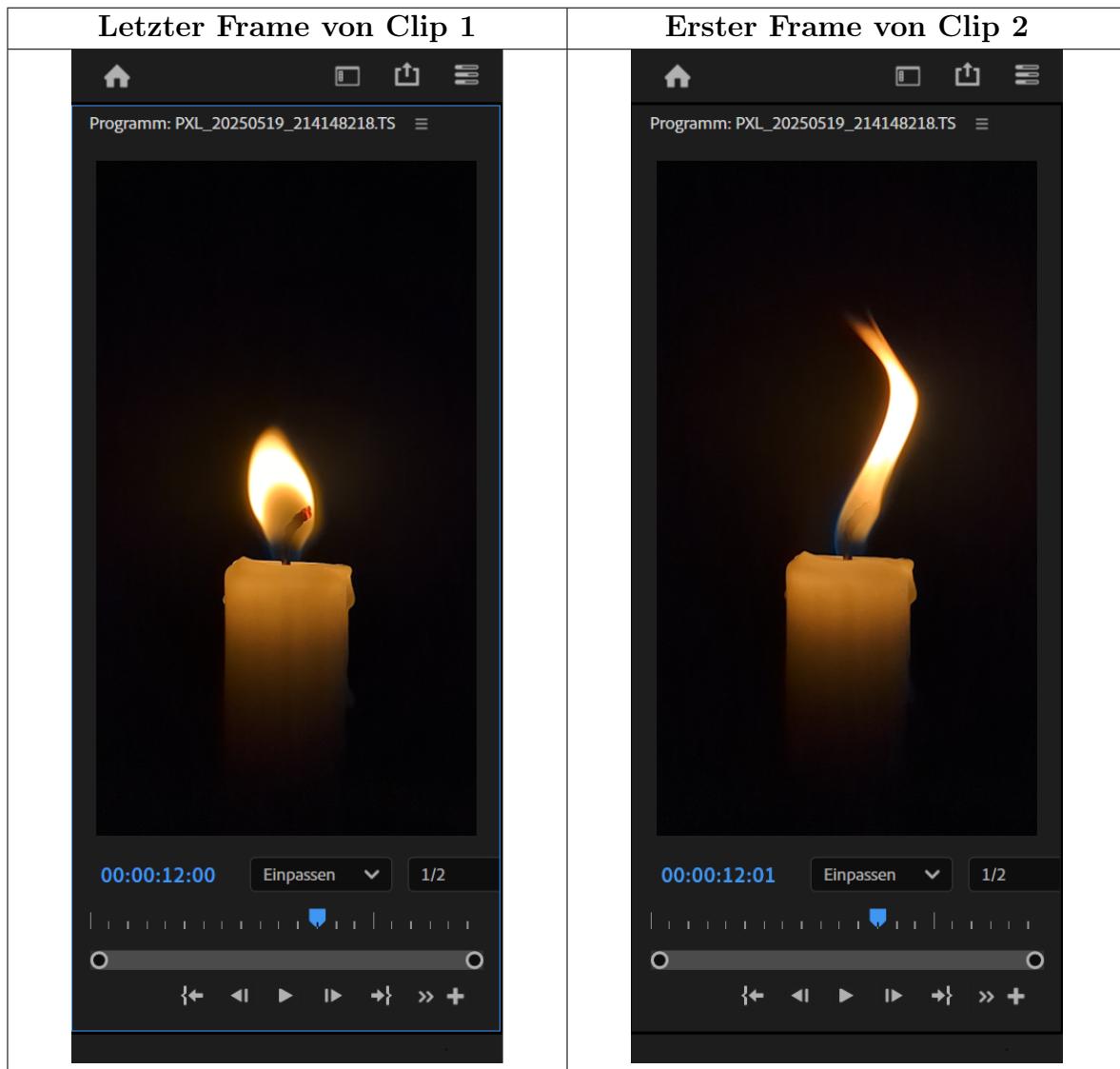


Tabelle 27: Zwei aufeinanderfolgende Frames mit hartem Schnitt zwischen zwei Clips aus demselben Ursprungsmaterial.

Um einen ruckartigen Übergang von Clip 1 in den Clip 2 zu vermeiden, benutzt man eine Blende. So sieht es aus, wenn die Mitte der Blende erreicht ist.

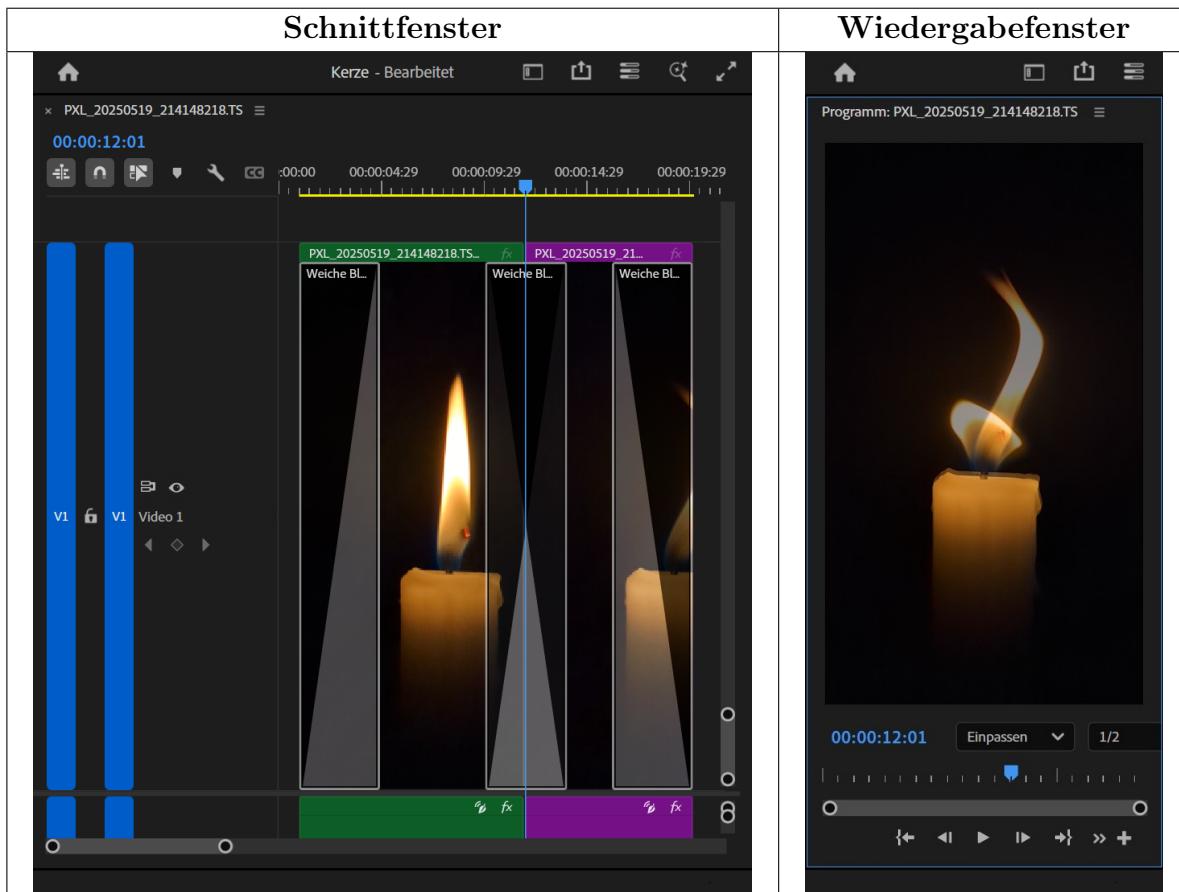


Tabelle 28: Darstellung von Fade-in, Cross-fade und Fade-out im Schnitt- (links) und Wiedergabefenster (rechts).

