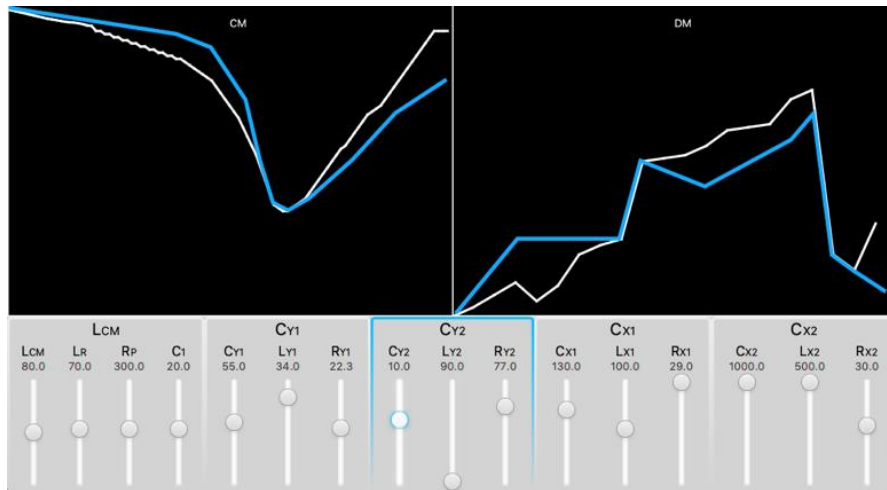


Pflichtenheft

Fachlicher Teil



«DJ» EMI Filter für Netzteil

Pro2E - Team 5

Auftraggeber: Luca Dalessandro

Dozierende: Anita Gertiser
Pascal Buchschacher
Peter Niklaus
Sebastian Gaulocher
Richard Gut

Projektteam: Marina Taborda, Projektleiterin
Michel Alt, Stv. Projektleiter
Frank Imhof
Luca Krummenacher
Richard Britt
Fady Hanna

Windisch, 23.03.2019

Inhaltsverzeichnis

1 Übersicht.....	3
1.1 Ausgangslage	3
1.2 Projektziele.....	4
1.3 Wunschziele.....	4
1.4 Nicht-Ziele	5
1.5 Lieferobjekte.....	5
2 Theoretische Grundlagen	6
2.1 Einleitung EMI Filter	6
2.2 Aufbau EMI Filter.....	6
2.3 Störungsarten.....	6
2.4 Definition Einfügungsverluste «Insertion loss»	7
2.5 Parasitäre Parameter	7
2.6 Vorgehensweise Berechnung Einfügungsverluste.....	8
2.7 S- Parameter	9
3 Softwarekonzept.....	12
3.1 Anforderungen	12
3.2 Software/Hardware.....	12
3.3 Mock-Up.....	12
3.4 Menüleiste (noch nicht realisiert)	12
3.5 Anzeigefenster DM/CM	13
3.6 Haupt- und parasitäre Parameter	13
3.7 Beschreibung Programmablauf	13
4 Softwarestruktur	15
4.1 Klassendiagramm.....	15
5 Testkonzept.....	16
5.1 Kontrolle bei der Entwicklung	16
5.2 Überprüfung mit MATLAB	16
5.3 Integrationstest.....	16
5.4 Softwaretest	16
6 Literaturverzeichnis	18
7 Abbildungsverzeichnis.....	18
8 Tabellenverzeichnis.....	18
9 Anhang.....	19

1 Übersicht

1.1 Ausgangslage

Alle elektronischen Geräte benötigen heute Netzschaltteile, um die Netzspannung in die gewünschte Gerätespannung zu transformieren. Die verwendeten Bauteile in den Schalt-
netzteilen erzeugen dabei leitungsgebundene Störungen.

Damit diese Störungen (Differential Mode und Common Mode) auf der Netzseite keine ne-
gativen Auswirkungen haben, müssen die Netzschaltteile die normativen Anforderungen
erfüllen und dementsprechend dimensioniert werden.

Diese Normen werden von den europäischen Normengremien festgelegt und sind ständig in
Bearbeitung.

HIER MUSS NOCH WAS VON FILTERN STEHN. WIR MACHEN EIN PROGRAMM, DAS
DAS DESIGNNE VON FILTER EINFACHER MACHT. ES ZEIGT DIE EFFEKTE DES FIL-
TERS (INSERTION LOSS) UND NICHT DIE STOERUNGEN. ODER HABE ICH DAS
FALSCH VERSTANDEN? Gruss frank

Das Ziel dieser Arbeit ist die Entwicklung eines bedienungsfreundlichen Programms, das in
der Lage ist, die Störungen zu berechnen sowie graphisch darzustellen.

Dabei kann man das Verhalten der Störung durch die Veränderung der Werte der Bauteile
untersuchen.

1.2 Projektziele

Folgende Anforderungen wurden als Pflichtziele definiert. Diese sind notwendig und müssen beim Endprodukt komplett erfüllt werden.

Nr.	Bereich	Beschreibung
1	GUI-Struktur	<ul style="list-style-type: none"> Das GUI ist schlicht und logisch Programm als MVC Framework GUI soll selbsterklärend sein
2	Art der Dateneingabe der Messdaten	Messwerteingabe über Textfelder und einstellbar über Slider
3	Berechnungen	Berechnungen müssen korrekte Ergebnisse liefern
4	Ausgabe der Messwerte	<ul style="list-style-type: none"> Ausgabe der Resultate mit Graphen 2-dimensionale Darstellung
5	Bedienbarkeit	Soll einfach zu bedienen sein
6	Design	Soll übersichtlich dargestellt werden

Tabelle 1: Projektziele

1.3 Wunschziele

Je nach zeitlichem Verlauf des Projektes können (müssen aber nicht) Zusatzfunktionen implementiert werden. Voraussetzung ist, dass das Projekt keine Verzögerungen erfährt.

