

# «DJ» EMI Filter für Schaltnetzteil

Fachbericht

Windisch, 20.04.2019



**Hochschule** Hochschule für Technik - FHNW

**Studiengang** Elektro- und Informationstechnik

**Auftraggeber** Dr. Luca Dalessandro

**Betreuer** Prof. Dr. Sebastian Gaulocher  
Prof. Peter Niklaus  
Prof. Dr. Richard Gut  
Dr. Anita Gertiser  
Pascal Buchschacher

**Autoren** **Gruppe 1**  
Niklaus Schwegler  
Lukas von Däniken  
Pascal Puschmann  
Simon Rohrer  
Marco Binder

**Version** 2.0

Abstract  
und im hauptfile

**Zusammenfassung**

## **Inhaltsverzeichnis**

# 1 Einleitung

Entwurf Einleitung P2 EMI-Filter Team 1

Gemäss des Lastenhefts wurde der Auftrag erteilt, eine Simulationssoftware zu entwickeln. Diese Simulationssoftware soll die Einfügedämpfung eines EMI-Filters simulieren und grafisch darstellen. EMI-Filter werden üblicherweise in Schaltnetzteile verbaut, um zu verhindern, dass Störungen zurück ins Netz gespeist werden. Netzgeräte können unter Umständen hohe Frequenzen erzeugen, die sich nicht gut mit der Netzfrequenz von 50 Hz vertragen. Der EMI-Filter filtert genau diese hochfrequenten Signale heraus, um zu verhindern, dass andere Geräte, die auch ans Netz angeschlossen werden, nicht davon beeinträchtigt werden.

Um die Einfügedämpfung zu ermitteln, soll das EMI-Filter bezüglich der Gleich- und Gegentaktschaltung untersucht werden. Dies geschieht anhand zweier Funktionen für Gleich- und Gegentaktschaltung. Die Funktionen zeigen die Einfügedämpfung für einen Frequenzbereich bis 30MHz. Die beiden Schaltungen beinhalten die parasitären Parameter der elektrischen Komponenten, sodass eine möglichst wahrheitsgetreue Simulation gemacht werden kann. Die entwickelte Software soll die Einfügedämpfung grafisch darstellen. Die Resultate werden für die Gleich- und Gegentaktschaltung in separaten Funktionen dargestellt. Des Weiteren sollen die elektrischen Komponenten der Schaltungen in der Simulationssoftware variiert werden können.

Damit sichergestellt werden kann, dass die Simulationen mathematisch korrekt sind, werden alle Berechnungen zuerst in MATLAB durchgerechnet. Diese Ergebnisse werden mit Simulationen der Simulationssoftware MPLAB mindi verglichen. Des Weiteren wird überprüft, wie die Schaltung vereinfacht werden kann. Dies erfolgt einerseits durch Symmetrien der Schaltung, was dazu führt, dass Komponenten zusammengefasst werden können. Andererseits auch durch Weglassen aufgrund von vernachlässigbarem Einfluss auf die Simulationen. Die Softwarestruktur orientiert sich am gängigen Prinzip der MVC(Model-View-Control). Diese Strukturierung begünstigt einen modularen Aufbau, was die Software einfach erweiterbar macht und zudem eine unkomplizierte Wartung ermöglicht. Des Weiteren wird die Software anhand des Testkonzepts Modul für Modul getestet.

Im Fokus des Fachberichts befindet sich die Software, da das zu entwickelnde Produkt eine Simulationssoftware ist. Der Fachbericht ist nach dem Top-Down-Prinzip aufgebaut. In einem ersten Schritt wird die Software als Ganzes beschrieben. In den darauffolgenden Kapiteln befinden sich die Dokumentationen der beiden Teile der Software. Die Software wird aufgegliedert in den Teil Benutzeroberfläche und den Teil Ermittlung der Einfügedämpfung. Um den Fachbericht schlank zu gestalten, werden sämtliche theoretische Grundlagen im Anhang platziert. Falls in Kapiteln entsprechende Theorie wichtig ist wird darauf verwiesen.

## 2 Grundlagen

### 2.1 Elektrotechnik

#### 2.1.1 Aufbau eines EMI-Filters

Ein EMI-Filter ist ein lineares Netzwerk aus R, L, C Bauteilen und einem Transformator. Somit besitzen sie eine reziproke Übertragungssymmetrie, was eine einfache Berechnung von verschiedenen Zusammenhängen erlaubt. Einem reziproken Netzwerk ist die Betriebsrichtung gleichgültig. Die Schaltung 2.1 Original Schaltung [1] zeigt den Filteraufbau, wie er der Aufgabenstellung zu entnehmen ist. Um das Gegentaktrauschen und das Gleichtaktrauschen bestimmen zu können, werden die beiden Schaltungsäquivalente gebildet.

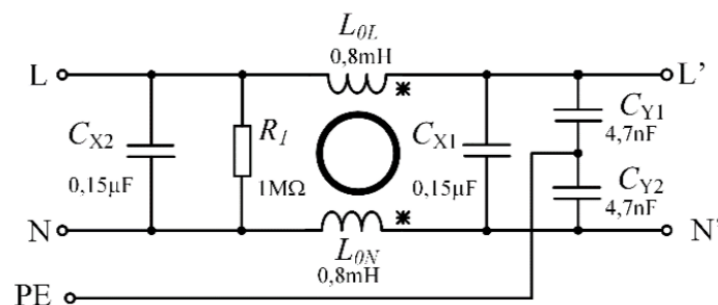
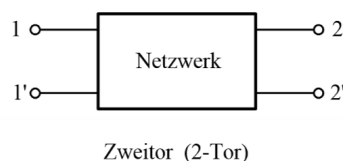


Abbildung 2.1: Original Schaltung [1]

#### 2.1.2 2-Tore

In der Elektrotechnik sind 2-Tore, Systeme mit 2 Klemmenpaaren (Tore). Diese sind intern aus beliebigen R (ohmischer Widerstand), L (Induktivität), C (Kapazität) und M (Gegeninduktivität) Komponenten aufgebaut. Dabei wird ein Tor als Eingang für ein Elektrisches Signal verwendet. Folglich wird bei den Ausgangsklemmen das Ausgangssignal abgegriffen. Es wird zwischen aktiv und passiv unterschieden. Bei aktiven Zweitoren wird die Leistung, die beim Eingang eingespeist wird, verstärkt. Im Vergleich dazu wird beim passiven die Leistung am Ausgang kleiner.