Die folgende Tabelle veranschaulicht die drei Übergangsarten im Videoschnittprozess.

Fade-in	Cross-Fade	Fade-out

Der Clip wird eingeblendet Überblendung Der Clip wird ausgeblendet

Tabelle 29: Gegenüberstellung der drei Übergangsarten im Videoschnitt: Fade-in, Cross-Fade und Fade-out

Beim genaueren Betrachten wird deutlich, dass im Überblendungsbereich ein Kreuz entsteht, wenn die beiden Rampen im Schnittfenster aufeinandertreffen. Im Prinzip ist das

ein Crossfading für jedes Pixel. Ein Crossfade ist ein fließender Übergang zwischen zwei Clips – ähnlich wie zwei Farben, die ineinander übergehen. Der eine Clip wird dabei allmählich ausgeblendet, während der nächste eingeblendet wird. Zusätzlich gibt es am Anfang einen „Fade-in“ und am Ende einen „Fade-out“. Bei diesem Schnittfenster wird in den grünen Clip aus dem nichts eingeblendet. Dann wird in den violetten Clip und dann ins nichts ausgeblendet. Anstelle des Video Fade-outs könnte man auch weitere Clips von der Ursprungsquelle hinzufügen und einblenden, ad infinitum. Der beschriebene Vorgang ist das Grundprinzip des Algorithmus.

6.2 Crossfading

Das Crossfading ist ein wesentlicher Bestandteil des Algorithmus auf dem Mikrocontroller und der Stromregelung. Deswegen wird in einem eigenen Kapitel erklärt. Ein in der leisen Bibliothek sitzender Mensch empfindet ein plötzlich startendes Düsenflugzeug als harten Sprung. Um es mathematisch zu veranschaulichen, werden für das erste Signal ein Cosinus und für das zweite ein Rechtecksignal verwendet.

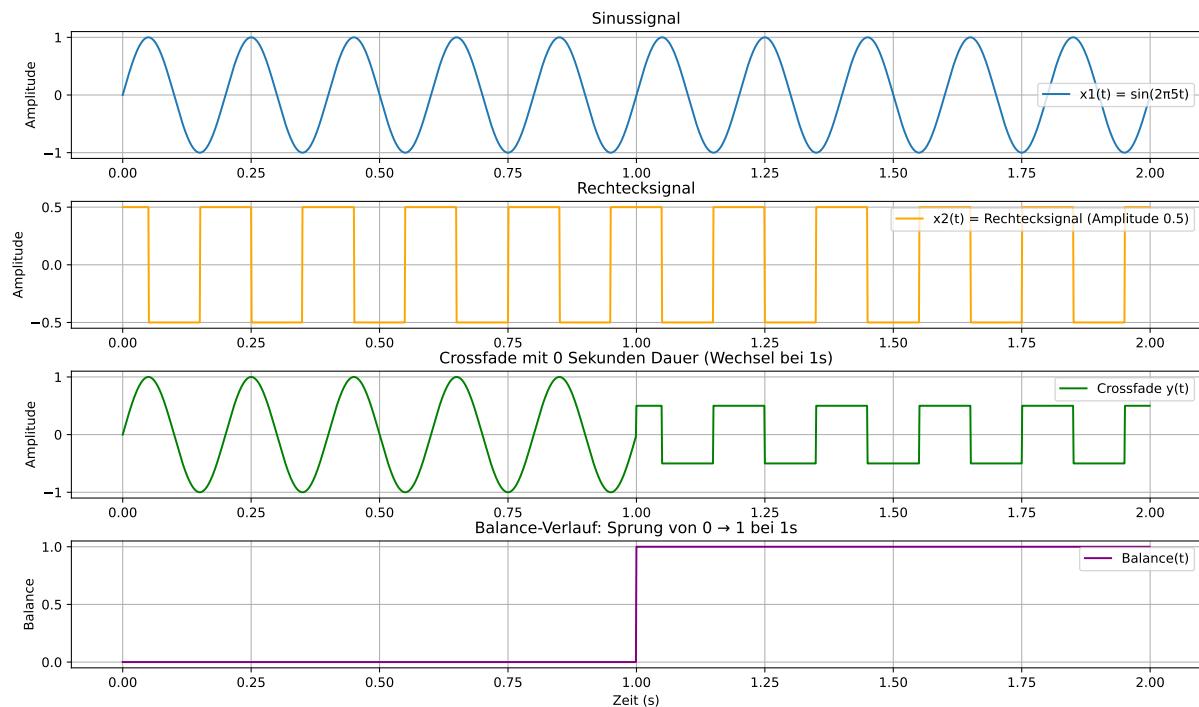


Abbildung 45: Darstellung eines harten Übergangs ohne Crossfade. Das erste Signal endet abrupt, das zweite beginnt ohne Überblendung. Den zugehörigen Beispielcode findet man auf [GitHub](#).

Um diesen Übergang weicher zu gestalten, kommt eine Überblendung (*Crossfade*) zum Einsatz — ein bewährtes Verfahren in der Film- und Audiotechnik. Eine gute Einführung zur Erstellung von Crossfades bietet das offizielle Tutorial von Audacity[11].

Es existieren verschiedene Arten von Crossfades. Eine einfache Form ist der lineare Crossfade, bei dem zwei Audiosignale $x_1[n]$ und $x_2[n]$ anhand eines Balance-Werts überblendet werden. Die Berechnung erfolgt nach folgender Gleichung [12]:

$$y[n] = x_1[n] + \text{Balance} \cdot (x_2[n] - x_1[n]) \quad (51)$$

Hierbei steht:

- $x_1[n]$: erstes Signal
- $x_2[n]$: zweites Signal
- Balance $\in [0, 1]$: Mischfaktor

Für Balance = 0 ergibt sich $y[n] = x_1[n]$, bei Balance = 1 ist $y[n] = x_2[n]$. Wird der Balance-Wert kontinuierlich von 0 auf 1 verändert, entsteht ein weicher Übergang.