Nr.	Bereich	Beschreibung
1	Ausgabedatei	Speichert die Graphen und Parameter in eine Pdf Datei ab.
2	Eingabedatei	Gespeicherte Datei kann wieder mit den eingegebenen Daten wiederhergestellt werden. (Save-Load-Option)
3	Programm	<p>Animierte Funktionen (z.B. vibrierende Taster bei falscher Eingabe, etc.)</p> <p>Eine Funktion, bei der sich der Wert beim Slider selbst immer wieder verändert (Sweeping) und man die Veränderung graphisch sieht.</p>

Tabelle 2: Wunschziele

1.4 Nicht-Ziele

Es gibt auch einige Ziele, welche bewusst nicht zu erreichen sind. Diese würden den Rahmen des Projektes sprengen, da sie zu zeitintensiv sind.

Nr.	Bereich	Beschreibung
1	Darstellung	3-dimensionale Darstellung
2	Eingabefelder	Zusätzliche Eingabefelder für gewünschte Kurve. Dabei werden die optimalen Bauteile dazu berechnet
3	Webapplikation	Tablet-Version und Smartphone Version

Tabelle 3: Nicht-Ziele

1.5 Lieferobjekte

Die Lieferobjekte sind in der unten aufgeführten Tabelle festgelegt. Die Berichte werden per Mail an die Herren Dalessandro, Niklaus, Gaulocher und Buchschacher geschickt.

Datum	Lieferobjekte
28.02.2019	KIS
24.03.2019	Pflichtenheft Version 1
31.03.2019	Statusbericht 1
07.04.2019	Pflichtenheft Endversion
11.04.2019	Zwischenpräsentation
02.05.2019	Disposition, Einleitung und Statusbericht 2
05.05.2019	Statusbericht 2
19.05.2019	Statusbericht 3
13.06.2019	Statusbericht 4
13.06.2019	Fachbericht und Präsentation

Tabelle 4: Lieferobjekte

2 Theoretische Grundlagen

2.1 Einleitung EMI Filter

Nahezu jedes elektrische Gerät besitzt ein Schaltnetzteil um die Netzspannung auf die benötigte Spannung zu regeln. Betrachtet man die Eingangsspannung ohne Netzfilter, wird man auf dem ganzen Frequenzspektrum, d.h. von Netzfrequenz bis zu mehreren MHz Störungen feststellen. Die Aufgabe vom EMI (elektromagnetische Interferenzen) Filter ist es, diese Störungen zu filtern, so dass keine anderen Geräte gestört werden. Damit dieses eingesetzt werden darf, muss jedes Schaltnetzteil sich an bestimmte Normen im Bereich EMV halten.

2.2 Aufbau EMI Filter

Ein EMI Filter für einphasige Geräte besteht nur aus wenigen Bauteilen. Zwei X-Kondensatoren, zwei Y-Kondensatoren, einer Drossel mit zwei Windungen, welche um einen Ferit Ring gewickelt sind und einem Widerstand. Diese Schaltung kann sehr kompakt verbaut werden, was in folgendem Filter von Schaffner sichtbar wird.

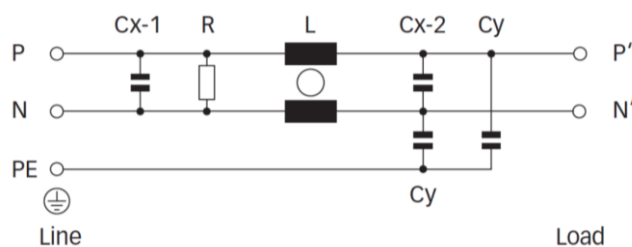


Abbildung 1: Schaltung des FN 2020 Filters (Schaffner), sowie der Filter selbst

Die stromkompensierte Drossel (L) ist in der Lage, Gleichtaktstörungen (CM) zu filtern. Diese Störungen treten gleichzeitig auf beiden Leitungen auf.

Die Y-Kondensatoren, welche gegen Erde geschaltet sind, sind ebenfalls dazu da, um CM-Störungen zu filtern. Diese müssen jedoch eine sehr hohe Überspannungsfestigkeit besitzen, um beispielsweise bei einem Blitzschlag keinen Kurzschluss im Gehäuse zu verursachen.

Störungen zwischen den Zuleitungen, so genannte Gegentaktstörungen (DM), werden mit den X-Kondensatoren gedämpft. [1]

2.3 Störungsarten

Die existierenden EMV Normen gelten für das Gesamtrauschen. Doch in der Praxis wird einfachheitshalber die Gesamtstörung in Gleichtaktrauschen, Common Mode (CM) und Gegentaktrauschen, Differential Mode (DM) gesprochen.

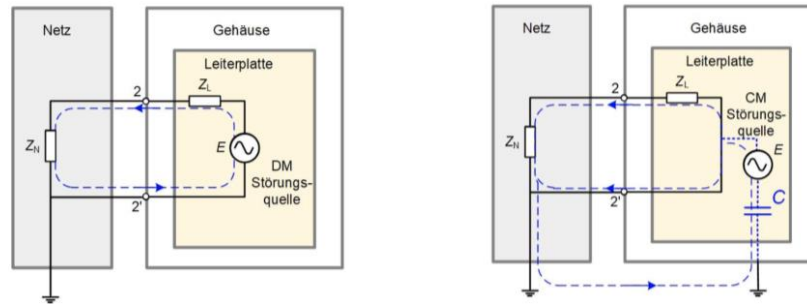


Abbildung 2: Stromzirkulation der Störungen im DM- und CM-Mode

2.4 Definition Einfügungsverluste «Insertion loss»

Die Leistung eines EMI Filters wird mit den Einfügungsverlusten in Abhängigkeit der Frequenz bestimmt. Diese Funktion lautet:

$$|H(j\omega)| = 20 \log \frac{|U_{20}|}{|U_2|}$$

U_{20} : Lastspannung gemessen ohne Filter mit einer Last von 50Ω

U_2 : Lastspannung gemessen mit EMI Filter

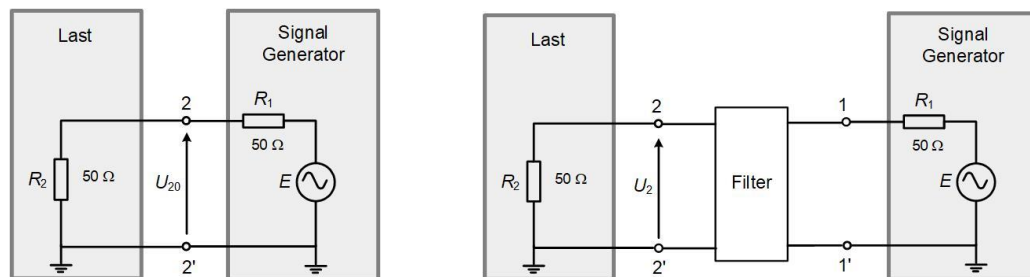


Abbildung 3: Lastspannung ohne Filter und Lastspannung mit EMI Filter

2.5 Parasitäre Parameter

In der Realität verhalten sich die Bauteile eines EMI-Filters leider nicht genau so, wie wir es idealerweise annehmen. Jedes Bauteil hat aufgrund der physikalischen Gegebenheiten eine Ersatzschaltung mit den parasitären Parametern. Diese Ersatzschaltungen sehen wie folgt aus:

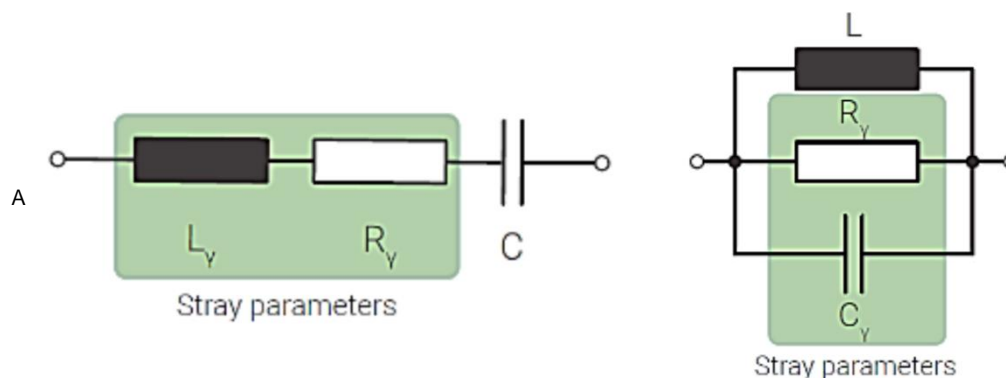


Abbildung 4: Ersatzschaltung für einen realen Kondensator - eine reale Spule

Betrachtet man diese Ersatzschaltbilder, sieht man eigentlich einen Serie- und Parallelschwingkreis. Weit unterhalb der Resonanzfrequenz haben die parasitären Parameter jedoch keinen grossen Einfluss. Im Resonanzfall haben wir aber einen Serie- oder Parallelschwingkreis. Danach wirkt die Spule nicht mehr als Induktivität und der Kondensator nicht mehr als Kapazität. Die Resonanzfrequenzen liegen dabei meist bei mehreren MHz. Doch bei unserer Anwendung wollen wir auch über diesem Frequenzbereich unsere Schaltung berechnen. Der Widerstand hat auch ein Ersatzschema, dieses wurde aber in dieser Schaltung nicht berücksichtigt, da dieses keinen grossen Einfluss auf die Gesamtschaltung hat.

Deshalb hat man für das Grundschemata des Filters, die parasitären Parameter gemäss folgenden Ersatzschemas ergänzt.

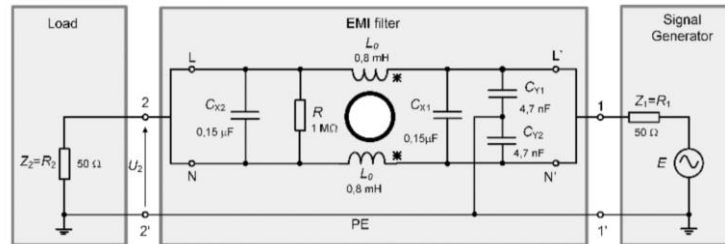


Abbildung 5: CM-Ersatzschaltung ohne parasitäre Parameter

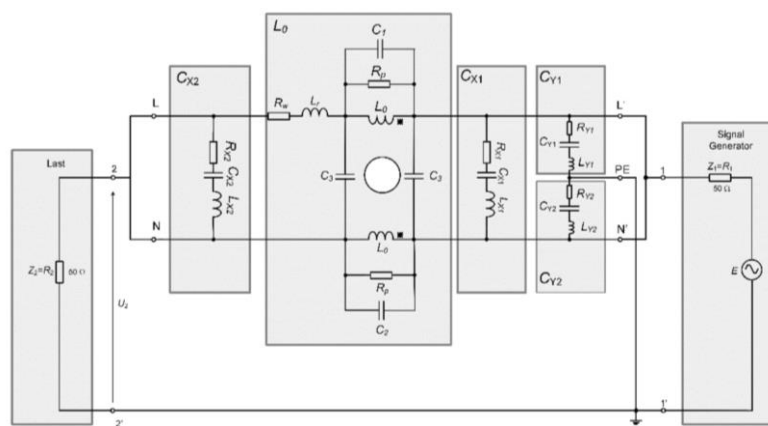


Abbildung 6: CM-Ersatzschaltung mit parasitären Parametern

2.6 Vorgehensweise Berechnung Einfügevverluste

Wir befassen uns zunächst mit der CM-Ersatzschaltung, da sich diese leichter vereinfachen lässt. Man nutzt die Symmetrie der Schaltung aus, indem man bei der CM-Schaltung die beiden Leiter zusammenfasst. Ausserdem lässt man die CX-Kondensatoren weg, da diese bei dieser Schaltung trivial sind. Die reduzierte Schaltung sieht nun schon wesentlich schlanker aus.