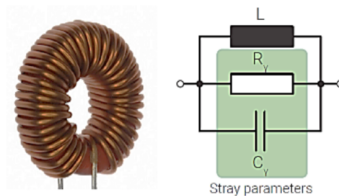


#### 2.1.3 Parasitäre Paramter

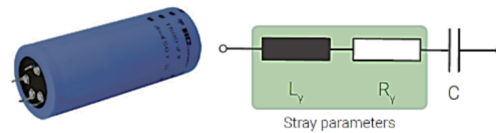
In diesem Unterkapitel werden grundsätzlich die Einflüsse und Eigenschaften von Parasitären Parametern in Realen Bauteilen, besonders Spule und Kondensator, erklärt

Ideale Bauteile beschreiben eine Funktion. Da reale Bauteile aus Materialien mit physikalischen Eigenschaften bestehen treten bei der Umsetzung dieser Funktion unerwünschte Nebeneffekte auf. Sie entstehen, weil die einzelnen Bauteile im betrieb elektrische Felder oder Magnetfelder erzeugen. Oder einfach durch die Leitfähigkeit eines Materials. Diese physikalisch bedingten Effekte werden als parasitär bezeichnet. Sie treten als Widerstand, Induktivität oder Kapazität

auf. Da sie gut klassifiziert werden können werden sie als Parameter bezeichnet. Um eine Schaltung präzise zu simulieren ist es unerlässlich, die elektrischen Bauelemente mit den passenden parasitären Parameter zu ergänzen. In Abbildung ?? und ?? werden die parasitären Parameter von Spule und Kondensator gezeigt. Man zeigt sie als zusätzliche Bauteile auf.



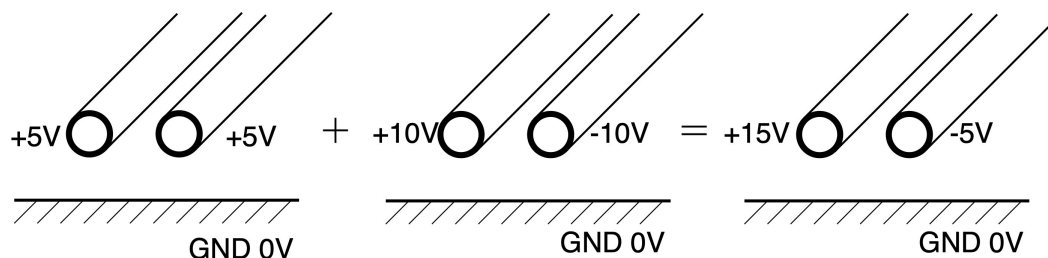
**Abbildung 2.2:** Parasitäre Elemente einer Induktivität [1]



**Abbildung 2.3:** Parasitäre Elemente einer Kapazität [1]

### 2.1.4 Gegentaktschaltung

In der realen Stromverteilung wird beabsichtigt, dass der Stromfluss über einen Zuleiter zum Verbraucher hinein-, respektive über einen Ableiter herausgeführt wird. Diese Art der Signalübertragung wird als Gegentakt-Betrieb bezeichnet. Im realen Stromnetz ist allerdings auch der sogenannte Gleichtakt-Betrieb vorhanden. Dabei wirken alle Leiter als Zuleiter, der Gesamte Strom wird durch die Erde weggeleitet. Durch das Gesetz der Superposition ist es möglich, den Gleichtakt- und den Gegentaktanteil getrennt voneinander zu betrachten. Dieses Phänomen wird Anhand der Abbildung 2.4 klar.



**Abbildung 2.4:** Beispiel der aufgetrennten Leitung

An einen geerdeten Verbraucher sind 2 Phasen angeschlossen. An der Zuleitung liegt eine Spannung von 15 Volt an, an der Rückleitung liegen -5 Volt an. Diese Leitung wird nun aufgeteilt in eine Gleichtaktleitung, bei welcher über beide Phasen 5 Volt eingespeist werden und in eine Gegentaktleitung, in welcher durch die Zuleitung 10 Volt, respektive in der Rückleitung -10V eingespeist werden. Während in der Gleichtaktleitung die Addierten 10 Volt gegenüber der Erde anliegen, werden sie in der Gegentaktleitung abgeführt.

### 2.1.5 Einfügedämpfung

Einfügedämpfung (engl. Insertion loss) wird in Dezibel (dB) angegeben. Im folgendem wird beschrieben, wie die Einfügedämpfung für die Software errechnet wird. Die Einfügedämpfung werden analytisch ermittelt. Im ersten Schritt werden die Berechnungen in MATLAB getätigt und geplottet. Diese Plots werden dann mit Simulationen in MPLAB Mindi verglichen um festzustellen ob diese korrekt sind. Nach Überprüfung auf vollständigkeit und korrektheit der

Berechnungen, können diese in das Java implementiert werden. Um die Einfügungsdämpfung bestimmen zu können, wird das Model der 2-Tore verwendet. Einzelne Schaltungsteile werden in der Kettenmatrix ?? abgebildet, welche dann durch Kaskadierung der einzelnen Kettenmatrixen zusammengeführt werden. Die Einfügungsdämpfung werden aus den Streuparameter?? abgeleitet, welche direkt aus der Kettenmatrix berechnet werden.[2].

### 2.1.6 Streuparameter

Der S-Parameter  $S_{21}$  gibt den Transmissionsgrad der Wellen an, die vom Tor 1 zum Tor2 übertragen wird. Die S-Parameter sind abhängig von den Bezugswiderständen (Innenwiderstand der Quelle sowie Lastwiderstand). In unserem Fall sind die Bezugswiderstände mit 50Ohm gegeben.

$$IL = |H(j\omega)| = 20 * \log\left(\frac{|U_{20}|}{|U_2|}\right) \quad (2.1)$$

In der Definition kann das Spannungsverhältnis durch den Streuparameter ?? (S-Parameter)  $S_{21}$  ersetzt werden 3.1.

$$IL = -20 * \log(|S_{21}|) \quad (2.2)$$

Dieser Parameter beschreibt den Transmissionsgrad des Filters. Die Einfügungsverluste wären auch mit dem Verhältnis von eingehende zu abgegebene Leistung zu berechnen, jedoch eignet sich diese Methode mehr beim messtechnischen bestimmen der Einfügungsverluste.

Die Streuparameter (S-Parameter) werden in der Hochfrequenztechnik verwendet, um das Verhalten von n-Toren zu beschreiben. Bei einem 2-Tor sind vier Streuparameter von nöten um das Verhalten zu beschreiben. Sie beschreiben die Transmission von Tor 1 zu Tor 2, sowie von Tor 2 zu Tor 1. Des weiteren zeigen sie die Reflexion an den Toren auf. Abbildung 2.5 2-Tor Wellengrößen und Anschlussleitungen [2] zeigt die Streuparameter an einem 2-Tor.