In der nachfolgenden Grafik beginnt das Signal x_2 ab $t = 0,5\text{ s}$ und wird linear eingebettet. Bei $t = 1,5\text{ s}$ endet die Überblendung.

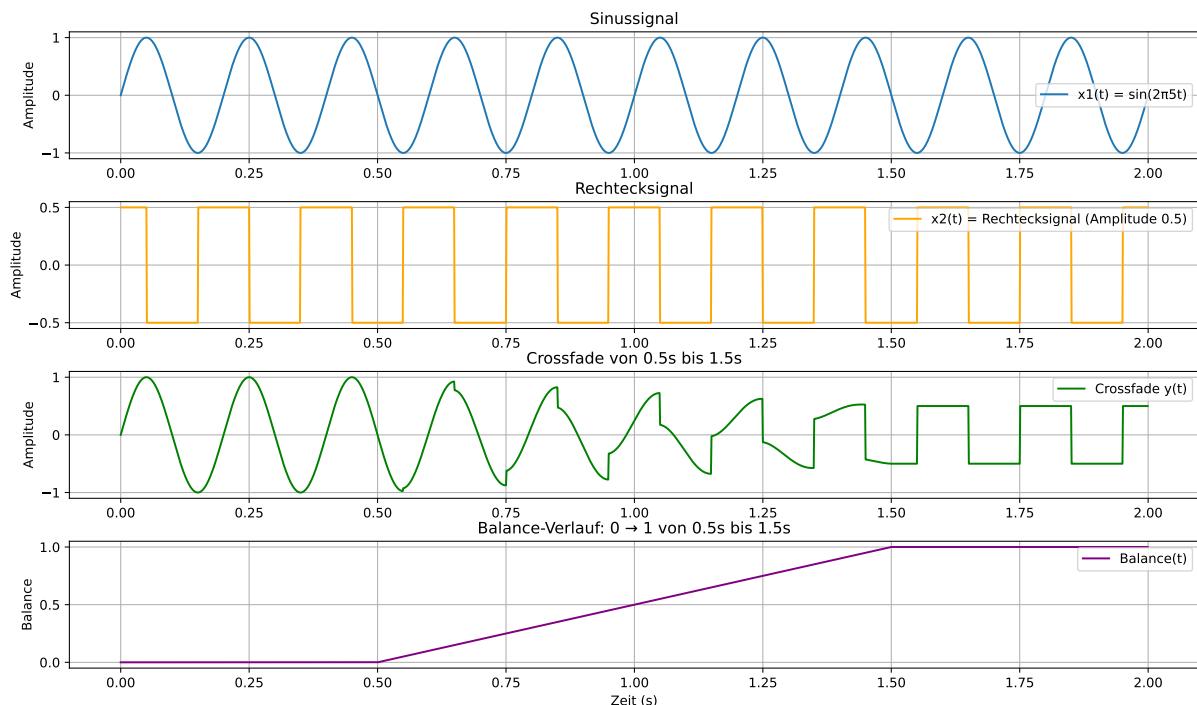


Abbildung 46: Beispiel eines linearen Crossfades zwischen zwei Signalen

Jedoch beginnt und endet der lineare Crossfade abrupt. Um das zu vermeiden, eignet sich das *Cosinus-Crossfading*, das einen weicheren Übergang erzeugt. Die Balance-Funktion lautet:

$$\text{Balance}[n] = \frac{1 - \cos(\pi \cdot \frac{n}{N})}{2} \quad (52)$$

Dabei ist:

- n : aktueller Sample-Index im Übergangsbereich ($0 \leq n \leq N$)
- N : Länge der Überblendung in Samples

Die Balance-Werte verhalten sich wie folgt:

- $n = 0$: Balance = 0 \Rightarrow nur $x_1[n]$ aktiv
- $n = \frac{N}{2}$: Balance = 0,5 \Rightarrow beide Signale gleich stark
- $n = N$: Balance = 1 \Rightarrow nur $x_2[n]$ aktiv

Durch die flachen Flanken der Cosinusfunktion verläuft der Übergang deutlich harmonischer. Diese Methode eignet sich besonders für Echtzeit-Anwendungen auf Mikrocontrollern, da die Berechnung effizient ist und kaum Rechenzeit beansprucht.

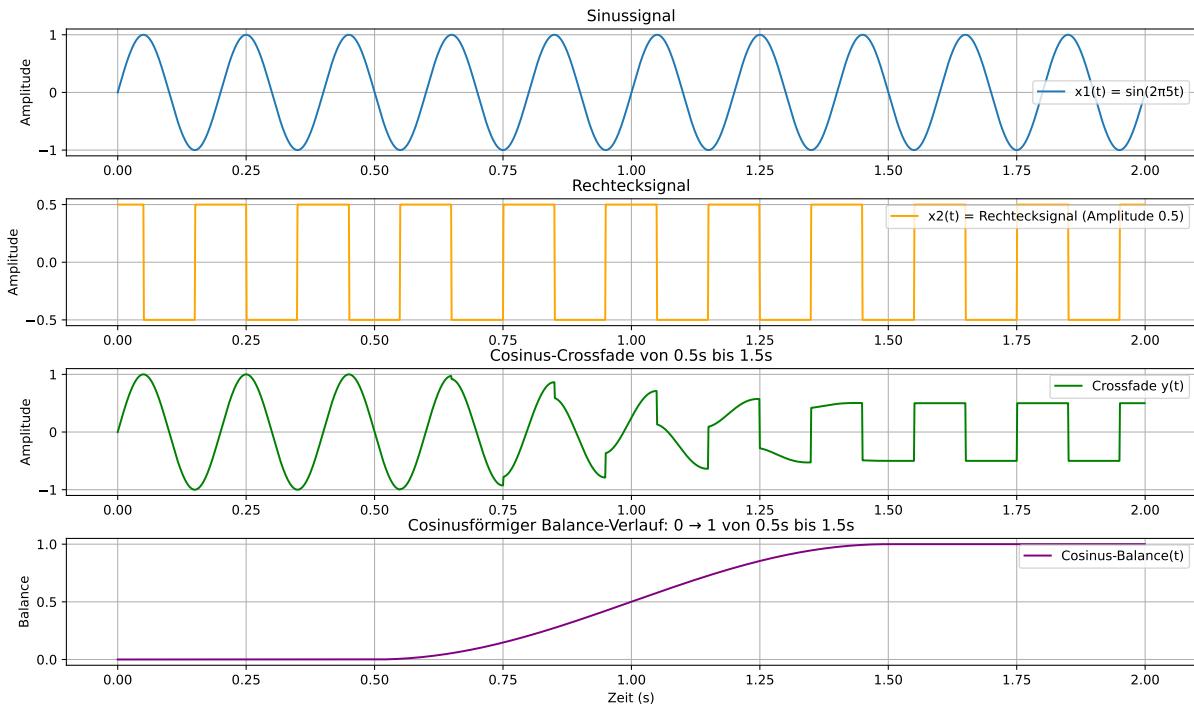


Abbildung 47: Weicher Übergang mit Cosinus-Crossfade

Diese Methode eignet sich besonders für Übergänge in der Audio- und Musikproduktion, bei denen ein natürlicher Verlauf ohne wahrnehmbare Lautstärkeschwankungen gewünscht ist.