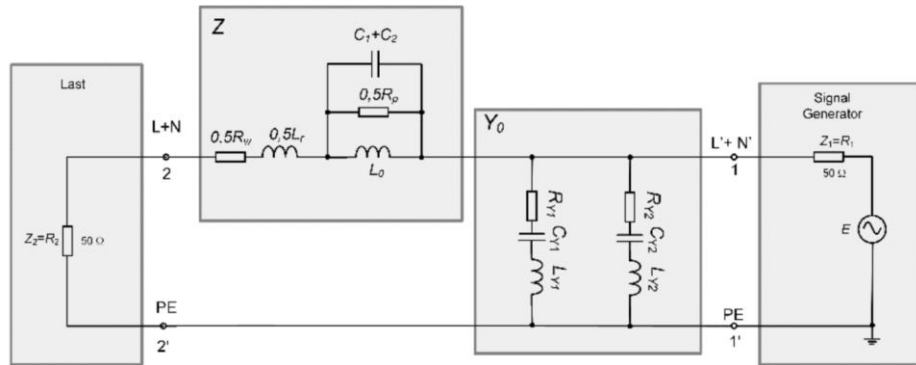


Abbildung 7: Vereinfachte CM-Schaltung

Nun bestimmen wir die Gesamtimpedanz (Z_1), sowie die Gesamtimpedanz (Z_2).

$$Z_1 = 0.5R_W + 0.5L_T j\omega + \left(\frac{1}{\frac{1}{0.5R_P + L_0 j\omega + \frac{1}{(C_1 + C_2)j\omega}}} \right)$$

$$Z_2 = \frac{1}{\frac{1}{\left(R_{Y1} + L_{Y1} j\omega + \frac{1}{C_{Y1} j\omega} \right)} + \frac{1}{\left(R_{Y2} + L_{Y2} j\omega + \frac{1}{C_{Y2} j\omega} \right)}}$$

Wobei $\omega = 2\pi f$ ist und wir später die Impedanzen für ein Frequenzspektrum von 50Hz bis 30MHz berechnen werden.

Wir berechnen nun die Kettenmatrix (A_1) für die Längsimpedanz (Z_1) sowie (A_2) für die Querimpedanz (Z_2).

$$A_1 = \begin{bmatrix} 1 & Z_1 \\ 0 & 1 \end{bmatrix}$$

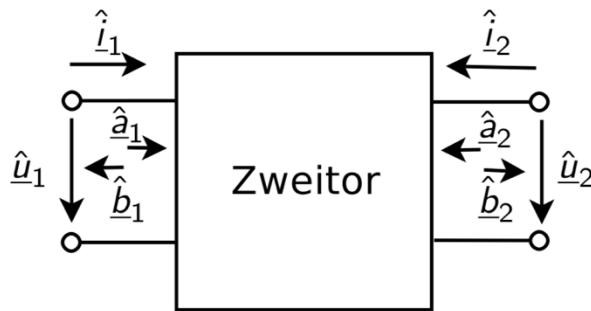
$$A_2 = \begin{bmatrix} 1 & 0 \\ 1/Z_2 & 1 \end{bmatrix}$$

Für den S21-Parameter benötigen wir die Gesamtmatrix, diese berechnet man mit Hilfe der Kettenschaltung, in dem man die beiden Matrizen miteinander multipliziert. [2]

$$A = A_1 * A_2 = \begin{bmatrix} 1 & Z_1 \\ 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ 1/Z_2 & 1 \end{bmatrix} = \begin{bmatrix} Z_1/Z_2 + 1 & Z_1 \\ 1/Z_2 & 1 \end{bmatrix}$$

2.7 S- Parameter

Sobald man die Kettenmatrix für ein Zwei-Tor berechnet hat, lassen sich daraus die S-Parameter berechnen. Deshalb ist es der einfachste Weg, die Einfügungsverluste mit Hilfe dieser Parameter zu bestimmen. Im folgenden Zwei Tor sind die Wellengrößen zur Berechnung der S-Parameter angegeben.



Wobei \hat{a}_1 und \hat{a}_2 die Amplituden der einflussenden Wellen sind.

Die Grössen \hat{b}_1 und \hat{b}_2 sind hingegen die Amplituden der reflektierten Wellen.

Abbildung 8: Zwei Tor mit Wellengrössen

Daraus lässt sich die Streumatrix S wie folgt berechnen:

$$\begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{bmatrix}$$

Die einzelnen Parameter haben dabei folgende Bedeutung [3]:

S_{11} : Eingangsreflexionsfaktor

S_{12} : Rückwärtstransmissionsfaktor

S_{21} : Vorwärtstransmissionsfaktor

S_{22} : Ausgangsreflexionsfaktor

Wir benötigen den Rückwärtstransmissionsfaktor. Dieser benötigt die Bezugswiderstände R_w , welche bei uns am Ein- sowie am Ausgang 50Ω betragen. Daraus berechnen wir den Streuparameter folgendermassen:

$$S_{21} = \frac{2}{A_{11} + \frac{A_{12}}{R_w} + A_{21}R_w + A_{22}} = \frac{2}{\frac{Z_1}{R_w} + \frac{R_w}{Z_2} + \frac{Z_1}{Z_2} + 2}$$

Nun können wir die Einfügungsdämpfung a in dB berechnen:

$$a_I = -20 \log |S_{21}| \text{ [dB]}$$

Diese Berechnungen haben wir in MATLAB realisiert und für verschiedene Frequenzen die Dämpfungen berechnet, dabei haben wir folgende Kurven bekommen. Der Code für die Berechnungen befindet sich im Anhang. [4] [5]