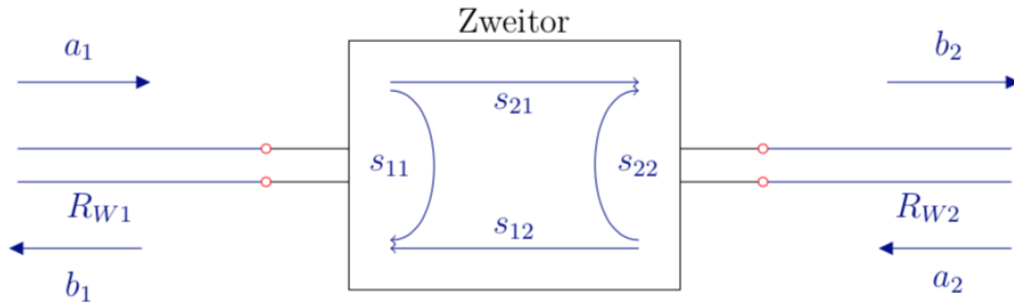


Abbildung 2.5: 2-Tor Wellengrößen und Anschlussleitungen [2]

Bei den S-Parameter werden die Eingangs- und Ausgangsgrößen nicht direkt anhand elektrischer Ströme und Spannungen beschrieben. Sie werden mithilfe von Wellengrößen beschrieben, wobei  $a_i$  die einlaufenden Wellen sind und  $b_i$  die reflektierenden Wellen. Der Index  $i$  stellt den Torindex dar. Formel 2.3 und 2.4 zeigen wie die Wellengrößen  $a_i$  sowie  $b_i$  definiert sind.

$$a_i = \frac{U_i + R_{Wi} I_i}{2 * \sqrt{R_{Wi}}} \quad (2.3)$$

$$b_i = \frac{U_i - R_{Wi} I_i}{2 * \sqrt{R_{Wi}}} \quad (2.4)$$

Die Wellengrößen gelten nur für den gegebenen Bezugswiderstand  $R_{Wi}$ . Der Bezugswiderstand kann einerseits der Innenwiderstand der angeschlossenen Quelle sein oder der Lastwiderstand der angeschlossenen Last.

Aus der Abbildung 2.3 lässt sich folgende Streumatrix darstellen (Formel 2.5):

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} * \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \quad (2.5)$$

Die Elemente der S-Matrix sind:

$$s_{11} = b_1/a_1 \text{ Eingangreflexionsfaktor bei angepasstem Ausgang (} a_2=0 \text{)} \quad (2.6)$$

$$s_{12} = b_1/a_2 \text{ Rückwärtstransmissionsfaktor bei angepasstem Eingang (} a_1=0 \text{)} \quad (2.7)$$

$$s_{21} = b_2/a_1 \text{ Vorwärtstransmissionsfaktor bei angepasstem Ausgang (} a_2=0 \text{)} \quad (2.8)$$

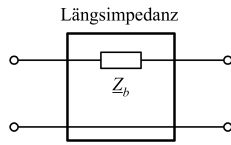
$$s_{22} = b_2/a_2 \text{ Ausgangsreflexionsfaktor bei angepasstem Eingang (} a_1=0 \text{)} \quad (2.9)$$



### 2.1.7 Kettenmatrix

Die Kettenmatrix ist eine Variante, um das Verhalten von 2-Toren zu beschreiben. Andere Varianten sind die Z-Matrix oder die Y-Matrix. Die Kettenmatrix hat jedoch den Vorteil, dass man in Serie geschaltene 2-Tore ohne grossen Aufwand zusammen rechnen kann. Sobald man die einzelnen Kettenmatrizen gebildet hat und die Schaltung soweit vereinfacht ist, dass nur noch in Serie geschaltene ABCD-Matrizen vorzufinden sind, können diese miteinander multipliziert werden. Das Matrix-Produkt stellt dann die Kettenmatrix der Gesamtschaltung dar. Folgende gängigen Schaltungen helfen die Kettenmatrizen der einzelnen Schaltungsteilen zu bilden

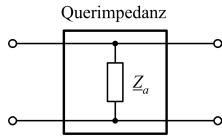
Die Längsimpedanz lässt sich anhand der Kettenmatrix  $A_L$  (Formel 2.10) darstellen



$$A_L = \begin{bmatrix} 1 & Z_b \\ 0 & 1 \end{bmatrix} \quad (2.10)$$

**Abbildung 2.6:** Längsimpedanz [3]

Die Querimpedanz lässt sich anhand der Kettenmatrix  $A_Q$  (Formel 2.11) darstellen



$$A_Q = \begin{bmatrix} 1 & 0 \\ \frac{1}{Z_a} & 1 \end{bmatrix} \quad (2.11)$$

**Abbildung 2.7:** Querimpedanz [3]

Sobald die Kettenmatrix einer Schaltung gebildet wurde, kann diese direkt in die Streuparameter umgewandelt werden. Der  $s_{21}$  Parameter kann wie in Formel 2.12 beschrieben, durch einsetzen der Kettenmatrix bestimmt werden. Für den Widerstand  $R_w$  muss die verwendete Bezugsimpedanz eingesetzt werden.

$$s_{21} = \frac{2}{A_{11} + \frac{A_{12}}{R_w} + A_{21} * R_w + A_{22}} \quad (2.12)$$

Die Indexierung der Kettenmatrix wird in Abbildung 2.13 gezeigt.

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad (2.13)$$

## 2.2 Programmieren

### 2.2.1 MVC-Struktur

Das MVC-Framework wird zur Softwarestrukturierung verwendet. Durch diese Strukturierung werden die Berechnungen der Daten (eng. model), die Steuerung (engl. controller) und dessen graphischer Repräsentation (engl. view) getrennt. In der Abbildung (TODO) ist dieser Aufbau in einem Beispielklassendiagramm dargestellt.

Der Ablauf dieser Struktur ist wie folgt:

1. Benutzereingabe löst Event aus
2. Die Aktion wird dem Controller übergeben. Dieser holt die Daten in der View, leitet diese dem Model weiter und löst die Berechnungen aus
3. Das Model führt die Berechnungen aus und informiert den Observer
4. Das Model führt die Berechnungen aus und informiert den Observer
5. Der Observer löst ein Event in der View aus. View kann die Daten vom Model holen und Ausgeben

### 3 Software

Die Software ist als Model-View-Controller Entwurf aufgebaut. Diese Struktur wird im Anhang erläutert. Somit sind die Berechnungen von der Benutzeroberfläche getrennt. Während im Model die Schaltbilder aufgebrochen werden um die Einfügedämpfung zu berechnen wird die Benutzeroberfläche in einzelne Panels aufgeteilt. Diese Panels beinhalten jeweils eine Grundfunktion der Software. Im Klassendiagramm (Abbildung ?? ) sind die Inhalte der verschiedenen Programmteile, sowie deren Zusammenspiel ersichtlich. Der Controller wird standardmässig nur zur Übergabe der Befehle zwischen View und Model verwendet. Ausserhalb dieser Struktur befindet sich noch das Framework, welches die gesamte Software initialisiert. Im Klassendiagramm sind auch noch weitere externe Klassen vorhanden, welche der Software verschiedene Methoden zur Verfügung stellen. Diese gehören nicht zum MVC Entwurf da sie allgemeingültig sind und nicht für ein spezifisches Problem erstellt wurden.