6.3 Simulation

Für die Visualisierung und Simulation des Algorithmus wurde ein Python-Programm geschrieben, das die Arbeitsweise aus signaltechnischer Sicht erklärt. Die drei Plots in Abb. 48 illustrieren nacheinander

1. einen einzelnen Videopixel (graue Rampe) mit 25 fps, 8 bit Auflösung und 265 Frames (≈ 11 s);
2. die zufällig gewählten *in-* und *out-Points* nebst Crossfade-Intervall (gelb) sowie den bereits vorbereiteten nächsten Clip;
3. den zeitlich unendlichen Verlauf des resultierenden Kerzen-„Films“ (rote Markierung *play*).

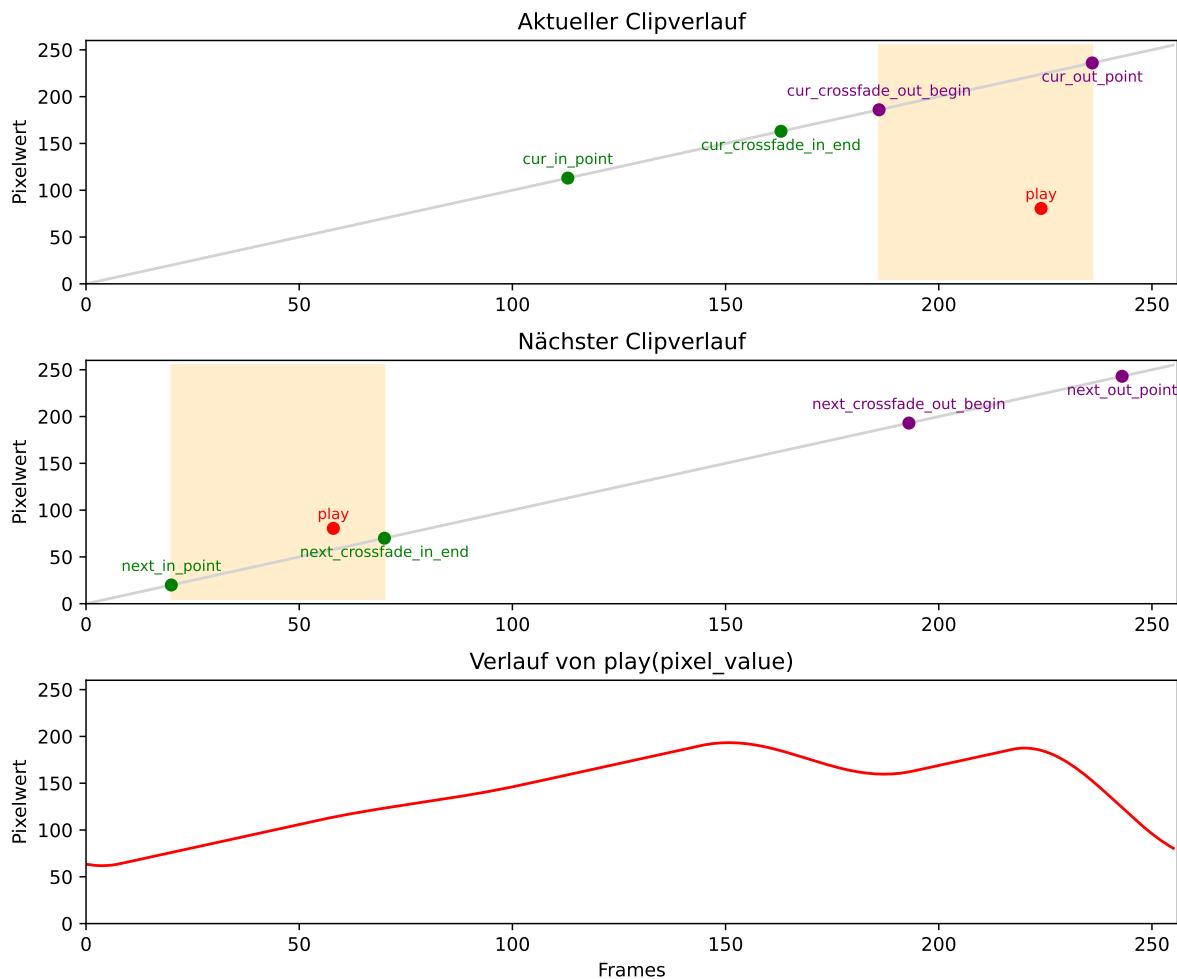


Abbildung 48: Schrittweise Entstehung des unendlichen Lichtsignals: **oben** Originalrampe mit markierten Schnittpunkten, **mitte** Vorbereitung des nächsten Clips, **unten** kontinuierlicher Ausgangsverlauf. Den zugehörigen Code findet man auf [GitHub](#).

Im Folgenden wird der Ablauf noch einmal stichpunktartig zusammengefasst:

1. **Zufälliges Schneiden:** Auswahl von *in- und out-Point* im Ursprungsvideo (grüne bzw. violette Punkte in Plot 1).
2. **Definition des Crossfade-Intervalls:** Länge `crossfade_length = 50` Frames \Rightarrow gelber Bereich zwischen `crossfade_in_end` und `crossfade_out_begin`.
3. **Abspielen des ersten Clips:** Die Markierung *play* wandert frameweise bis zum Crossfade-Start; die Pixelwerte werden in den dritten Plot geschrieben.
4. **Crossfading:** Während des gelben Intervalls mischt die Crossfade-Funktion den ausblendenden Clip 1 mit dem einblendenden Clip 2. \Rightarrow kontinuierlicher Stromwert pro Kanal.
5. **Schiebefenster:** Nach jedem vollständigen Crossfade wird Clip 1 verworfen, Clip 2 wird zu Clip 1, und ein neuer zufälliger Clip 2 wird vorbereitet. Die Darstellung in Plot 3 behält stets die Länge des Ursprungsvideos, indem am linken Rand alte Werte herausgeschoben werden.

Als vorletzten Schritt führt das Programm ein *Upsampling* durch: Die ursprünglichen 25 fps (8 bit) werden per linearer Interpolation auf 1000 fps bei $\text{PWM}_{\text{carrier}} = 16 \text{ kHz}$ angehoben. Ein nachgeschalteter IIR-Filter beseitigt Diskretisierungsrelikte, so dass jeder der vier PWM-Kanäle schließlich ≈ 1000 Helligkeitsstufen bei 1000 fps erreicht.