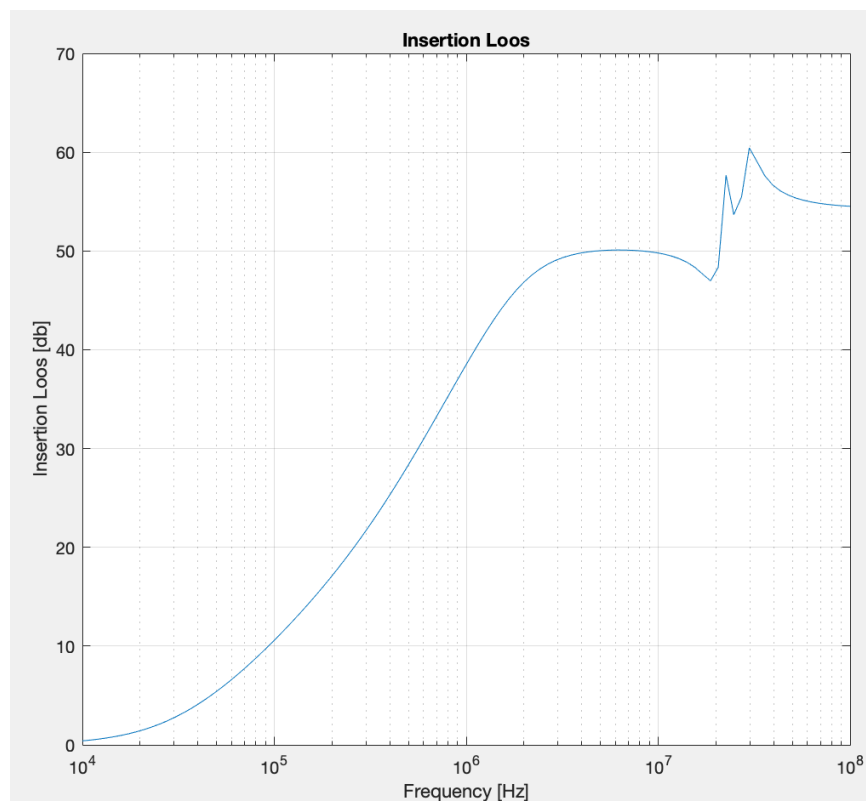


Abbildung 9: Insertion loss Kurve aus MATLAB

3 Softwarekonzept

3.1 Anforderungen

Es soll ein bedienungsfreundliches Programm entwickelt und realisiert werden, mit dem das Frequenzverhalten und die Einfügungsverluste von CM und DM des EMI-Filters vorhergesagt werden können. Die Auswirkungen, die die parasitären Parameter auf die Einfügungsverluste des Filters haben, müssen im Programm eingesehen werden können. Das Programm soll die beiden Filtervarianten hinsichtlich der Leistung untersuchen und die bessere empfehlen. Optional ist eine Untersuchung der Filtervarianten auf Fussabdruck (CO₂? Material? Graue Energie?) und Kosten.

Es ist erwünscht, dass das Programm eine Bedienoberfläche hat, welche einem Mischpult mit Schiebereglern ähnlich sieht.

3.2 Software/Hardware

Das Programm wird in der Sprache Java als Desktop-Applikation für die Betriebssysteme MacOS und Windows entwickelt. Um den Inhalt möglichst von der Darstellung trennen zu können, wird die Library JavaFX verwendet.

3.3 Mock-Up

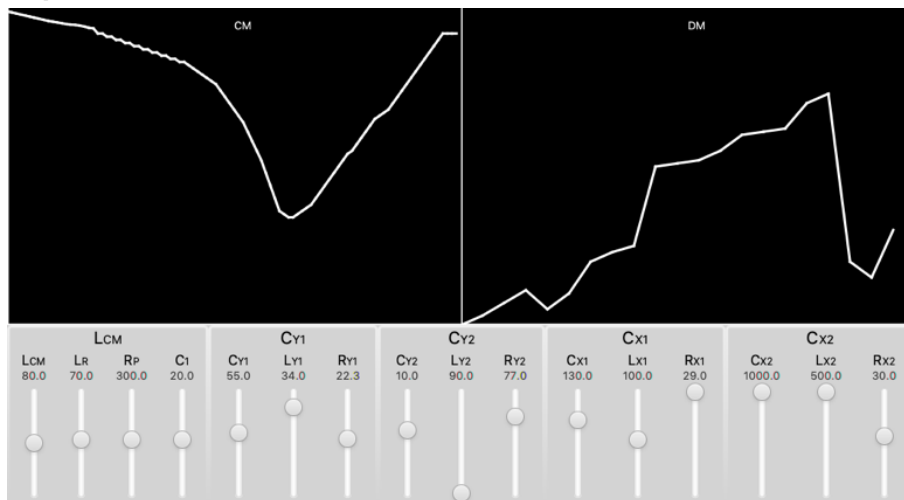


Abbildung 10: GUI Übersicht

Die grafische Bedienoberfläche besteht voraussichtlich aus einer Menüleiste, einem Anzeigefenster für jeweils DM und CM, Haupt- und parasitär Parameter. Diese Bestandteile werden in den folgenden Unterkapitel genauer beschrieben.

3.4 Menüleiste (noch nicht realisiert)

Von hier aus haben Nutzende Zugriff auf verschiedene gröbere Einstellungen und Funktionen. Per Mausklick sollen Daten gespeichert, geladen oder ausgedruckt werden können.

3.5 Anzeigefenster DM/CM

Hier werden die CM- und DM-Einfügungsverluste als voneinander getrennte Kurvendiagramme dargestellt. Eine logarithmische Skalierung der Frequenzachse ermöglicht die Darstellung der Verluste innerhalb eines Frequenzbereichs von 1Hz bis 30MHz. Pro Diagramm können auch mehrere Kurven gezeichnet werden.

