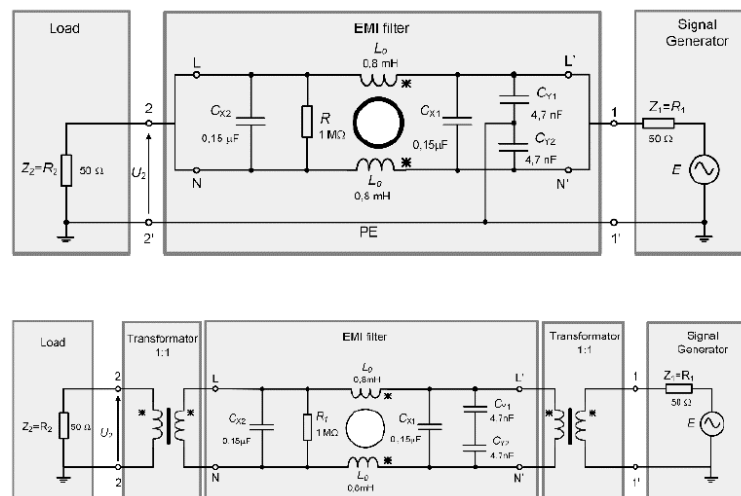


«DJ» EMI Filter für Schaltnetzteil

Pflichtenheft technischer Teil

Windisch, 23.03.2019



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
Claudio Alfaro
Simon Rohrer
Marco Binder

Version 1.0

Inhaltsverzeichnis

1	Übersicht	1
1.1	Ausgangslage	1
1.2	Projektziele	2
1.3	Lieferobjekte	3
2	Theoretische Grundlagen	4
2.1	EMI-Filter	4
2.2	Schaltungen	5
2.3	Einfügungsverluste ermitteln	7
2.4	Streuparameter	8
2.5	ABCD-Matrix	9
2.6	Berechnungsbeispiel	11
3	Softwarekonzept	13
3.1	Anforderung Software	13
3.2	GUI	13
3.2.1	Menu	14
3.2.2	Filtertabelle	15
3.2.3	CM/DM Plot	15
3.2.4	Buttonfenster	15
3.2.5	Eingabefenster	15
3.3	Softwarestruktur	16
3.3.1	Model	16
3.3.2	View	16
3.3.3	Controller	16
3.3.4	Klassendiagramm	17
3.4	Programmablauf	18
3.5	Libraries	18
4	Testkonzept	19
4.1	1. Testphase	19
4.2	2. Testphase	19
4.3	3. Testphase	19
5	Projektvereinbarung	20

References	21
6 Anhang	22

1 Übersicht

1.1 Ausgangslage

In der modernen Gesellschaft hängen von Jahr zu Jahr mehr elektrische Verbraucher am Stromnetz. Der stetig steigende Leistungsbedarf dieser Verbraucher führt dazu, dass ihre Versorgung angepasst werden muss. Aus dem konventionellen Trafo-Netzteil entstand das sogenannte Schaltnetzteil. Dieses hat grosse Vorteile gegenüber dem Trafo-Netzteil, sowohl wirtschaftlich gesehen, als auch leistungsbezogen.

Allerdings haben sie auch einen Nachteil. Sie lassen hochfrequente Störungen (EMI), entstehen, welche ins Netz zurückfliessen. Diese Störungen, welche man als Gleichtakt- und Gegentaktrauschen bezeichnet, können wiederum in anderen Verbrauchern Störungen verursachen.

Aufgrund dieses Problems wurden verschiedene Normen an Geräte gestellt um diese Emissionen zu minimieren. Daher werden in moderne Schaltnetzteile EMI-Filter verbaut. Diese, auch Netzfilter genannt, bestehen aus einem Netzwerk von aus Widerständen, Kondensatoren und Spulen. Da im Schaltnetzteil die Netzfrequenz in eine hochfrequente Spannung gewandelt wird, reagieren die Bauteile als Impedanzen und sie filtern, je nach Bauform, verschiedene hochfrequente Signale aus der Rückspeisung.

Der Auftrag von Dr. Dalessandro lautet eine Applikation zu entwickeln, welche in der Entwicklung von solchen Filtern eingesetzt werden kann. Die Anforderungen an die Applikation sind, dass die Dämpfungseigenschaften des Filters simuliert und graphisch angezeigt werden können. Dabei sollen die Gleichtakt- und die Gegentakststörungen differenziert betrachtet werden können. Ebenfalls soll die Applikation in der Lage sein, die parasitären Einflüsse der einzelnen Parameter (Bauteile) um $\pm 30\%$ zu variieren.

Dieses Pflichtenheft beschreibt die technischen Aspekte des Auftrags und liefert bereits Ideen betreffend der Umsetzung.