6.3.1 Upsampling und Auflösungsvergrößerung

Als vorletzter Verarbeitungsschritt erfolgt das Upsampling. Das System ist nun in der Lage, kontinuierliche „Kerzenfilme“ in Echtzeit Bild für Bild zu erzeugen. Im vorherigen Schritt wurde bewusst mit einer reduzierten zeitlichen und räumlichen Auflösung gearbeitet, um den Rechenaufwand erheblich zu reduzieren.

Im nächsten Schritt werden die Helligkeitswerte der Videopixel in Steuergrößen für die PWM-Ausgabe umgewandelt. Die gespeicherten Videos besitzen typischerweise eine Auflösung von 8 Bit pro Kanal und eine Bildfrequenz zwischen 25 und 30 Bildern pro Sekunde, abhängig vom Dateiformat. Diese Daten werden anschließend hochskaliert, um von einem diskreten Verlauf in einen kontinuierlichen Verlauf zu gelangen. Die folgende Grafik zeigt die Verarbeitungsschritte.

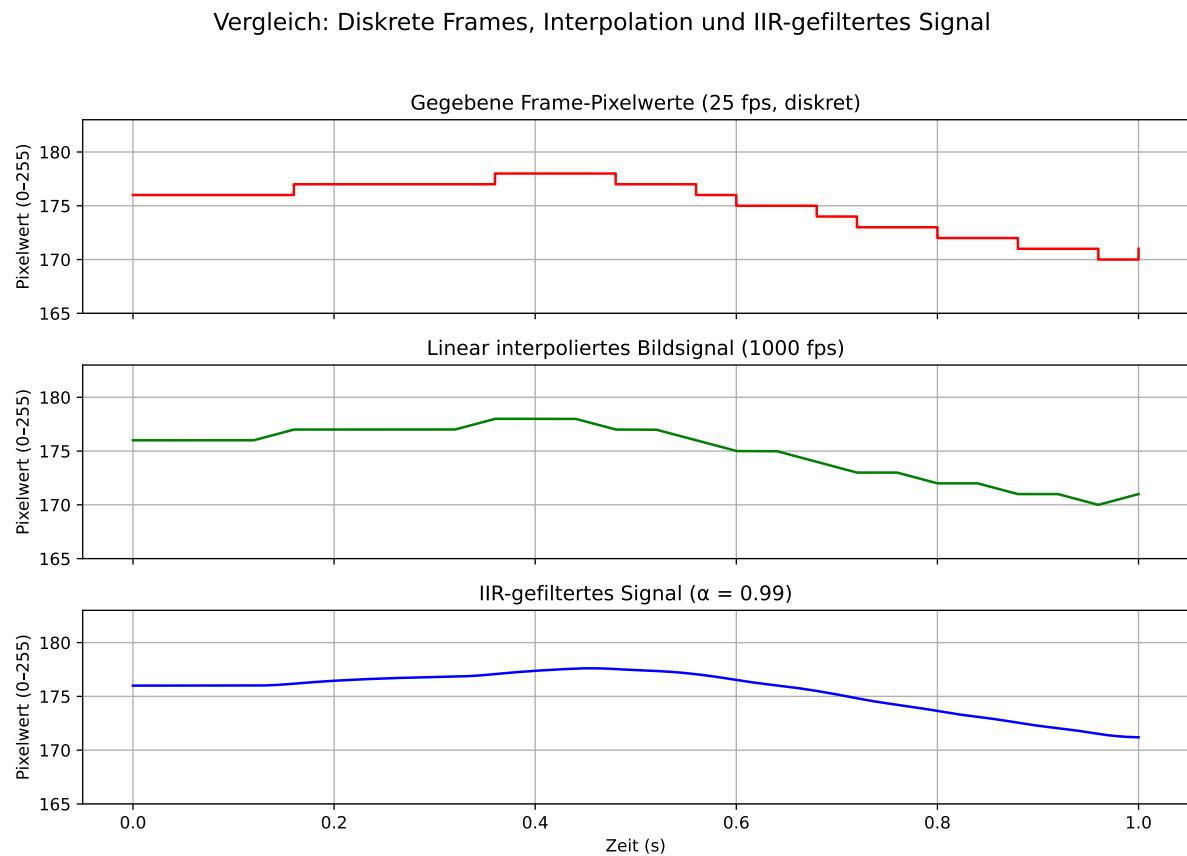


Abbildung 49: Verarbeitungskette vom diskreten Videosignal zum hochauflösten PWM-Stromverlauf.

Der verwendete Timer (TIM2) arbeitet mit einer PWM-Trägerfrequenz von 16 kHz und aktualisiert seinen Duty-Cycle jede ms. Die Erhöhung der Auflösung wird erreicht, indem durch lineare Interpolation die zeitliche Auflösung auf 1000 fps erhöht wird, um ein visuelles zeitdiskretes Signal in ein scheinbar zeitkontinuierliches Signal zu verwandeln.

Der obige Plot wurde mit echten Kerzendaten aus dem *Light Source Extractor* erzeugt. Im letzten Schritt des Algorithmus werden die Relikte der Diskretisierung entfernt durch einen IIR-Filter.

Der Verlauf ist eine direkte Repräsentation des Stroms, welcher durch die LEDs fließt. Im Programm wird für jeden Kanal ein Nennstrom eingetragen. Es wird festgelegt, dass bei 100 % Nennstrom die Helligkeit der LED bei 100 % ist.

6.3.2 Zufall

Die folgende Grafik zeigt, wie der Algorithmus mit einem Kerzenvideo mit 8-Bit-Pixelauflösung, 750 Frames, 25 FPS und Kanal 1 funktioniert.