3.6 Haupt- und parasitäre Parameter

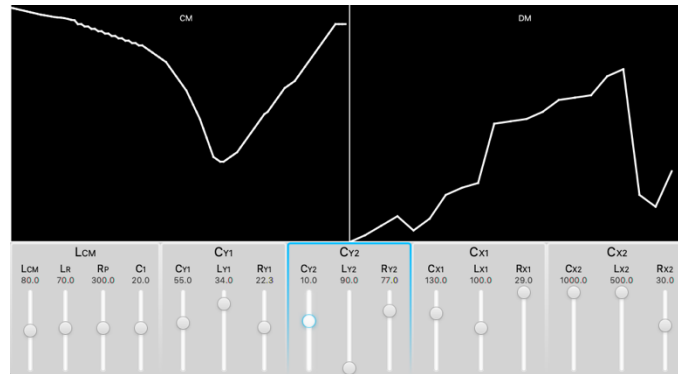


Abbildung 11: Parametereinstellungen

Jeder Hauptparameter wird durch eine gewisse Anzahl parasitärer Parameter dargestellt. Letztere können einzeln via Schieberegler um maximal $\pm 30\%$ von ihrem ursprünglichen Wert verschoben werden. Jede Veränderung eines Schiebereglers bewirkt eine erneute Darstellung der Kurvendiagramme. Oberhalb der Regler werden die momentanen Werte der parasitären Parameter aufgezeigt.

3.7 Beschreibung Programmablauf

Beim Starten des Programmes werden bestenfalls die Daten einer zuvor beendeten Session wieder geladen. Nach erfolgreichem Laden erfolgt automatisch das Zeichnen der Kurve und das Setzen der Schieberegler auf ihren Wert. Nutzende können auch via Menüleiste gespeicherte Sessions laden oder mit einer neuen beginnen.

Durch Verändern der parasitären Parameter wird das Neuzeichnen der Kurven ausgelöst. Durch Verwendung verschiedener Farben in den Kurvendiagrammen könnten optimale Einstellungen hervorgehoben und von suboptimalen unterschieden werden.

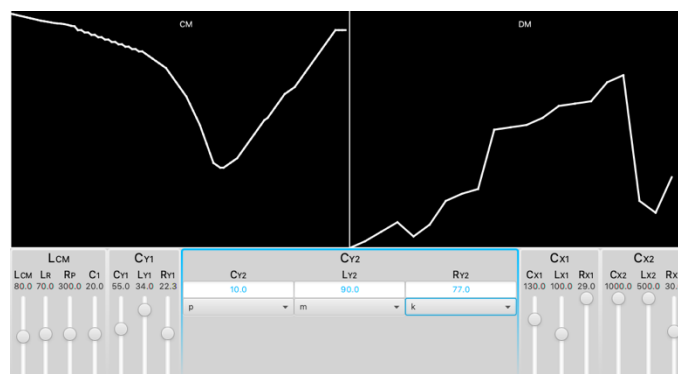


Abbildung 12: Anpassung der Parameter

Bei Doppelklick auf einen Parameter wechselt die Darstellung in seinem Fenster. Nun können die ursprünglichen Werte der parasitären Parameter via Textfelder und anderen Kontrol-

Einheiten geändert werden. Bei erneutem Doppelklick geht die Ansicht wieder zurück zu den Schiebereglern.

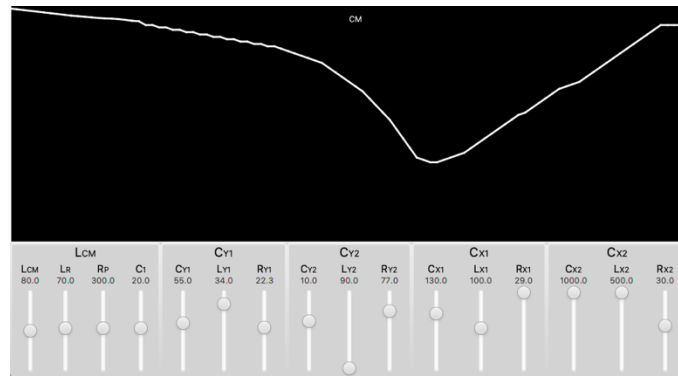


Abbildung 13: Kurvenansicht

Bei Doppelklick auf ein Diagramm, übernimmt dieses die ganze Breite des Fensters und kann so genauer angesehen werden. Bei erneutem Doppelklick wird der Vorgang rückgängig gemacht, so dass wieder beide Diagramme sichtbar sind.

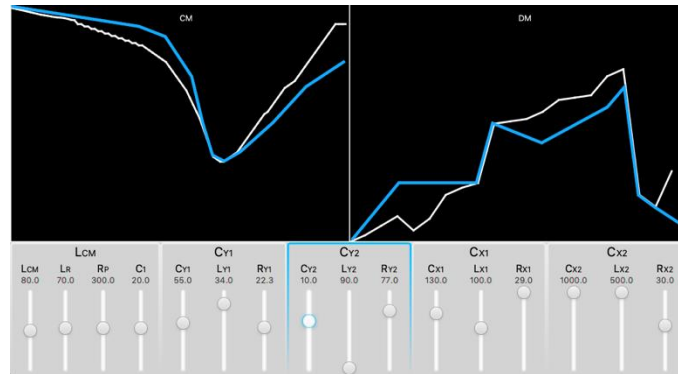


Abbildung 14: Parameterauswirkung

Wenn Nutzende den Mauszeiger über einen Schieberegler bewegen, soll im Diagramm gezeigt werden, welche Auswirkungen dieser Schieberegler bei voller Auslenkung ($\pm 30\%$) auf die Kurve hätte. Somit können die unterschiedlichen Auswirkungen aller parasitären Parameter schnell eingesehen werden, ohne dass die Nutzenden die Schieberegler einzeln verschieben müssen.