1.2 Projektziele

Folgende Ziele wurden festgelegt:

Ziel	Beschreibung
1. Elektrotechnik	
1.1 Frequenzverhalten	Verhalten des Rauschens in den Frequenzbereichen von 0-500kHz, von 500kHz-5MHz und von 5MHz-30MHz.
1.2 Eingangsdämpfung	(Insertion loss) Berechnung der Grösse der Eingangsdämpfung, dargestellt in einer Kurve.
1.3 Parasitäre Parameter	(Empfindlichkeitsanalyse) Erkennen und berechnen der Einflüsse der einzelnen Bauteile (Parameter).
1.4 Filterdesign	Analyse der Schaltung um diese zu Vereinfachen. Wenn möglich auf regelmässige Glieder.
2. Software	
2.1 Plattform	Die Plattform soll möglichst unabhängig von Betriebssystemen gewählt werden. Daher ist JAVA die präferierte Wahl.
2.2 Struktur	Um die Applikation möglichst erweiterbar zu halten ist das MVC-Framework eine gute Grundstruktur.
2.3 GUI	Die Gui soll vorallem die Eingangsdämpfung Anzeigen. Ebenfalls sehr attraktiv ist es, wenn die Parameter via Schieberegler direkt verstellt werden können.
2.4 Dateneingabe	Die Applikation soll möglichst vor Eingabefehlern schützen. Um die Bedienung noch weiter zu erleichtern erhält sie zudem noch einen Speicher um voreingestellte Filter zu speichern und wieder zu laden.
2.5 Datenausgabe	Der Plot soll logarithmisch dargestellt werden. Zusätzlich soll er auch exportiert werden können.
3. Wunschziele	
3.1 Verschiedene Filter	Um einen Mehrwert zu schaffen wäre es Praktisch direkt mehrere Layouts zu vergleichen.
3.2 Monte Carlo Simulation	Falls die Zeit reicht wäre die Implementation einer Monte Carlo Simulation noch sinnvoll.
3.3 Diverse Bedienungshilfen	Falls die Zeit reicht können noch diverse Bedienungshilfen und Gadgets eingebaut werden.

1.3 Lieferobjekte

Nachfolgend werden alle Lieferobjekte aufgelistet:

Beschreibung	Datum
1. Organisatorisches Pflichtenheft	24.03.19 und 04.04.19
2. Technisches Pflichtenheft	24.03.19 und 04.04.19
3. Disposition und Einleitung	02.05.19
4. Fachbericht	13.06.19

2 Theoretische Grundlagen

2.1 EMI-Filter

Das vorgegebene EMI-Filter muss bezüglich der Einfügungsverluste (Insertion Loss) untersucht werden. Die Einfügungsverluste hängen vom Gesamtrauschen der Schaltung ab. Es wird ein Ansatz verwendet, der in der Praxis weit verbreitet ist, bei welchem das Gesamtrauschen in zwei Komponenten unterteilt wird. Man spricht vom Gegen-(=Differential Mode=DM) und Gleichtaktrauschen (=Common Mode=CM). Anhand der vorgegebenen CM- und DM-Äquivalenten Schaltungen (Abbildungen 2.4, 2.5) werden die Einfügungsverluste in Funktion der Frequenz berechnet. Die Berechnungen decken einen Bereich von 0 bis 30MHz ab.

Die Einfügungsverluste sind wie folgt definiert:

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

In der Definition kann das Spannungsverhältnis durch den Streuparameter 2.4 (S-Parameter) S_{21} ersetzt werden 2.2.

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

Dieser Parameter (Formel 2.8) 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.

2.2 Schaltungen

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.

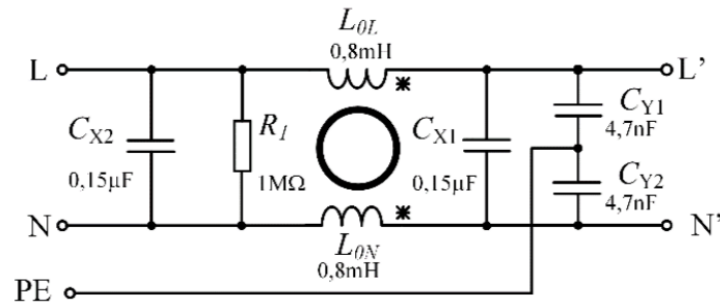


Abbildung 2.1: Original Schaltung [1]

Hierbei müssen die elektrischen Bauelemente, wie Spule und Kondensator mit den passenden parasitären Parameter ergänzt werden. In Abbildung ?? und ?? werden die parasitären Parameter von Spule und Kondensator gezeigt.

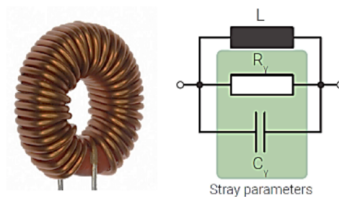


Abbildung 2.2: Parasiäre Elemente einer Induktivität [1]

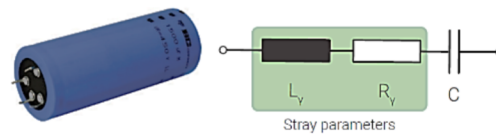


Abbildung 2.3: Parasiäre Elemente einer Kapazität [1]

Folgende Schaltungen stellen die CM- und DM-Äquivalenten Schaltungen. Da die Berechnungen in einem Bereich von bis zu 30 MHz gemacht werden, ist es notwendig die parasitären Parameter von Spule und Kondensator miteinzubeziehen.