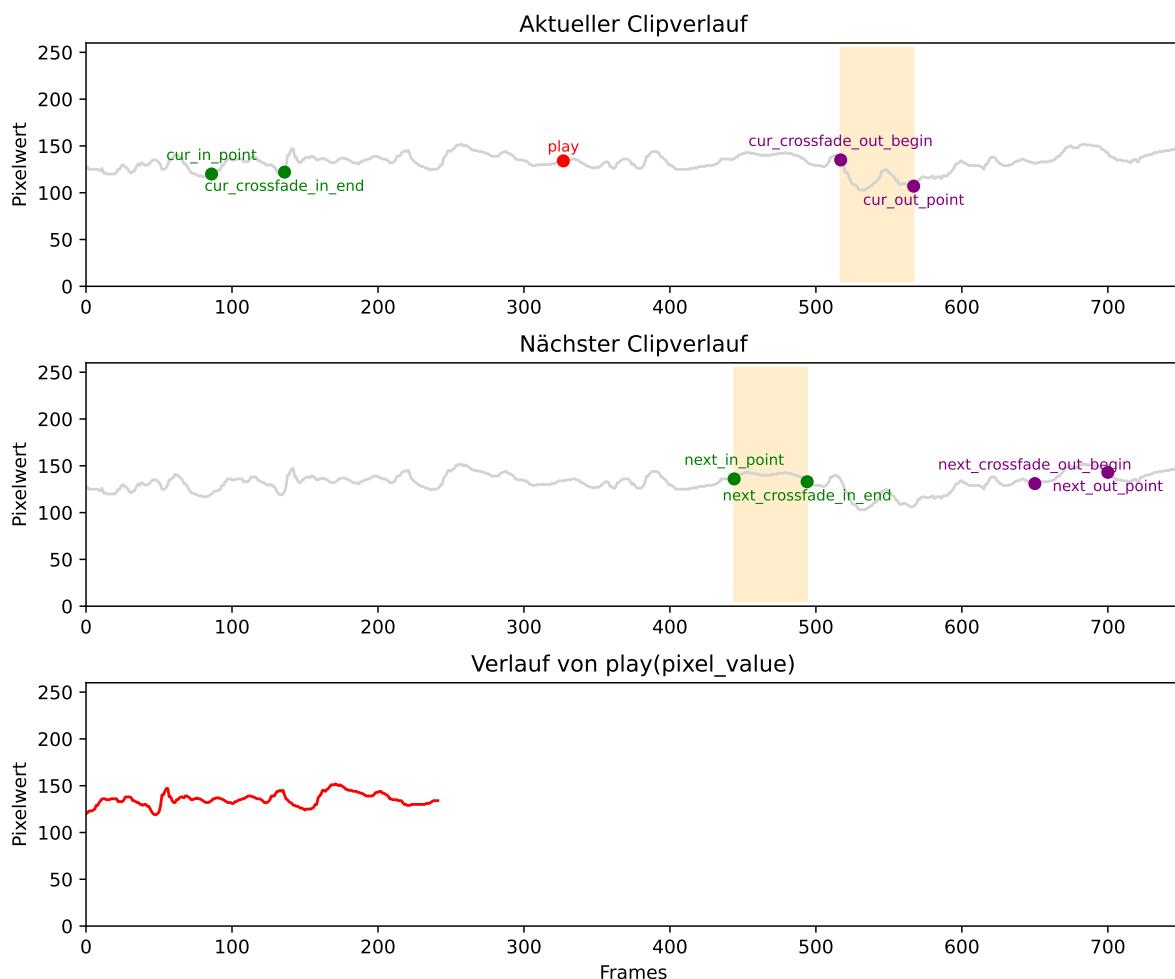


Abbildung 50: Signalverlauf eines einzelnen Kanals (8 bit, 750 Frames, 25 FPS) nach mehrfacher Clip-Montage ohne Zufallsparameter. Den zugehörigen Code findet man auf [GitHub](#).

Man erkennt sequenziell, dass das Ausgangssignal oft die gleichen Komponenten des Ursprungsmaterials wiederholt. Um den Effekt der Periodizität zu reduzieren, kann man ein längeres Quellvideo nehmen, Clips mit zufälligen Schnittstellen und Längen aussuchen, mit Rauschen oder Helligkeitsunterschieden oder Geschwindigkeitsunterschieden. Bei Bedarf kann man weitere Parameter manipulieren.

Echte Kerzen verhalten sich unvorhersehbar: Wind, Helligkeit, Luftbewegungen, Temperatur oder auch das Brennmuster verändern ständig die Brenndauer. Deshalb kann man diese auch nur schätzen. Um dieses natürliche Verhalten möglichst realistisch zu simulieren, wurden in den Algorithmus gezielt Zufallsparameter eingebaut.

Es gibt diverse Zufallsalgorithmen. Für dieses Projekt wird der Zufall durch den *WELL512*-Algorithmus erzeugt. Es handelt sich um einen Pseudorandom-Number-Generator (PRNG), d.h. einen Algorithmus, der durch eine Folge mathematischer Operationen eine große Menge scheinbar zufälliger Zahlen produziert. Mit einer Periode von $2^{512} - 1$ wiederholt sich seine Sequenz praktisch nie. Außerdem hat dieser Algorithmus eine bessere Zufallsverteilung als der übliche LCG-PRNG. PRNGs sind immer deterministisch: Wird er mit demselben Startzustand initialisiert, liefert er stets dieselbe Zahlenfolge.

Der WELL512-Algorithmus benötigt nur 64 B internen Speicher und wenige Taktzyklen pro Zufallszahl, sodass er selbst auf einem STM32 in Echtzeit läuft. Damit jede Kerze ein individuelles Lichtmuster zeigt, wird der Generator bei jedem Einschalten mit aktuellen Messwerten der Schaltung gestartet. Dazu liest der Mikrocontroller mehrmals hintereinander die Batteriespannung und die Temperatur über den ADC ein, kombiniert diese und nutzt sie zur Initialisierung, um die Entropie zu verbessern.

Die Kombination aus physikalisch verrauschten Messwerten und einem starken Zufallsgenerator wie dem WELL512 macht jede Kerze einzigartig und quasi nicht deterministisch. Der Zufall wird benutzt, um für Clips immer zufällige *in-Points* und *out-Points* zu erzeugen, die sich maximal voneinander unterscheiden. Außerdem kann man den Zufall benutzen, um bei jedem Start der Kerze einen individuellen kleinen Helligkeitsoffset zu geben oder eine leicht andere Abspielgeschwindigkeit.

7 Ergebnis

Die in der Einleitung eingeführten Herausforderungen der Herstellung eines für die hohen Anforderungen der Filmindustrie geeigneten elektrischen Kerzenlichts erforderten die Entwicklung eines innovativen Produkts mit einer Kombination aus neuartigen und bewährten Hard- und Softwarelösungen. Im Laufe der Arbeit wurden die Herausforderungen aufgezeigt und die technischen Lösungen in ihrer Theorie und Praxis erläutert.

Jede digitale Kerze ist durch den Algorithmus, der lineare Interpolation, Crossfading und den Zufall verwendet, ein Unikat und dadurch für realistische Szenen in der Filmproduktion geeignet. Durch den Light Source Extractor lassen sich auf jeder Kerze andere Lichtmuster erzeugen, z. B. Wunderkerzen, Teekerzen, usw. und diverse Lichtsituation nachahmen, etwa das Flackern bei starkem Wind, das ruhige Leuchten bei Windstille oder Verlöschen.

Die Flamme der digitalen Kerze bewegt sich, wie bei einer echten Kerze, im 3D-Raum. Dieser Effekt wird realisiert durch vier im Kerzenkopf angeordnete LED-Filamente, die in jede Richtung des Raumes zu jeder Zeit ein individuelles Lichtbild erzeugen können. Hierin liegt die größte Innovation meiner multifunktionalen Lichtquelle. Die vier LED-Filamente sind so angeordnet, dass sich ihre Lichtemissionen räumlich überlagern und ein 3D-Leuchtbild erzeugen. Das heißt die hier entwickelte LED-Kerze erzeugt auch bewegten Schatten.