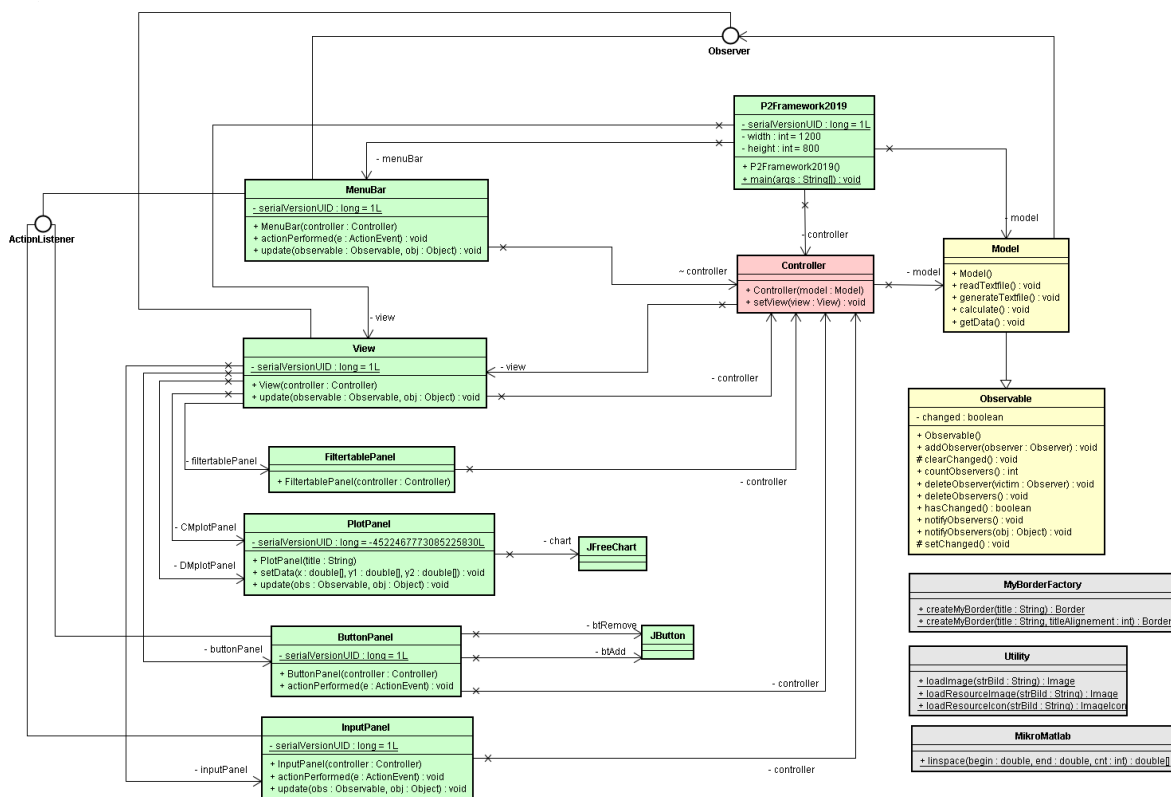


Abbildung 3.1: Klassendiagramm

### 3.1 Ermittlung der Einfügedämpfung

In diesem Kapitel wird erläutert, wie von den Schaltungen für Gegentakt und Gleichtakt, die Einfügedämpfung(engl. Insertion Loss, IL) berechnet wird. Unter anderem wird Modular beschrieben, wie die Berechnungen im Model der Software implementiert ist.

Die Einfügedämpfung ist eine Grösse, die verwendet wird, um das Verhalten einer Schaltung zu beschreiben. In diesem Fall handelt es sich um einen EMI-Filter. Sie beschreibt das Verhältnis zwischen der eingehenden Leistung zur abgegebenen Leistung. Es handelt sich um eine logarithmische Grösse. Um die Einfügedämpfung in einem Bereich von bis 30MHz abzudecken wird die Formel 3.1 verwendet. In der Formel 3.1 wird die Einfügedämpfung mittels Streuparameter(S-Parameter) berechnet. Die Streuparameter werden in den Grundlagen(Verweis) ausführlich beschrieben. Der Streuparameter  $S_{21}$  beschreibt im wesentlichen den Transmissionsgrad des eingehenden Signals. Der Transmissionsgrad beschreibt, welchen Anteil des eingehenden Signals am Ausgang wieder herauskommt.

$$IL = -20 * \log(|S_{21}|) \quad (3.1)$$

Die Einfügedämpfung wird auf Gleichtakt und Gegentakt aufgetrennt. Dieses Prinzip ist eine gängige Methode und wird im Anhang(Verweis) ausführlich beschrieben. Für die beiden Schaltungen, welche der Aufgabenstellung(Verweis Aufgabenstellung) entnommen wurden, werden die S-Parameter berechnet. Damit dies möglich ist, wurden die Schaltungen weitgehend vereinfacht. Dies wird in den Grundlagen(Verweis) Schritt für Schritt beschrieben. Aus den Vereinfachungen gehen folgende Schaltungen hervor(Abbildung ??, ??).