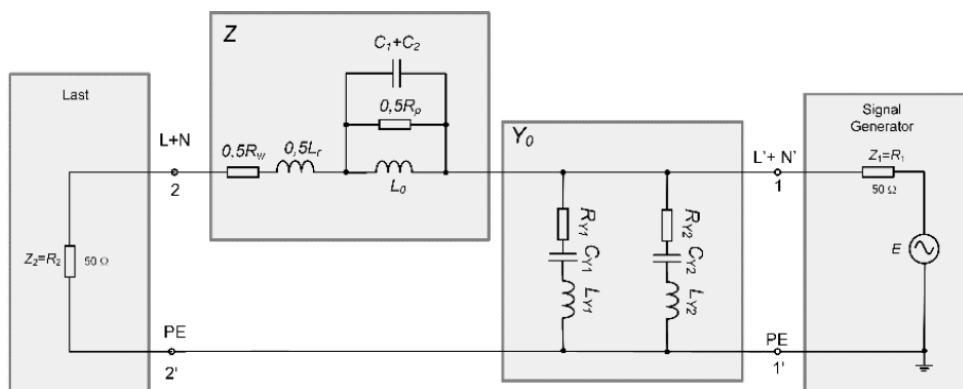


Abbildung 2.4: CM-Schaltungäquivalent [1]

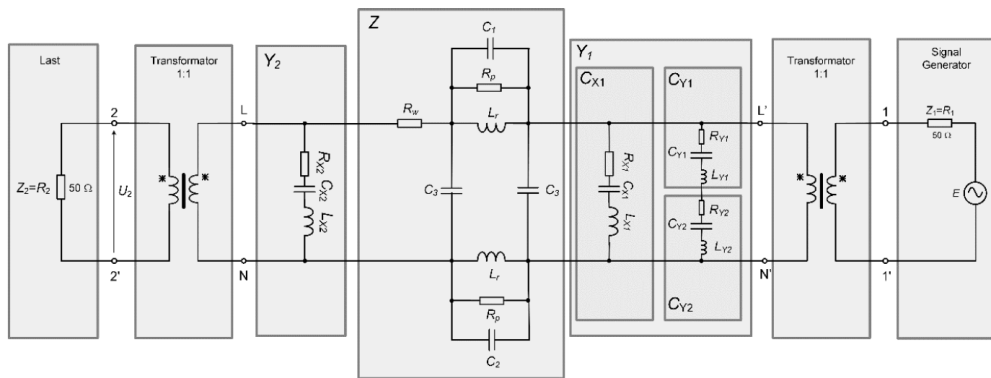


Abbildung 2.5: DM-Schaltungsäquivalent [1]

2.3 Einfüguingsverluste ermitteln

Folgende Theorieabschnitte wurden überwiegend anhand folgender Quellen zusammengestellt: [2]. Die Einfüguingsverluste werden analytisch ermittelt. Im ersten Schritt werden die Berechnungen in MATLAB gemacht. Somit können die Funktionen geplottet werden. Diese Plots werden dann mit Simulationen in MPLAB Mindi verglichen um festzustellen ob diese korrekt sind. Die vollständigen und korrekten Berechnungen können somit in Java implementiert werden. Um die Einfüguingsverluste bestimmen zu können, wird das Model der 2-Tore verwendet. Einzelne Schaltungsteile werden in ABCD-Matrixen 2.5 abgebildet, welche dann durch Kaskadierung der einzelnen ABCD-Matrixen zusammengeführt werden. Die Einfüguingsverluste werden aus den Streuparameter 2.4 abgeleitet, welche direkt aus der ABCD-Matrix berechnet werden. 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.

2.4 Streuparameter

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 2.6 2-Tor Wellengrößen und Anschlussleitungen [2] zeigt die Streuparameter an einem 2-Tor.

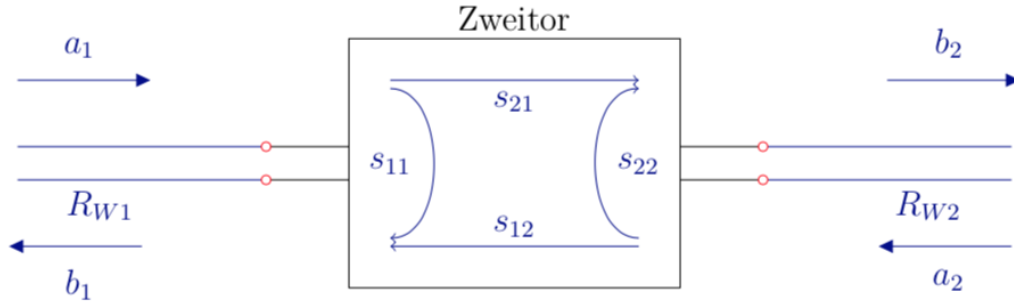


Abbildung 2.6: 2-Tor Wellengrößen und Anschlussleitungen [2]

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:

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

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

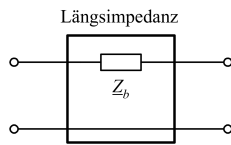
$$s_{21} = b_2/a_1 \text{ Vorwärtstransmissionsfaktor bei angepasstem Ausgang } (a_2=0) \quad (2.8)$$

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

2.5 ABCD-Matrix

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

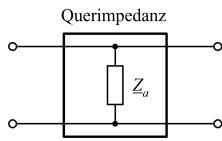
Die Längsimpedanz lässt sich anhand der ABCD-Matrix A_L (Formel 2.10) darstellen



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

Abbildung 2.7: Längsimpedanz [3]

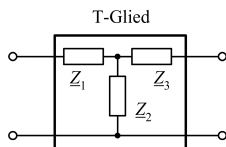
Die Querimpedanz lässt sich anhand der ABCD-Matrix A_Q (Formel 2.11) darstellen



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

Abbildung 2.8: Querimpedanz [3]

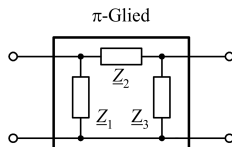
Die Impedanz eines T-Glieds lässt sich anhand der ABCD-Matrix A_T (Formel 2.12) darstellen



$$A_T = \begin{bmatrix} 1 + \frac{Z_2}{Z_3} & Z_1 + Z_3 + \frac{Z_1 Z_3}{Z_2} \\ \frac{1}{Z_2} & 1 + \frac{Z_3}{Z_2} \end{bmatrix} \quad (2.12)$$

Abbildung 2.9: T-Glied [3]

Die Impedanz eines π -Glieds lässt sich anhand der ABCD-Matrix A_π (Formel 2.13) darstellen



$$A_\pi = \begin{bmatrix} 1 + \frac{Z_2}{Z_3} & Z_2 \\ \frac{1}{Z_1} + \frac{1}{Z_3} + \frac{Z_2}{Z_1 Z_3} & 1 + \frac{Z_2}{Z_1} \end{bmatrix} \quad (2.13)$$

Abbildung 2.10: Pi-Glied [3]

Wenn die ABCD-Matrix einer Schaltung gebildet wurde, kann diese direkt in die Streuparameter umgewandelt werden. Der s_{21} Parameter kann wie in Formel 2.14 beschrieben, durch

einsetzen der ABCD-Matrix bestimmt werden. Für den Widerstand R_w muss die verwendete Bezugsimpedanz eingesetzt werden.

$$s_{21} = \frac{2}{A_{11} + \frac{A_{12}}{R_w} + A_{21} * R_w + A_{22}} \quad (2.14)$$

Die Indexierung der ABCD-Matrix wird in Abbildung 2.15 gezeigt

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \quad (2.15)$$

2.6 Berechnungsbeispiel

Folgender Abschnitt zeigt auf, wie die Einfügungsverluste ermittelt werden. Als Beispielschaltung wird das CM-Schaltungsäquivalent verwendet. Die Schaltung beinhaltet eine Queradmittanz und eine Längsimpedanz, welche in Schritt 1 zusammengefasst werden. Im Schritt 2 werden diese dann in die ABCD-Matrixen (Kettenmatrixen) eingesetzt. Durch kaskadieren der einzelnen Kettenmatrixen wird in Schritt 3 die Kettenmatrix der kompletten Schaltung erstellt. Diese Kettenmatrix kann nun zusammen mit dem Bezugswiderstand in die Definition des Streuparameters s_{21} eingesetzt werden (Schritt 4).

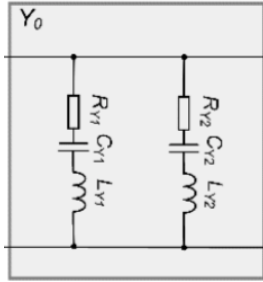


Abbildung 2.11: CM-Admittanz [1]

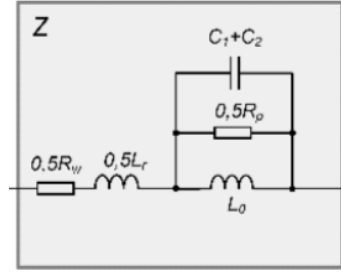


Abbildung 2.12: CM-Impedanz [1]

Schritt 1: Berechnen der Längsimpedanz Z und der Queradmittanz Y0

Für die Queradmittanz Y_0 ergibt sich die Formel

$$Y_0 = \frac{1}{R_{Y1} + \frac{1}{j\omega * C_{Y1}} + j * \omega * L_{Y1}} + \frac{1}{R_{Y2} + \frac{1}{j\omega * C_{Y2}} + j * \omega * L_{Y2}} \quad (2.16)$$

Für die Längsimpedanz ergibt sich folgende Formel

$$Z = 0.5 * R_w + j * \omega * L_r + \frac{1}{\frac{1}{0.5 * R_p} + j * \omega * L_r * (C_1 + C_2) + \frac{1}{j\omega * L_0}} \quad (2.17)$$