Durch die angemessen hohe PWM-Frequenz und Signalglättung kann man selbst bei professionellen Filmkameras kein Flackern mehr erkennen. Außerdem ist die Auflösung von 1000 Helligkeitsstufen bei 1000 fps je Kanal sehr hoch. Der Kopf der Kerze hat durch die Filamente die vielfache Lichtstärke einer echten Kerze guten Farbwiedergabeindex. Durch

die Anordnung der vier Filamente in Segmente lassen sich die natürlichen Bewegungen einer Kerzenflamme simulieren. Durch das kompakte Design der Platine kann diese in diverse Gehäuseformen verbaut werden.

8 Ausblick

Eine Vielzahl an Weiterentwicklungen und möglicher Anwendungen sind denkbar. Das nächste Ziel ist die und Erreichung der CE-Konformität. Außerdem soll in der nächsten Generation ein Funkstandard wie Bluetooth oder WiFi integriert werden, um eine Ansteuerung über App und effiziente Gruppenschaltungen zu ermöglichen. Dafür soll auch die RED-Prüfung zukünftig bestanden werden. Ein Kompasssensor soll eingebaut werden, damit eine Synchronisation zwischen den Kerzen möglich ist, z. B. ein gleichgerichtetes Verhalten bei einem Windstoß. Weiter soll die Elektronik um eine Speicherreserve ergänzt werden. Ein externer SPI-Electrically Erasable Programmable Read-Only Memory (EEPROM) (4 MB) soll Platz für längere Lichtsequenzen und einfachere Updates schaffen. Parallel soll geprüft werden, ob ein kleines RNN auf einem leistungsfähigeren Mikrocontroller in Echtzeit realistische Helligkeitsverläufe liefert. Weiter soll ein Kerzenmodell entwickelt werden, welches ein Hochfrequenz-Verneblermodul betreibt, um das Licht durch Nebelaustritt durch dünne Daüsen zu leiten. Dadurch wird die Flamme nicht mehr durch Filamente realisiert, sondern im Kerzenschacht mit dem Nebel erzeugt und dann über den Docht nach außen geleitet. Dabei dient der Nebel als Lichttransportträger und 3D-Flamme, die sich abhängig von Wind und Trägheit bewegt. Die entwickelte Hardware die dadurch gewonnenen Erkenntnisse bilden die Grundlage, um weitere Anwendungen zu ermöglichen.

Abbildungsverzeichnis

1	Prototyp der digitalen Kerze	5
2	LED-Filamente	6
3	Kerzenkopf mit LED-Filamenten, vergossen in Epoxidharz	6
4	SMD-LED der Bauform 0402 vom Typ Everlight 16/916/T1D/AP1Q2QY-/3T	7
5	LED-Filament ohne Phosphorschicht	8
6	Aufgeschnittenes LED-Filament	8
7	Vorder- und Rückseite der Kopfplatine mit SMD-LEDs (3D-Modell)	12
8	Overlay beider Seiten (relevanter Ausschnitt)	12
9	KI-generiertes Stimmungsbild	12
10	Spektrenvergleich und CRI der Wachskeze, Filament und SMD-LED. Den zugehörigen Code findet man auf GitHub.	14
11	Ansteuerung des LED-Filaments über Kanal 1: Verbindung zwischen FPC-Steckverbinder HEADER und der Steuerplatine.	16
12	Finaler Kerzenkopf vor dem Verguss mit Epoxidharz und Flachbandkabel .	17
13	Finaler Kerzenkopf nach dem Verguss mit Epoxidharz und Flachbandkabel	17
14	FFC-Kabel (HC-FFC05-10P-100-A-01)	18
15	Stromfluss von 10 A durch durch alle Kanäle parallel geschaltet	19
16	6A fließt durch 2 Adern (einen Kanal). Das Kabel raucht.	20
17	Mit der Schaltung kompatible Batterien der Typen 14650 und 14430	21

18	Powermanagement der Steuerplatine. Links der Batterieladeregler, rechts der Linearregler	23
19	Triptychon der Steuerplatine. Die farblich markierten Bereichen werden in der folgende Tabelle 15 erläutert.	24
20	Abbildung des eingesetzten MOSFETs [7]	26
21	Schaltplan-Ausschnitt der PWM-Ansteuerung	27
22	Plots zum visuellen Effekt des Tiefpassfilters am Gate. Tastrate liegt bei 5 %. Den zugehörigen Code findet man auf GitHub	28
23	Zur besseren Veranschaulichung wird ein weiterer Plot als Helligkeitsverlauf visualisiert.	29
24	Der rote Pfeil zeigt auf die internen Bauteile einer LED-Birne (Kondensatoren, Trafo, Spule)	30
25	Foto des Kerzenkopfes aufgenommen mit einer Canon 750D bei einer Verschlusszeit von 1/4000 und ISO 12800. Tastrate für alle Kanäle bei 50 %.	34
26	Übersicht der realisierten Strommessschaltung	36
27	Schaltplan der Strommessung	36
28	Konzeptioneller Verlauf von PWM, Sprungsignal und Tiefpass-Ausgang. Den zugehörigen Code findet man auf GitHub	39
29	Visualisierung der Messintervalle eines gefilterten PWM-Signals	40
30	Darstellung des digitalen Mittelwertverfahrens mit Oversampling	41
31	Verbesserte Glättung des PWM-Signals durch zweite Filterstufe	43
32	Nachweis, dass die Slew-Rate in beiden Filtervarianten nicht verletzt wird	44
33	Benutzeroberfläche des Light Source Extractors beim Start	45
34	Zusammenhang zwischen linearer und wahrgenommener Helligkeit	46
35	Frame mit grünem Ausschnitt (relevante Lichtinformation) und blauer Tracking-Bahn	48
36	Bestimmung der ROI (roter Rahmen) aus allen grünen Ausschnitten	49
37	Frameanalyse einer Tafelkerze (Pipeline 2)	50
38	Segmentierung in Lichtregionen	50
39	Rotationskorrektur mittels Durchschnittsvektor	51
40	Visualisierung der normierten Helligkeitsverläufe der vier Bildsegmente über die Zeit. Die Daten dienen als Grundlage für die spätere PWM-Signalrekonstruktion.	52
41	Schnittfenster in Adobe Premiere Pro mit importiertem Videomaterial einer Kerze. Das zugehörige Beispielprojekt findet man auf GitHub	54
42	Zufälliger Ausschnitt aus dem Ursprungsmaterial (Clip 1)	54
43	Zweiter Zufallsclip (auseinandergeschoben)	55
44	Zweiter Zufallsclip (<i>zusammengezogen</i>)	55
45	Darstellung eines harten Übergangs ohne Crossfade. Das erste Signal endet abrupt, das zweite beginnt ohne Überblendung. Den zugehörigen Beispielcode findet man auf GitHub	58
46	Beispiel eines linearen Crossfades zwischen zwei Signalen	59
47	Weicher Übergang mit Cosinus-Crossfade	60
48	Schrittweise Entstehung des unendlichen Lichtsignals: oben Originallampe mit markierten Schnittpunkten, mitte Vorbereitung des nächsten Clips, unten kontinuierlicher Ausgangsverlauf. Den zugehörigen Code findet man auf GitHub	61
49	Verarbeitungskette vom diskreten Videosignal zum hochauflösten PWM-Stromverlauf.	62