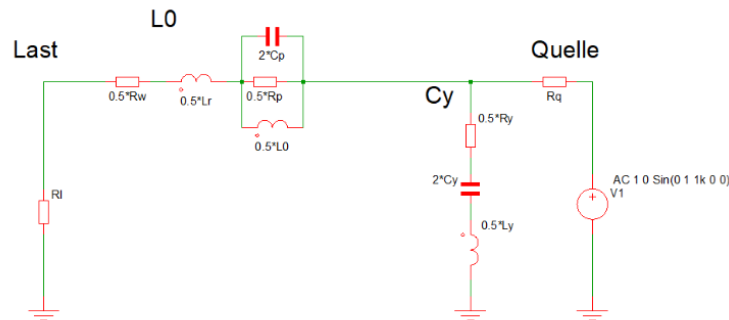


Abbildung 3.2: reduzierte Gleichtaktschaltung

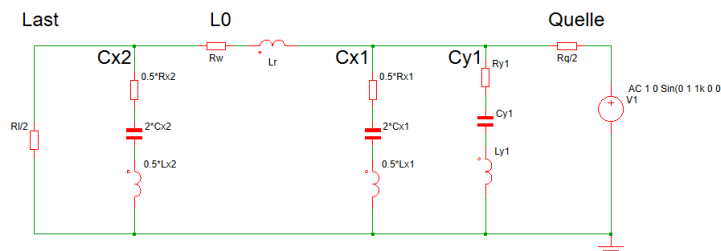


Abbildung 3.3: reduzierte Gegentaktschaltung

Um die beiden Schaltungen anhand der Streuparameter zu beschreiben, werden die einzelnen Schaltungsteile anhand von Kettenmatrizen beschrieben (siehe Anhang Verweis Kettenmatrix). Wie im Grundlagen (Verweis Längs- Querimpedanz) beschrieben, werden dazu die Kettenmatrizen für Längsimpedanzen und Querimpedanzen verwendet. Dies führt dazu, dass die einzelnen Schaltungsteile entweder eine Längsimpedanz oder eine Querimpedanz bilden müssen. Die Aufteilung der einzelnen Schaltungsteile in Längs-Querimpedanz und ist in den Abbildungen (Verweis Aufteilung 1 und 2) grafisch dargestellt, wobei „LI“ eine Längsimpedanz kennzeichnet und „QI“ eine Querimpedanz.

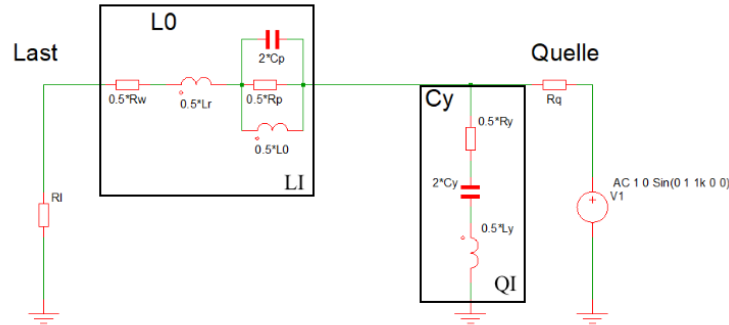


Abbildung 3.4: Einteilung der Gleichtaktschaltungsteile

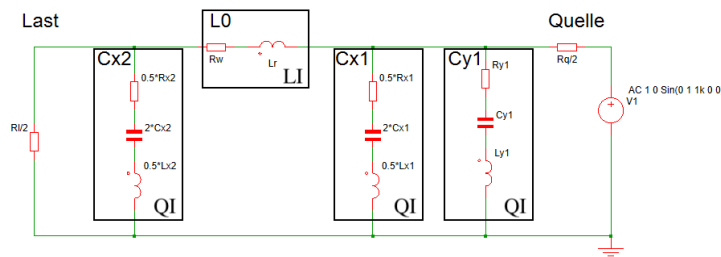


Abbildung 3.5: Einteilung der Gegentaktschaltungsteile

Im nächsten Schritt werden die Impedanzen der einzelnen Schaltungsteile gebildet, welche in die passenden Kettenmatrizen eingesetzt werden. Um die Kettenmatrizen der Gesamtschaltungen zu bilden, werden die einzelnen Kettenmatrizen durch Kaskadieren zusammengefasst (Verweis Theoretische Grundlagen: Kettenmatrix). Die Kaskadierung entspricht im wesentlichen der Matrixmultiplikation der Kettenmatrizen.

$$s_{21} = \frac{2}{A_{11} + \frac{A_{12}}{R_w} + A_{21} * R_w + A_{22}} \quad (3.2)$$

Anhand der Kettenmatrizen der beiden Schaltungen, wird mit der Formel 3.2 der Streuparameter  $S_{21}$  berechnet. A1 bis A4 entsprechen den einzelnen Einträge der Matrix.  $R_w$  ist die Bezugsimpedanz. Die Bezugsimpedanz bezieht sich auf die Innenimpedanz der Quelle und die Lastimpedanz. Bezüglich der Aufgabenstellung (Verweis Aufgabenstellung) ist dieser auf 50 Ohm festgelegt. Ausserdem ist es wichtig, dass die beiden Impedanzen gleich gross sind damit die Schaltung Reziprok ist (Verweis Theoretische Grundlagen: Reziproke Schaltungen). (Wieso ?) Bei der reduzierten Gegentaktschaltung (Abbildung ??) werden die beiden Impedanzen mit

25 Ohm aufgeführt. Dies ergibt sich durch das Zusammenfassen der Gegentaktschaltung.

### 3.2 Zusammenfassen der Schaltungen

Folgendes Kapitel zeigt Schritt für Schritt auf, wie die gegebenen Schaltungen vereinfacht werden können. Zudem wird beschrieben welche Komponenten vernachlässigt werden können. Der erste Abschnitt behandelt die Gleichtaktschaltung und in einem zweiten Abschnitt wird die Gegentaktschaltung behandelt.

#### Gleichtaktschaltung

Abbildung 3.6 zeigt die Schaltung aus der Aufgabenstellung.

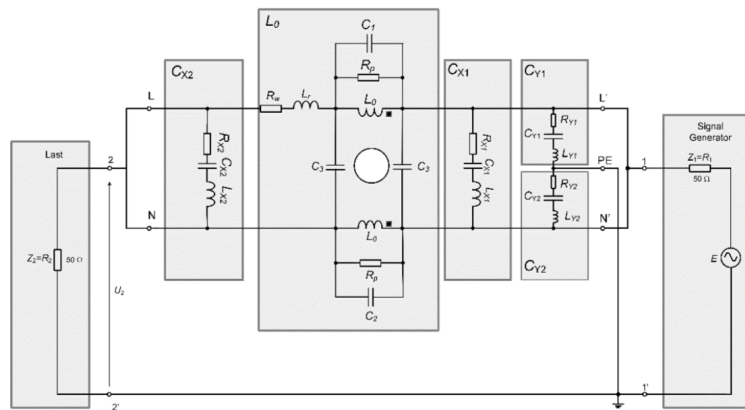


Abbildung 3.6: Originale Gleichtaktschaltung[1]

Die Originalschaltung ist mit den Komponenten  $R_w$  und  $L_r$  ergänzt worden, sodass sie symmetrisch ist (siehe Abbildung 3.7). Dies macht es möglich, dass die Schaltung zur Simulation wie folgt vereinfacht werden kann.