4 Softwarestruktur

4.1 Klassendiagramm

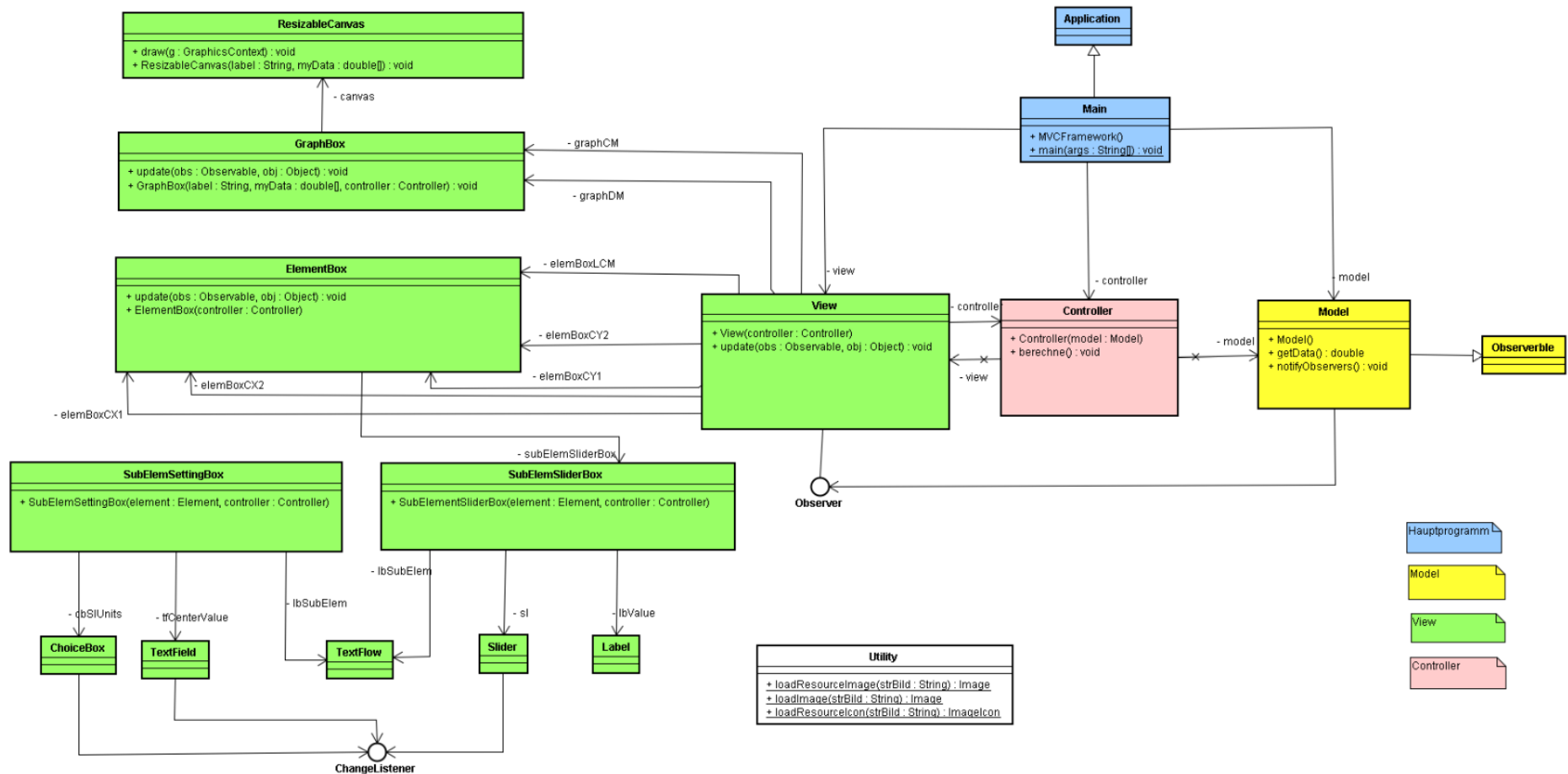


Abbildung 15: Klassendiagramm

5 Testkonzept

Das Testkonzept beschreibt Methoden, welche für die Überprüfung des Programms verwendet werden. Das Programm soll am Ende alle Anforderungen des Auftraggebers erfüllen und einwandfrei funktionieren. Ebenfalls sollte das Programm benutzerfreundlich und selbsterklärend sein. Damit alle Forderungen erfüllt werden können, müssen alle Teilbereiche des Produkts fortgehend kontrolliert und getestet werden.

5.1 Kontrolle bei der Entwicklung

Damit keine schwerwiegenden Fehler auftreten, wird das Programm während der Entwicklung schon getestet. Die erkannten Fehler sollen dabei korrigiert und protokolliert werden. Dies hat den Vorteil, dass keine grossen Zeitverluste durch Fehlerbehebungen entstehen.

5.2 Überprüfung mit MATLAB

Die Richtigkeit der Berechnungen und der Rückgabewerte der Software werden mit MATLAB überprüft. Dabei müssen die Werte in allen Bereichen übereinstimmen. Ausserdem werden die Kurven von MATLAB mit Simulationen der Filterhersteller verglichen.

5.3 Integrationstest

Die verschiedenen Klassen und Methoden werden isoliert geprüft. Damit kann sichergestellt werden, dass alle Klassen/Methoden die vorgesehenen Rückgabewerte und Aktionen korrekt ausführen. Soweit die Prüfungen mit den erwarteten Werten übereinstimmen, werden die Klassen/Methoden in das Programm integriert.

5.4 Softwaretest

Die Software wird auf verschiedene Weisen und Ebenen getestet. Das Programm sollte allen Anforderungen des Auftraggebers gerecht werden und die im Auftrag beschriebenen Anforderungen erfüllen. Dafür kontrollieren die Teammitglieder, ob die Benutzeroberfläche komplett ist und das Programm so funktioniert wie erwünscht. Falls auffällt, dass etwas fehlt oder nicht in Ordnung ist, wird dies dann beseitigt.