Schritt 2: Erstellen der ABCD-Matrixen

Somit ergeben sich die ABCD-Matrixen wie folgt

$$A_1 = \begin{bmatrix} 1 & 0 \\ Y_0 & 1 \end{bmatrix} \quad (2.18)$$

$$A_2 = \begin{bmatrix} 1 & Z \\ 0 & 1 \end{bmatrix} \quad (2.19)$$

Schritt 3: Gesamt-Kettenmatrix bilden

Die ABCD-Matrixen haben den Vorteil, dass man sie sehr unkompliziert kaskadieren kann, indem man das Produkt bildet.

$$A = A_1 * A_2 = A_2 = \begin{bmatrix} 1 & 0 \\ Y_0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & Z \\ 0 & 1 \end{bmatrix} \quad (2.20)$$

Schritt 4: s_{21} Parameter bilden

Der s_{21} Parameter ist wie folgt definiert.

$$s_{21} = \frac{2}{A_{11} + \frac{A_{12}}{R_w} + A_{21} * R_w + A_{22}} \quad (2.21)$$

Somit sind alle gesuchten Werte gegeben und der s_{21} Parameter wird durch einsetzen der Werte gebildet.

Schritt 5: Einfügungsverluste bilden

Durch einsetzen des Streuparameters s_{21} in die Definition der Einfügungsverluste lassen sich diese darstellen. Folgende Grafik 2.13 zeigt die Berechnungen in MATLAB (Mitte) im Direktvergleich mit der Grafik aus der Aufgabenstellung (Links) sowie die Simulation in MPLAB Mindi (Rechts). Die Differenzen lassen sich dadurch erklären, dass in der Aufgabenstellung die Werte der Bauelemente nicht gegeben sind.

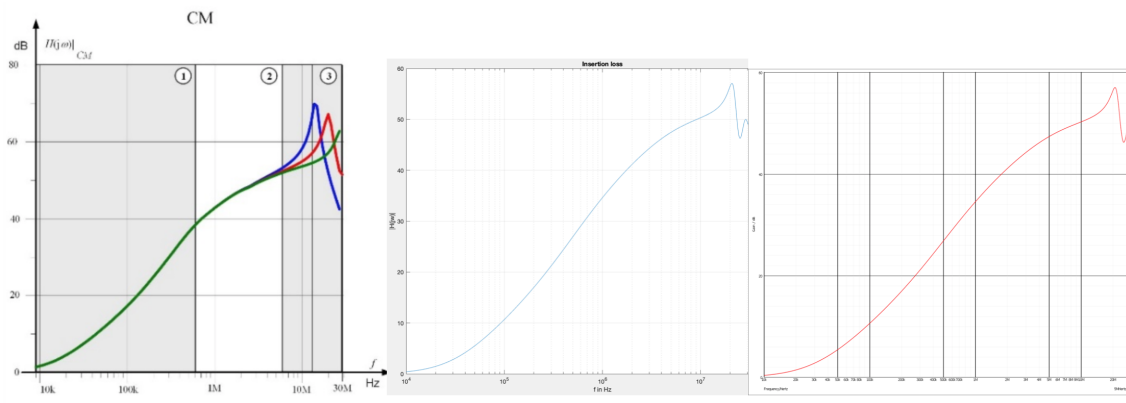


Abbildung 2.13: Vergleich Vorgabe/ Berechnung/ Simulation

3 Softwarekonzept

3.1 Anforderung Software

Die Software soll das Frequenzverhalten und die Einfügungsverluste von CM und DM des EMI-Filters simulieren können. Das Werkzeug soll insbesondere mit einer Empfindlichkeitsanalyse die Auswirkungen der parasitären Parameter auf die Einfügungsverluste des Filters darstellen. Die parasitären Filterparameter können in einem Bereich von $\pm 30\%$ variiert werden. Der Filter wird mit Hilfe von CM- und DM äquivalenten Schaltungsmodellen berechnet. Um die Auswirkungen der Parametervariation besser sichtbar zu machen, wird der Frequenzbereich des Filters in 3 Sektoren aufgeteilt: 0 kHz bis 500 kHz, 500 kHz bis 5 MHz und 5 MHz bis 30 MHz.

Die folgende Konzeptbeschreibung der Software bezieht sich auf die maximal anzustrebende Lösung, dass heisst mit allen Wunschzielen inkludiert. Die Software wird so strukturiert, dass wenn diese Lösung nicht erreicht wird, die Software trotzdem gut zur maximal anzustrebenden Lösung erweiterbar ist. Das Klassendiagramm ist in der Abbildung 3.6 Klassendiagramm ersichtlich.

3.2 GUI

Die GUI wird in 5 Teilbereiche aufgeteilt: Menu, Filtertabelle, CM/DM Plot, Buttonfenster und Eingabefenster. In der Abbildung 3.1 GUI ist die Benutzerfläche des Programms dargestellt.