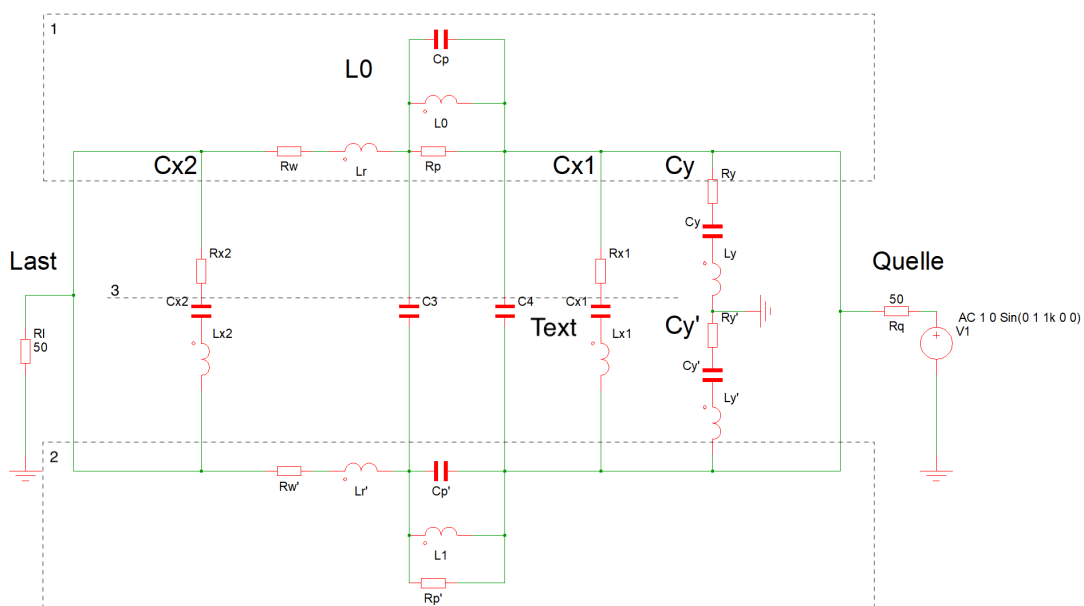


Abbildung 3.7: Ergänzte Gleichtaktschaltung

Da der obere (siehe Abbildung 3.7, Nr. 1) und untere Strang (siehe Abbildung 3.7, Nr. 2) identisch sind und es keinen Potentialunterschied zwischen ihnen gibt, kann die Schaltung, wie folgt, zusammen gefasst werden (siehe Abbildung ??). Die Schaltung wird entlang der Symmetrie-Achse (siehe Abbildung 3.7, Nr. 3) aufgetrennt. Somit fallen die Kondensatoren  $C_3$ ,  $C_4$ ,  $C_{x1}$  und  $C_{x2}$  komplett weg. Die übrigen Komponenten von  $L_0$  bilden eine Parallelschaltung, welche sich durch halbieren der Widerstände und Induktivitäten und verdoppeln der Kapazitäten zusammenfassen lässt. Zusätzlich werden die beiden  $C_y$  und  $C'_y$  parallel auf das Bezugspotential geschaltet. Da  $C_y$  und  $C'_y$  identisch sind, werden sie wie in Abbildung ?? zusammengefasst. Diese vereinfachte Schaltung bildet die Grundlage für die Berechnungen der Software.

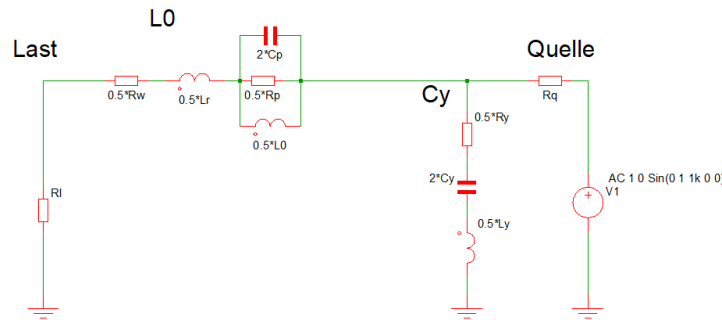


Abbildung 3.8: Vereinfachte Gleichtaktschaltung

### Gegentaktschaltung

Abbildung ??

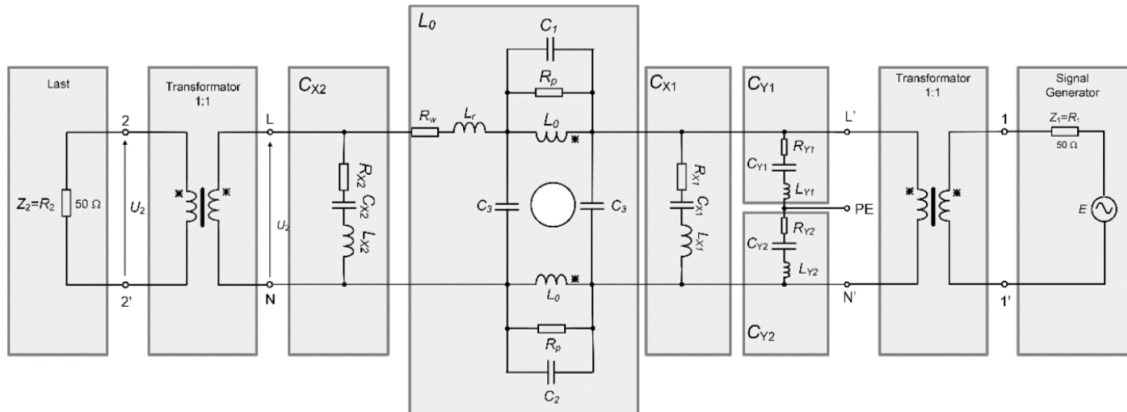


Abbildung 3.9: Ergänzte Gegentaktschaltung

## 3.3 Software

In diesem Kapitel wird die Umsetzung der Software dokumentiert und wie die in den Grundlagen beschriebenen Themen eingebaut werden. Die Software ist nach dem MVC Framework (TODO:Verweis) aufgebaut. Das Klassendiagramm der Software befindet sich im Anhang (TODO: Verweis)