50	Signalverlauf eines einzelnen Kanals (8 bit, 750 Frames, 25 FPS) nach mehrfacher Clip-Montage ohne Zufallsparameter. Den zugehörigen Code findet man auf GitHub.	63
----	--	----

Tabellenverzeichnis

1	Wie kann sich die Flamme der Kerze bewegen?	9
2	Jedes Bild aus der Tabelle wird nun in 4 gleiche Segmente aufgeteilt.	9
3	Nun wird jedes Segment durch ein kurzes LED-Filament repräsentiert.	9
4	Filament-Anordnungen und 3D-Eindruck	9
5	Lange Filamente – Varianten	10
6	Kurze Filamente übereinander – Varianten	10
7	Kombination lange / kurze Filamente	10
8	Platzbedarf im Kerzenkopf	11
9	Platine des Kerzenkopfs – Untere und obere Lage	15
10	Nutzung des FFC-Kabels	18
11	Vergleich geeigneter Li-Ionen-Zellen	21
12	Lastfälle im Sollbetrieb	21
13	Lastfälle im Maximalbetrieb	22
14	Lastfälle im Grenzbetrieb	22
15	Vergleich mit echten Kerzen	22
16	Signifikante Funktionseinheiten der Steuerplatine	24
17	Vergleich der Betriebsarten bezüglich des Wirkungsgrads	33
18	Parameter zur Auslegung des Tiefpassfilters	33
19	Bewertung möglicher Signalpfade für die Strommessung	35
20	Erläuterung der Leitungen in der Strommessschaltung	36
21	Auswirkungen unterschiedlicher Verstärkungsfaktoren auf die Strommessung	37
22	Berechnungsgrundlage zur Komponentenwahl und Toleranzbetrachtung . .	44
23	Gegenüberstellung von Gamma-Bereich und linearer Bereich	47
24	Position des Trackingpunkts in den beiden Pipelines	49
25	Strukturierte Zerlegung des ROI-Bereichs in vier gleich große Bildsegmente (Quadranten). Links: Original-Frame mit markiertem ROI; rechts: 2×2 -Raster zur Identifikation der ruhigsten Lichtregion.	51
26	Struktur des C-Exports der extrahierten Lichtkanäle mit farblicher Hervorhebung	53
27	Zwei aufeinanderfolgende Frames mit hartem Schnitt zwischen zwei Clips aus demselben Ursprungsmaterial.	56
28	Darstellung von Fade-in, Cross-fade und Fade-out im Schnitt- (links) und Wiedergabefenster (rechts).	57
29	Gegenüberstellung der drei Übergangsarten im Videoschnitt: Fade-in, Cross-Fade und Fade-out	57

Quellenverzeichnis

- [1] LCSC Electronics. *Everlight 16-916/T1D-AP1Q2QY/3T - 0402 SMD LED*. Zugriff am 17.06.2025. 2024. URL: https://lcsc.com/product-detail/image/16-916-T1D-AP1Q2QY-3T_C264412.html.

- [2] Pur-LED GmbH. *Filament COB LED (24 V DC) – Technisches Datenblatt*. Techn. Ber. Lichtausbeute: 83 lm/W. Pur-LED GmbH, 2025. URL: https://www.pur-led.de/out/media/str_703022_24vdc_cob_filament_datenblatt_00827.pdf.
- [3] Bilder-Plus.de. *Lichtstärke, Lichtstrom und Abstrahlwinkel verständlich erklärt*. Abgerufen am 19.06.2025. 2025. URL: <https://www.bilder-plus.de/lichtstaerke.php>.
- [4] RechnerOnline. *Raumwinkel berechnen*. Abgerufen am 19.06.2025. 2025. URL: <https://rechneronline.de/winkel/raumwinkel.php>.
- [5] LCSC Electronics. *HC-FFC05-10P-100-A-01 Flexible Flat Cable (FFC)*. Zugriff am 17.06.2025. 2024. URL: https://lcsc.com/product-detail/FFC-Cable-Flexible-Flat-Cable_HCTL-HC-FFC05-10P-100-A-01_C6575021.html.
- [6] JST Mfg. Co., Ltd. *FH Connector Series (0.5 mm Pitch, Non-ZIF FPC Connector) – Datasheet*. Techn. Ber. Current rating: 0.5 A per contact; Voltage rating: 50 V AC/DC. JST Mfg. Co., Ltd., 2021. URL: <https://www.jst-mfg.com/product/pdf/eng/eFH.pdf>.
- [7] LCSC Electronics. *IRLML6244TRPBF - N-Channel MOSFET 30V 5.2A SOT-23*. Zugriff am 17.06.2025. 2024. URL: https://lcsc.com/product-detail/image/IRLML6244TRPBF_C143946.html.
- [8] Texas Instruments. *INA181 Precision, Bidirectional Current Sense Amplifier*. Datenblatt, Zugriff am 17.06.2025. 2023. URL: <https://www.ti.com/lit/ds/symlink/ina181.pdf>.
- [9] James M. Fiore. *Slew Rate and Power Bandwidth*. Zugriff am 17.06.2025. 2020. URL: [https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Electronics/Operational_Amplifiers_and_Linear_Integrated_Circuits_-_Theory_and_Application_\(Fiore\)/05%3A_Practical_Limitations_of_Op_Amp_Circuits/5.04%3A_Slew_Rate_and_Power_Bandwidth](https://eng.libretexts.org/Bookshelves/Electrical_Engineering/Electronics/Operational_Amplifiers_and_Linear_Integrated_Circuits_-_Theory_and_Application_(Fiore)/05%3A_Practical_Limitations_of_Op_Amp_Circuits/5.04%3A_Slew_Rate_and_Power_Bandwidth).
- [10] Elliot Williams. *RGB LEDs: How To Master Gamma And Hue For Perfect Brightness*. Hackaday. 23. Aug. 2016. URL: <https://hackaday.com/2016/08/23/rbg-leds-how-to-master-gamma-and-hue-for-perfect-brightness/> (besucht am 16.06.2025).
- [11] Audacity Support. *Making Crossfades*. Zugriff am 15.05.2025. 2025. URL: <https://support.audacityteam.org/audio-editing/mixing-and-rendering/making-crossfades>.
- [12] Christopher Dobrian. *Mix two signals more efficiently*. Zugriff am 17.06.2025. 2011. URL: <https://music.arts.uci.edu/dobrian/maxcookbook/mix-two-signals-more-efficiently>.