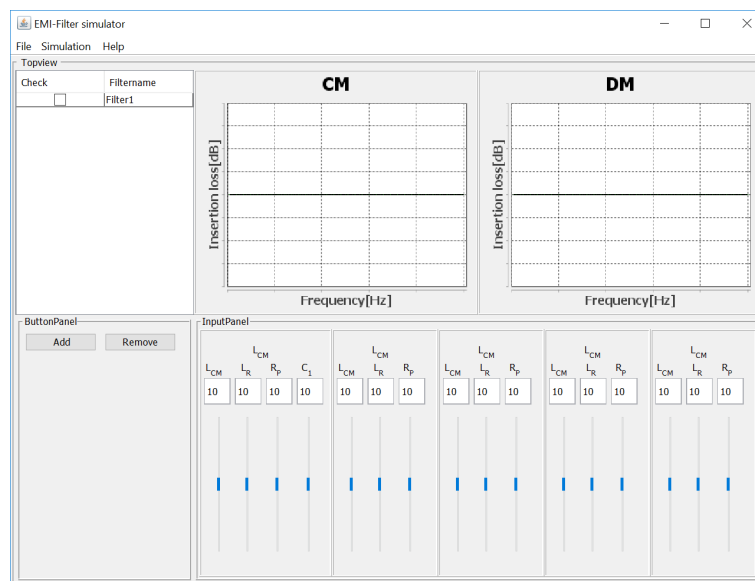


Abbildung 3.1: GUI

3.2.1 Menu

Im Menu können verschiedene Optionen ausgewählt werden. Diese sind ebenfalls alle mit einem Shortcut aufrufbar.

File

Im Menupunkt "File" können Filterprofile gespeichert und geladen werden. Bei beiden Optionen wird der Explorer geöffnet um die .txt Datei im gewählten Verzeichnis abzulegen oder zu holen. In der Option Exit kann das Programm geschlossen werden. Dieser Menupunkt ist in der Abbildung 3.2 Menuoption File dargestellt.

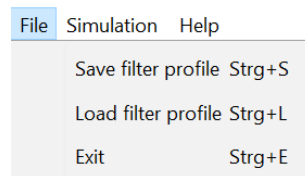


Abbildung 3.2: Menuoption File

Simulation

Im Menupunkt "Simulation" kann die Simulationsart Monte Carlo ausgewählt werden. Es öffnet sich ein neues Fenster in dem der Parameter, die Toleranz und die Anzahl Messungen eingestellt werden kann. Dieser Menupunkt ist in der Abbildung 3.3 dargestellt.

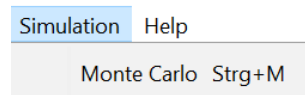


Abbildung 3.3: Menuoption Simulation

Help

Im Menupunkt "Help" können die beiden CM- und DM äquivalenten Schaltungsmodelle, die zur Berechnung verwendet werden, in einem separaten Fenster dargestellt werden. Dieser Menupunkt ist in der Abbildung 3.4 Menuoption Help dargestellt.

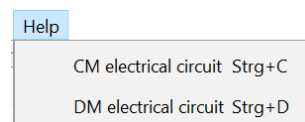


Abbildung 3.4: Menuoption Help

3.2.2 Filtertabelle

In der Filtertabelle werden alle erstellte Filterprofile dargestellt und verwaltet. Mit einer Check-box können einzelne Profile im Plot aus- bzw. eingeblendet werden. Zudem kann bei jedem Filterprofil einen Namen hinzugefügt werden. Die parasitären Filterparameter des ausgewählten Filterprofils werden in das Eingabefenster geladen und können dort verändert werden. Mit den Shortcuts Backspace and Delete können ausgewählte Profile gelöscht werden.

3.2.3 CM/DM Plot

In den CM/DM Plots werden die Berechnungen logarithmisch visualisiert. Die Plots sind für die bessere Darstellung in 3 Frequenzbereiche aufgeteilt: 0 kHz bis 500 kHz, 500 kHz bis 5 MHz und 5 MHz bis 30 MHz. Mit einem Rechtsklick auf den Plot können verschiedene Optionen ausgewählt werden. So können die Eigenschaften (Farbe, Darstellung, Schrift usw.) und der Zoom individuell eingestellt werden. Der Plot kann auch direkt in eine .png Datei abgespeichert oder gedruckt werden. Diese Optionen sind in der Abbildung 3.5 Ploteinstellungen ersichtlich.

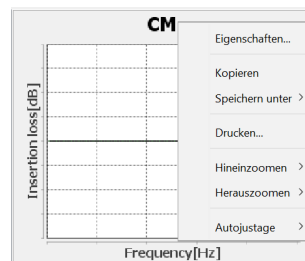


Abbildung 3.5: Ploteinstellungen

3.2.4 Buttonfenster

Im Buttonfenster können Filterprofile in die Filtertabelle geladen oder entfernt werden. Mit dem Button Add werden die eingegebene parasitären Filterparameter in einem neuen Filterprofil gespeichert. Mit dem Button Remove wird das ausgewählte Filterprofil gelöscht. Der Button dient als alternative zu den Shortcuts.

3.2.5 Eingabefenster

Im Eingabebereich befinden sich die einzelnen parasitären Filterparameter. Es können Werte eingegeben und diese mit einer Toleranz von $\pm 30\%$ mit einem Schieberegler variiert werden. Die Werte werden direkt in die Filterprofile geladen, um eine Neuberechnung durchzuführen.