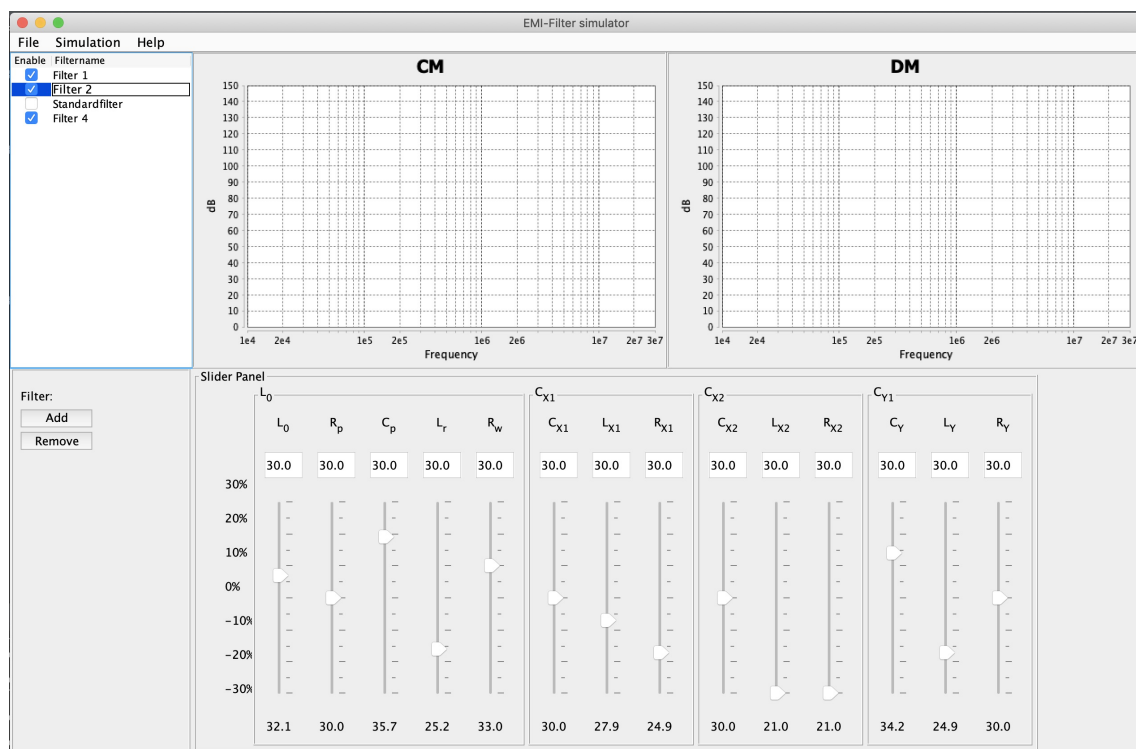


Abbildung 3.10: Benutzeroberfläche unfertig

Die Oben abgebildete Benutzeroberfläche ist die Maske, welche sich beim aufstauen der Software auf dem Bildschirm präsentiert. Wie die meisten gängigen Applikationen verfügt die GUI eine am oberen Rand platzierten Menubar, um Dateien zu verwalten sowie Hilfestellung zu bieten. Im Zentrum ist im oberen Bereich das Plotpanel platziert, welches die Simulation sowohl im Gleichtakt (CM), als auch im Gegentakt (DM) als grafische Kurve darstellt. Gleich unterhalb befindet sich das Inputpanel, es ermöglicht mittels den Textfeldern eine effiziente Eingabe der Parameter. Mittels den Sleidern können die Parameter prozentual angepasst werden. Die Filtertabelle links dient zur Verwaltung der Filterprofilen.

### 3.3.1 View

Die View ist für die Darstellung der Daten und die Verarbeitung der Benutzereingaben zuständig. Zur View gehören die Panels: InputPanel (1), FiltertablePanel (2), ButtonPanel (3), Menubar (4) und PlotPanel (5). In diesem Abschnitt werden die Panels dokumentiert Diese bilden zusammen die Benutzeroberfläche, die in der Abbildung (TODO verweis) ersichtlich ist. Im Klassendiagramm (TODO Verweis Anhang) sind alle Klassen die zur View gehören, mit der Farbe (TODO: Farbe) markiert.

#### Menu

Die Menübar verfügt über drei Menüs das File-, Simulation- und ein Help-Menü.

Das File-Menü dient zur Datenverwaltung, es erlaubt Filterprofile mittels SSaveäbzuspeichern und "Loadßu laden. Das Programm wandelt das Filterprofil in ein Komma-getrenntes Textfile.

Mittels SSave" wird das angewählte Filterprofil abgelegt. Es öffnet sich ein Dialog-Fenster, ein File Chooser, dieser erlaubt es den gewünschten Speicherpfad auszuwählen und die Datei zu benennen. Die Eingabe wird mit SSave" bestätigt. Nun ladet das Programm den zugehörigen



Array welcher das Filterprofil enthält, mithilfe des Printwriter in ein Textfile. Das Textfile wird nun unter dem gewünschten Pfad abgelegt.

**Inputpanel**

**Plotpanel**

**Filterpanel**

**Buttonpanel**

**3.3.2 Model**

**3.3.3 Controller**

## 4 Testkonzept

### 4.1 Aufbau

Das Testkonzept soll eine einwandfreie Funktion unserer Software garantieren und eine einfache Bedienung ermöglichen. Dies wird durch die unten aufgeführten Tests realisiert.

Einen zentralen Anteil haben die kontinuierlichen Tests. Diese werden von Beginn weg durchgeführt, um Folgefehler zu vermeiden, was uns schlussendlich Zeit spart, weil dann weniger nach Fehlern gesucht werden.

Bei der Software wird dies erreicht indem einzelne Methoden und Klassen einzeln ausgeführt werden, um die Funktion zu testen. Ausserdem wird so oft wie möglich, neu eingefügter Code vom ganzen Team angeschaut, damit alle Teammitglieder die Funktionsweise der Software verstehen.

Im Teil der elektrotechnischen Grundlagen ist es von grosser Bedeutung die ausgerechneten Werte zu hinterfragen und auf ihre Richtigkeit zu überprüfen, damit keine falschen Werte in die Software gelangen. Dafür wird angenommen, dass die dafür benutzten Rechenprogramme (MATLAB, MPLAB) stimmen.

Je nach Bedarf werden die Fachcoaches kontaktiert, falls noch Unklarheiten bei den Grundlagen oder der Software bestehen.

Nach Vollendung der Version 0.9.5 wird die Software mit internen Tests vom gesamtem Team komplett durchgetestet. Dafür wird ein Testprotokoll verwendet, das im Anhang zu finden ist. Die Software wird zum einen mit einem Kompatibilitätstest gefordert. Bei diesem Test werden Eingaben getätigt, die unsere Software an die Grenzen bringen dürfte. Es werden auch Fehleingaben gemacht, um zu sehen wie die Software darauf reagiert. Ein anderer Aspekt, um die Kompatibilität zu prüfen, ist die Verwendung von verschiedenen Betriebssystemen (Mac OS, Windows). Um eine saubere Darstellung auf allen Displays zu gewährleisten, werden zudem Bildschirme mit unterschiedlichen Auflösungen verwendet (Full HD, 4K).

Nach den internen Tests wird die Software dem Auftraggeber Dr. Luca Dalessandro zur Verfügung gestellt, damit er die Möglichkeit hat seine Meinung und Ideen noch einmal einzubringen. Des Weiteren wird mit externen Testpersonen zusammengearbeitet. Dabei werden Experten und auch Fachfremde Tester gesucht, wobei die Fachfremden eine grundlegende Einführung über EMI-Filter erhalten, damit sie in der Lage sind, zu verstehen wofür diese Software überhaupt entwickelt wurde. Über die Funktionsweise der Benutzeroberfläche erhalten die Testpersonen jedoch keine Einführung. Damit kann man sehr gut überprüfen, wie einfach und intuitiv die Softwarebedienung ist. Dafür wird das bereits erwähnte Testprotokoll verwendet, in das die Testpersonen ihre Meinung und allfällige Anregungen hineinschreiben können. Die Fachpersonen werden ein vereinfachtes Testprotokoll erhalten, indem nicht alle einzelnen zu testenden Punkte aufgeführt sind. Diese Protokolle werden ausgewertet und anschliessend wird sich im Team über allfällige Änderungen ausgetauscht. Vor der Abgabe wird die Software noch ein letztes Mal getestet und ein Abnahmeprotokoll erstellt, bis die Software den Weg zum Auftraggeber findet.