Durch regelmässige Absprache mit dem Auftraggeber, soll verhindert werden, dass der Auftraggeber mit der Benutzeroberfläche unzufrieden ist. Der regelmässige Austausch soll dafür sorgen, dass das Programm mit den Vorstellungen des Auftraggebers übereinstimmt.

Fehlerhafte Eingaben sollen vom Programm erkannt und bewältigt werden. Dafür wird durch absichtliche Falscheingaben vom Projektteam überprüft, ob das Programm richtig reagiert. Sollte dies nicht der Fall sein, müssen die Probleme schnellstmöglich behoben werden.

Für die Prüfung der Benutzerfreundlichkeit wird eine Prüfung durch Dritte gemacht. Das Projektteam kann anhand der Feedbacks Dritter nötige Anpassungen machen. So können eventuelle Verbesserungsmöglichkeiten bezüglich der Handhabung gefunden und das Programm verbessert werden.

Damit das Programm auf den Betriebssystemen MacOS und Windows einwandfrei funktioniert, wird innerhalb des Projektteams das Programm auf beiden Systemen getestet. Dies soll verhindern, dass das Programm nur auf einem Betriebssystem laufen kann.

6 Literaturverzeichnis

- [1] Schaffner, «Single-Stage Filters FN 2020,» Schaffner Group, Luterbach, 2018.
- [2] P. Niklaus, «Zweitore (2-Tore),» FHNW, Windisch, 2018.
- [3] A. Schaum, «Zweitore - Zusammenfassung,» Lehrstuhl für Regelungstechnik, Universität zu Kiel, 2017.
- [4] A. Maier und S. Rupp, «Hochfrequenztechnik, Teil 2 - Anwendungen,» DHBW, Stuttgart, 2016.
- [5] B. Mößlang, «EMC Filter Insertion Loss Simulation,» OMICRON Lab, Reutlingen, 2016.

7 Abbildungsverzeichnis

Abbildung 1: Schaltung des FN 2020 Filters (Schaffner), sowie der Filter selbst	6
Abbildung 2: Stromzirkulation der Störungen im DM- und CM-Mode	7
Abbildung 3: Lastspannung ohne Filter und Lastspannung mit EMI Filter	7
Abbildung 4: Ersatzschaltung für einen realen Kondensator - eine reale Spule	7
Abbildung 5: CM-Ersatzschaltung ohne parasitäre Parameter	8
Abbildung 6: CM-Ersatzschaltung mit parasitären Parametern	8
Abbildung 7: Vereinfachte CM-Schaltung	9
Abbildung 8: Zwei Tor mit Wellengrößen	10
Abbildung 9: Insertion loss Kurve aus MATLAB	11
Abbildung 10: GUI Übersicht	12
Abbildung 11: Parametereinstellungen	13
Abbildung 12: Anpassung der Parameter	13
Abbildung 13: Kurvenansicht	14
Abbildung 14: Parameterauswirkung	14
Abbildung 15: Klassendiagramm	15

8 Tabellenverzeichnis

Tabelle 1: Projektziele	4
Tabelle 2: Wunschziele	4
Tabelle 3: Nicht-Ziele	5
Tabelle 4: Lieferobjekte	5

9 Anhang

Diese MATLAB-Datei berechnet für einen Frequenzvektor (1kHz – 10GHz) die Einfügungsverluste und stellt diese logarithmisch dar.

```
%*****
% CM Mode
% Luca Krummenacher
%*****

f = logspace(4,8,100); %Frequenzvektor von 10^4 - 10^8 Hz
for k=1 : length(f)
    H(k) = -20 * log10(abs(get_s21(f(1,k))));
end

semilogx(f,H) %Logarithmische Skala verwenden

grid on

title('Insertion Loos')
xlabel('Frequency [Hz]')
ylabel('Insertion Loos [db]')
```

Die Funktion get_s21 gibt für eine bestimmte Frequenz den S21 Parameter zurück. Die Bauteilparameter sind in dieser Funktion bestimmt.

```
function [result] = get_s21(frequency)
%*****
% Projekt: s21 Parameter berechnen
% fuer die CM-Schaltung eines EMI-Filters
% Autor: Luca Krummenacher
% Datum: 13. Maerz
% Ort : FHNW Brugg-Windisch
% Version: 1.0
%*****
% Gibt den s21 parameter fuer eine bestimmte Frequenz zurück
% Die Bauteilparameter sind in dieser Funktion bestimmt
%*****
Rref = 50; %50 Ohm Eingangs und Lastwiderstand (Bezugswiderstand)
Rw = 11.3e-3;
Lr = 9.2e-6;
C1C2 = 12.93e-12;
Rp = 7.92e3;
L0 = 0.5e-3;
f = frequency;
w = 2*pi*f;

ZZtot = 0.5*Rw + 0.5*Lr*w*j + rpara(L0*w*j,rpara(0.5*Rp,(1/(j*w*C1C2))));
%ZY aus vereinfachter Schaltung:
Ry1 = 246e-3;
Cy1 = 3.85e-9;
Ly1 = 8e-9;
Ry2 = 100.1e-3;
Cy2 = 4.67e-9;
Ly2 = 10.1e-9;

ZY1 = Ry1 + Ly1*w*j + (1/(j*w*Cy1));
ZY2 = Ry2 + Ly2*w*j + (1/(j*w*Cy2));
ZYtot = rpara(ZY1,ZY2);

%A Matrizen aus Impedanzen berechnen:
%Laengsimpedanz A1:
A1 = [1 ZZtot; 0 1];
```

```
%Querimpedanz A2:  
A2 = [1 0; (1/ZYtot) 1];  
  
%Gesamtmatrix berechnen  
A = A1 * A2;  
  
%s21 berechnen  
result = 2 / ( A(1,1) + (A(1,2)/Rref) + (A(2,1)*Rref) + A(2,2) );  
  
end
```