3.3 Softwarestruktur

Die Software wird mit dem Model-View-Controller Entwurfsmuster (MVC Design Pattern) [4] strukturiert. Durch diese Strukturierung ist es weitgehend möglich die Daten und dessen graphischer Repräsentation zu trennen. Dies vereinfacht Wartungsarbeiten und die Wiederverwendbarkeit von Programmteilen. Die Struktur ist in die drei Teilen Modell(engl. model), Präsentation(engl. view) und Steuerung(engl. controller) unterteilt

3.3.1 Model

Das Model enthält Arbeitsdaten eines Programmes und ist für die Berechnungen zuständig. Das Model enthält niemals Referenzen auf die View und den Controller und ist somit von diesen Teilen abgegrenzt.

3.3.2 View

Die View ist für die Darstellung der Programmoberfläche (GUI) und der Berechnungen zuständig. Mit entsprechenden Schnittstellen werden die Benutzereingaben weitergeleitet.

3.3.3 Controller

Der Controller ist die vermittelnde Schicht zwischen der Grafischen Darstellung und den Berechnungen. Der Controller muss somit Zugriff auf die Programmoberfläche und auf die Daten haben.

3.3.4 Klassendiagramm

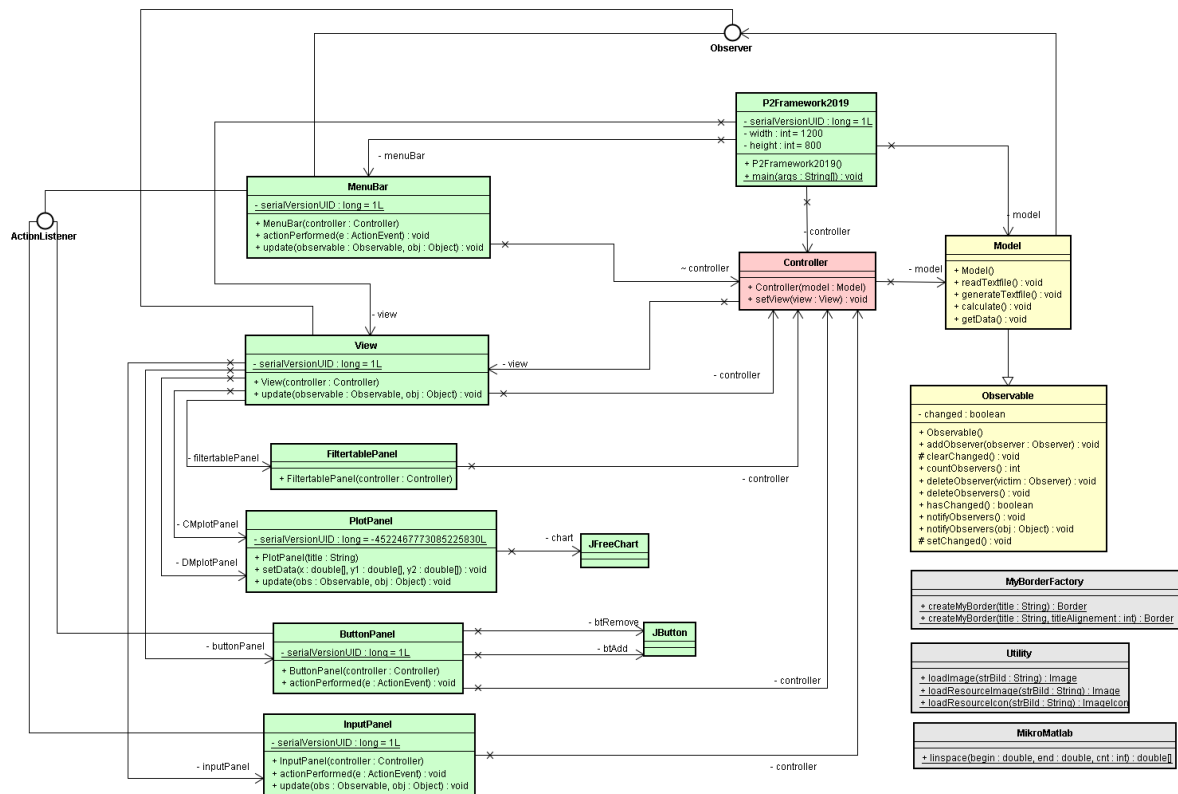


Abbildung 3.6: Klassendiagramm

3.4 Programmablauf

Nach dem Aufstarten des Programmes kann der Benutzer über die Programmoberfläche (3.1 GUI) seine Simulationen starten. Am Anfang ist ein default Filter initialisiert. Dieser ist in der Filtertabelle eingetragen, jedoch sind die parasitären Filterparameter im Eingabefenster noch undefiniert. Der Benutzer kann diese nun definieren und die berechneten Einfügungsverluste von CM und DM des EMI-Filters werden im Plot dargestellt. Mit dem Button Add wird der Filter abgespeichert und ein neuer Filter kann definiert werden. Es können somit mehrere Filter gleichzeitig dargestellt werden.