## 4.2 Erwartungen

Die zu erwartenden Testresultate variieren stark von den Testperson, die die Software testet. Die internen Tests werden die groben Fehler herausfiltern und schaffen eine stabile Grundlage auf die aufgebaut werden kann. Zudem wird überprüft, ob die alle Ziele erreicht wurden. Bei den Fachpersonen wird das Feedback höchst wahrscheinlich sehr umfangreich ausfallen. Es ist zu erwarten, dass Fehler entdeckt werden, die noch nicht bekannt sind oder die Software gar zum Absturz gebracht wird. Die Fachfremden Tester werden dies nicht erreichen, jedoch erhalten wir eine hilfreiche Rückmeldung, was die Benutzerfreundlichkeit betrifft, weil diese Personen einen anderen Blick auf das grosse ganze haben. Das Feedback des Auftraggebers wird sehr detailliert ausfallen, weil er genaue Vorstellungen hat was er von dem Produkt haben möchte. Es wird sich, aber vermutlich mehrheitlich, um die Funktionen drehen und nicht welche Fehler es gibt.

## 4.3 Validierung

Testergebnisse darstellen und Interpretieren

Ebenfalls wird hier beschrieben welche Werte wir mit der Simulation und mit Matlab erreicht haben

## 5 Schluss

## Literatur

- [1] L. Dalessandro, „EMI-Filter“, *Windisch: Hochschule für Technik der Fachhochschule Nordwestschweiz (FHNW)*, 21. März 2019.
- [2] S. Rupp, A. Maier, „Hochfrequenztechnik Teil 2 - Anwendungen“, *Duale Hochschule Baden-Württemberg*, 21. März 2019. Adresse: [http://www.lehre.dhbw-stuttgart.de/~srupp/HF/Hochfrequenztechnik\\_T2ELN3001\\_Teil\\_2\\_SR.pdf](http://www.lehre.dhbw-stuttgart.de/~srupp/HF/Hochfrequenztechnik_T2ELN3001_Teil_2_SR.pdf).
- [3] P. Niklaus, „Lerndokument: 2-Tor Tabellen“, *Windisch: Hochschule für Technik der Fachhochschule Nordwestschweiz (FHNW)*, 21. März 2019.

## 6 Anhang

### 6.1 Testkonzept

# Testprotokoll Team 1

Name des Testers:

---

Datum :

---

Softwareversion: 0.9




---




Test Art (intern oder extern):

---




**Test Übersicht**




<b>1. Test</b>	Software öffnen und schliessen. Schieberegler und Buttons auf die Funktion prüfen. Das Fenster grösser kleiner machen.
<b>2. Test</b>	Das Aussehen der GUI betrachten. Auf Vollständigkeit testen. (Gesamteindruck)
<b>3. Test</b>	Kontrollieren, ob der Plot gezeichnet wird. Und Filtertypen gespeichert werden können.
<b>4. Test</b>	Die Menu-Funktionen testen. <del>Shurcuts</del> ausprobieren
<b>5. Test</b>	Fehleingaben machen.
<b>6. Test</b>	Verschiedene Betriebssysteme und Displays verwenden (falls vorhanden).
<b>7. Test</b>	Code auf Übersichtlichkeit prüfen. (Experten)


Test Nr.	Der Test	Bewertung	Anmerkungen (Fehler)
1.	Software öffnen und schliessen. Schieberegler und Buttons auf die Funktion prüfen. Das Fenster grösser kleiner machen.	Gut  Mässig  Schlecht 	
<b>Kommentar / Verbesserungsvorschlag</b>			

Test Nr.	Der Test	Bewertung	Anmerkungen (Fehler)
2.	Das Aussehen der GUI betrachten.	Gut  Mässig  Schlecht 	
<b>Kommentar / Verbesserungsvorschlag</b>			






Test Nr.	Der Test	Bewertung	Anmerkungen (Fehler)
3.	Kontrollieren, ob der Plot gezeichnet wird. Und Filtertypen gespeichert werden können.	Gut  Mässig  Schlecht 	
Kommentar / Verbesserungsvorschlag			




Test Nr.	Der Test	Bewertung	Anmerkungen (Fehler)
4.	Die Menu-Funktionen testen	Gut  Mässig  Schlecht 	
Kommentar / Verbesserungsvorschlag			

Test Nr.	Der Test	Bewertung	Anmerkungen (Fehler)
5.	Fehleingaben machen.	Gut  Mässig  Schlecht 	

**Kommentar / Verbesserungsvorschlag**

Test Nr.	Der Test	Bewertung	Anmerkungen (Fehler)
6.	Verschiedene Betriebssysteme und Displays verwenden (falls vorhanden).	Gut  Mässig  Schlecht 	

**Kommentar / Verbesserungsvorschlag**

Test Nr.	Der Test	Bewertung	Anmerkungen (Fehler)
7.	Code auf Übersichtlichkeit prüfen.(Experten).	Gut  Mässig  Schlecht 	
<b>Kommentar / Verbesserungsvorschlag</b>			

## Abnahmeprotokoll Software

Ersteller: \_\_\_\_\_

Auftraggeber: \_\_\_\_\_

Auftrag: \_\_\_\_\_

Version: \_\_\_\_\_

Datum: \_\_\_\_\_

### Ziele

Berechnungen und GUI getrennt	
Berechnungszeit < 500 ms	
Verstellbare Parameter	
CM-, DM-Berechnung	
CM-, DM-Darstellung	
Feinjustierung mit Schieberegler +/- 30%	
Darstellung im Frequenzbereich bis 30MHz	
Zahlenwerte könne eingegeben werden	
Mehrere Plots gleichzeitig darstellen	
Unabhängigkeit von Betriebssystemen	
Schutz vor Fehleingaben	
Anpassung von Farbe, Darstellung und Schrift	
Zoom Möglichkeit	

### Kommentare

--

### Test

Kontinuierliche Tests	
Interne Tests	
Externe Tests durch Fachpersonen und Laien	
Test der 0.9.5 durch den Auftraggeber	

Projektleiter

Auftraggeber

\_\_\_\_\_  
Unterschrift, Ort, Datum

\_\_\_\_\_  
Unterschrift, Ort, Datum