Um die Filter richtig zu verwalten ist es möglich in der Filtertabelle über die Checkbox einzelne Filter im Plot ein und auszublenden. Ebenfalls können sie spezifisch benannt werden. Der Plot und die einzelnen Kurven können mit einem Rechtsklick auf den CM/DM Plot individuell angepasst und auch exportiert werden. Wird ein Filter nicht mehr benötigt kann er in der Filtertabelle angewählt und mit dem Button Remove entfernt werden. Damit der Benutzer nicht jedesmal die Filterprofile einstellen muss, können diese über File/Save filterprofil in einer .txt Datei abgespeichert werden. Bei einem Neustart des Programmes kann über File/Load filterprofile die Filterprofile wieder geladen werden.

Um die Auswirkungen einzelner parasitären Filterparameter besser zu analysieren, kann unter Simulation/Monte Carlo eine Monte Carlo Simulation gestartet werden. Es wird ein neues Fenster geöffnet in dem der Filterparameter, die Toleranz und die Anzahl Berechnungen eingestellt werden können. Jede Berechnung wird als einzelner Filter in die Filtertabelle geladen. Um nachzuschauen wo welche Filterparameter sich in der Schaltung befindet, können unter Help/CM electrical circuit und CM electrical circuit die Ersatzschaltungen, von der die Berechnungen ausgehen, angeschaut werden. Das Programm wird über File/Exit oder beim Schliessen des Fensters beendet.

3.5 Libraries

In der Software werden folgende Libraries verwendet:

Swing: Mit dem vorinstallierten Swing Framework von Java wird die GUI aufgebaut.

JFreeChart: Die Berechnungen werden mit JFreeChart grafisch also Plot dargestellt. [5]

Apache Math Commons Die Apache Math Commons Library beinhaltet wichtige Mathematikfunktionen, wie rechnen mit Komplexen Zahlen usw. [6]

Engineering Text Fields Die von Prof. Dr. Richard Gut zur Verfügung gestellte Klasse verhindert Fehleingaben und vereinfacht die Eingabe von Zahlen (nano,piko...)

4 Testkonzept

Um das fertige Produkt zu testen werden drei Testläufe unternommen. Das soll garantieren, dass das Programm einwandfrei funktioniert. Dabei wird stets der Testlauf durchgeführt und danach werden die Ergebnisse validiert. Zuerst wird das Programm durch das Projektteam getestet. Im zweiten Testlauf wird das Programm dem Auftraggeber abgegeben. Er testet das Programm und gibt ein Feedback. In der dritten Testphase wird das Programm an Mitstudenten oder in Elektrotechnik versierte Kollegen abgegeben.

4.1 1. Testphase

In der ersten Testphase werden die verschiedenen Teile des Programms unabhängig voneinander durch das Projektteam geprüft. Das GUI wird mit der TraceV5 Methode in Java auf die richtige Programmabfolge getestet. Das Model in dem die Berechnungen stattfinden, wird mit einer Simulation überprüft. Dazu werden die gleichen Parameter wie einer MPLAB Mindi Simulation übergeben und danach eine Auswertung gemacht. Stimmen die erhaltenen Dämpfungskurven überein, kann mit der zweiten Testphase begonnen werden.

4.2 2. Testphase

In der zweiten Testphase wird das fertige Programm an Herrn Dalessandro übergeben. Er kann nach seinem Test allfällige Änderungen vorschlagen, am Konzept wird jedoch nichts mehr geändert. Es geht dabei mehr um kleine Anpassungen in der GUI und um einen Test in der realen Umgebung. Es wird nach dem Test erneut eine Validierung und gegebenenfalls eine Anpassung vorgenommen.

4.3 3. Testphase

Nach der zweiten Testphase wird das Programm an Mitstudenten und Interessierte zum Test abgegeben. Dabei wird das Programm einem Stresstest unterzogen, also möglichst viele Plots sollen berechnet und geöffnet werden. In dieser Testphase geht es vor allem darum das Benutzererlebnis für möglichst viele Anwender zu testen und feedbacks zu sammeln. Nach dieser letzten Testphase werden die Ergebnisse ausgewertet und allfällige Änderungen vorgenommen.

5 Projektvereinbarung

Auftraggeber

Dr. Luca Dalessandro

Ort, Datum

Unterschrift

Projektleiter

Niklaus Schwegler

Ort, Datum

Unterschrift

References

- [1] Luca Dalessandro, „Aufgabenstellung EMI-Filter“, Mar. 21, 2019.
- [2] Stephan Rupp, Andreas Maier, „Hochfrequenztechnik“, Mar. 21, 2019.
- [3] Peter Niklaus, „2-Tor Tabellen“, Mar. 21, 2019.
- [4] javabeginners.de, „MVC Design Pattern“, Mar. 7, 2019.
- [5] Andreas Viklund. (Mar. 17, 2019). JfreeChart, [Online]. Available: <http://www.jfree.org/jfreechart/%EF%BF%BD>.
- [6] The Apache Software Foundation. (Mar. 17, 2019). Commons Math, [Online]. Available: <https://commons.apache.org/proper/commons-math/%EF%BF%BD>.

6 Anhang