

«DJ» EMI Filter für Schaltnetzteil

Pflichtenheft technischer Teil

Windisch, 04.04.2019



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
Alfare Claudio
Simon Rohrer

Version 1.0

Inhaltsverzeichnis

1	Übersicht	1
1.1	Ausgangslage	1
1.2	Projektziele	1
1.3	Lieferobjekt	1
1.4	Rahmenbedingungen	1
2	Theoretische Grundlagen	2
2.1	EMI-Filter	2
2.2	Schaltungen	2
2.3	Vorgehen	3
2.4	Beispiel	4
3	Softwarekonzept	6
3.1	Anforderung Software	6
3.2	GUI	6
3.2.1	Menu	7
3.2.2	Filtertabelle	8
3.2.3	CM/DM Plot	8
3.2.4	Buttonfenster	8
3.2.5	Eingabefenster	8
3.2.6	Statusleiste	8
3.3	Softwarestruktur	9
3.3.1	Model	9
3.3.2	View	9
3.3.3	Controller	9
3.3.4	Klassendiagramm	9
3.4	Programmablauf	9
4	Testkonzept	10
4.1	1	10
4.2	2	10
4.3	3	10
5	Projektvereinbarung	11
6	Anhang	12

1 Übersicht

1.1 Ausgangslage

Schaltnetzteile von elektrischen Geräten verursachen Leitungsgebundene Störungen. Deshalb schreiben Normen, wie beispielsweise die EN 55022 für IT-Ausrüstung, Grenzen für solche Störungen vor. Diese müssen deshalb mit einem Filter gedämpft werden. Sie werden unterteilt in Schaltfrequenzen, Oberschwingungen, und EMC / EMI Emissionen. EMC / EMI Emissionen sind in einem Frequenzbereich von 150 kHz – 30 MHz. Es gibt zwei Arten von Störungen oder Rauschen. Das Gleichtaktrauschen, oder common mode und das Gegentaktrauschen, differential mode. Beim common mode treten die Störspannungen zwischen Netzwerkleitern und Bezugsmasse auf und die Störströme fließen in Richtung der Netzwerksleiter. Beim differential mode tritt die Störspannung zwischen den Versorgungsleitungen auf und die Störströme fließen in Richtung der Netzwerkströme. Schaffner hat uns nun den Auftrag erteilt ein Simulationsprogramm für Netzwerkfilter zu entwickeln. Die Anforderungen an das Programm sind, dass die Dämpfungseigenschaften des Filters und die Einfügungsverluste ermittelt werden können, sowohl für common mode als auch für differential mode. Die Parameter der parasitären Einflüsse können um ± 30 Das Programm soll einen analytischen Ansatz verfolgen oder mit einer Simulation gelöst werden.

1.2 Projektziele

1.3 Lieferobjekt

1.4 Rahmenbedingungen

2 Theoretische Grundlagen

2.1 EMI-Filter

Das vorgegebene EMI-Filtes muss bezüglich der Einfügungsverluste (insertion loss) untersucht werden. Die Einfügungsverluste hängen vom Gesamttrauschen der Schaltung ab. Es wird ein Ansatz verwendet der in der Praxis weit verbreitet ist, bei welchem das Gesamttrauschen 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 werden die Einfügungsverluste in Funktion der Frequenz berechnet. Die Berechnungen decken einen Bereich von 0 bis 30MHz ab.

Die Einfügungsverluste sind wie folgt definiert:

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

Abbildung 2.1: insertionloss

Wobei H dem Streuparameter (S-Parameter) S index 21 entspricht. Die Einfügungsverluste sind auch mit dem Verhältnis von eingehenden zu abegegebenen Leistung zu Berechnen, jedoch eignet sich diese Methode mehr beim messtechnischen bestimmen der Einfügungsverluste. Da die Berechnungen in einem Bereich von bis zu 30 MHz gemacht werden, ist es notwendig die parasitären Parameter von Spule und Kondensator miteinzubeziehen. Zudem können sie in einem Bereich von +- 30

2.2 Schaltungen

CM:

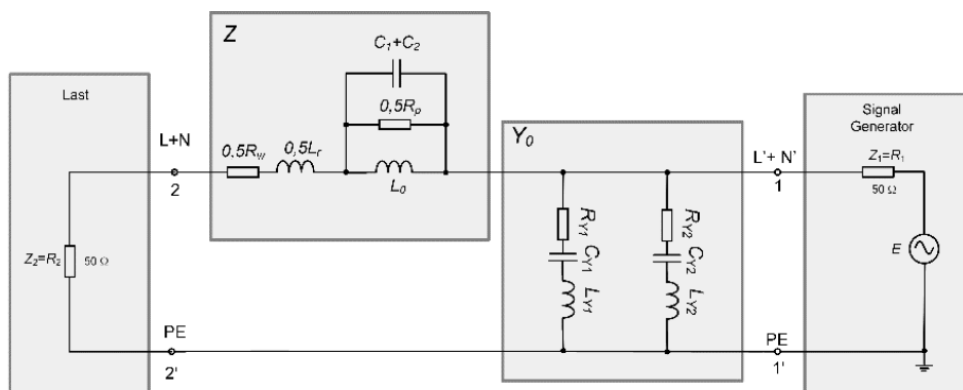


Abbildung 2.2: CM-Schaltungäquivalent

DM:

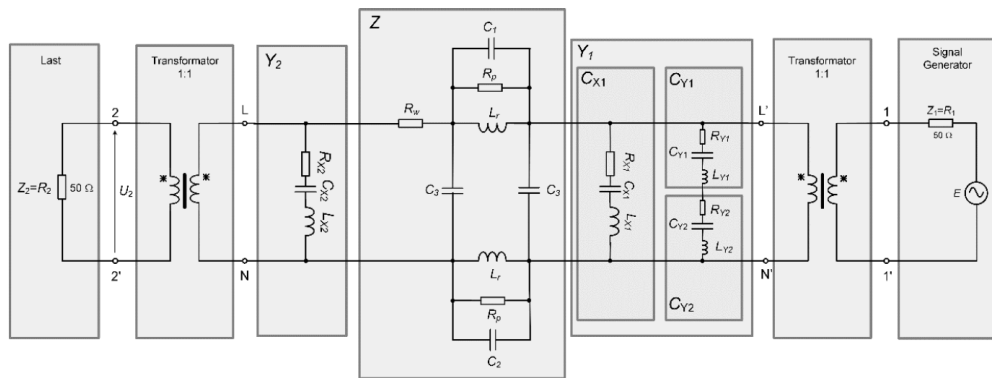


Abbildung 2.3: DM-Schaltungsäquivalent

2.3 Vorgehen

Die Einfügungsverluste werden analytisch ermittelt. Im ersten Schritt werden die Berechnungen in MATLAB gemacht. Somit können die Funktionen einfach 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 Einfügungsverluste bestimmen zu können, wird das Model der 2-Tore verwendet. Einzelne Schaltungsteile werden in ABCD-Matrixen abgebildet, welche dann durch Kaskadierung der einzelnen ABCD-Matrixen zusammengeführt werden. Die Einfügungsverluste werden den S-Parameter abgeleitet, welche direkt aus der ABCD-Matrix errechnet werden kann. Der S-Parameter Index 12 gibt den Tranmissionsgrad 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 Beispiel

$$IL = -20 * \log(|S_{21}|) \quad (2.1)$$

Bei der Definition der Einfügungsverluste kann für das Verhältnis der Spannungen der Streuparameter S21 eingesetzt werden. Der Streuparameter wird wie folgt berechnet. Schritt1: Berechnen der Längsimpedanz Z und der Queradmittanz Y0. Für die Querimpedanz Y0 ergibt sich die Formel

$$Y_0 = \frac{1}{R_{y1} + \frac{1}{j * \omega * C_{y1}} + j * \omega * L_{y1}} + \frac{1}{R_{y2} + \frac{1}{j * \omega * C_{y2}} + j * \omega * L_{y2}} \quad (2.2)$$

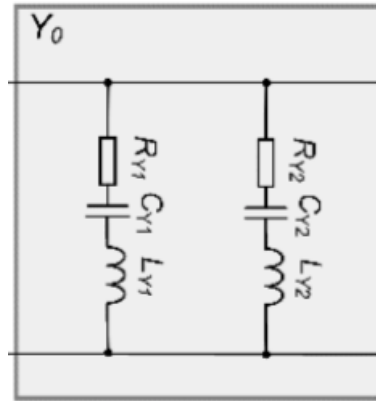


Abbildung 2.4: CM-Admittanz

Für die Impedanz Z ergibt sich folgende Formel

$$Z = 0.5 * R_w + j * \omega * L_r + \frac{1}{\frac{1}{0.5 * R_p} + j * \omega * L_r * (C_1 + C_2) + \frac{1}{j * \omega * L_0}} \quad (2.3)$$

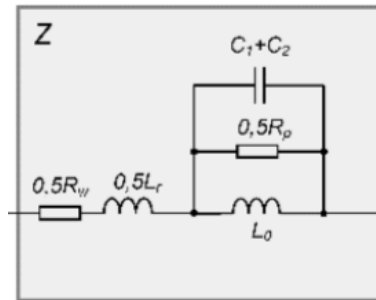


Abbildung 2.5: CM-Impedanz

Schritt2: Erstellen der ABCD-Matrixen.

Somit ergeben sich die ABCD-Matrixen wie folgt

$$A_1 = \begin{pmatrix} 1 & 0 \\ Y & 1 \end{pmatrix} \quad (2.4)$$

$$A_2 = \begin{pmatrix} 1 & Z \\ 0 & 1 \end{pmatrix} \quad (2.5)$$

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

$$A = A_1 * A_2 \quad (2.6)$$

Schritt3: S21 Parameter bilden.

Der S21 Parameter ist wie folgt definiert.

$$S_{21} = \frac{2}{A_{11} + \frac{A_{12}}{R_w} + A_{21} * R_w + A_{22}} \quad (2.7)$$

Somit sind alle gesuchten Werte gegeben und der S21 Parameter wird durch einsetzen der Werte gebildet.

Schritt4: Einfügungsverluste bilden

Durch Einsetzen des Streuparameters S21 in die Definition der Einfügungsverluste, lassen sich diese darstellen. Folgende Grafik zeigt die Berechnungen in MATLAB.

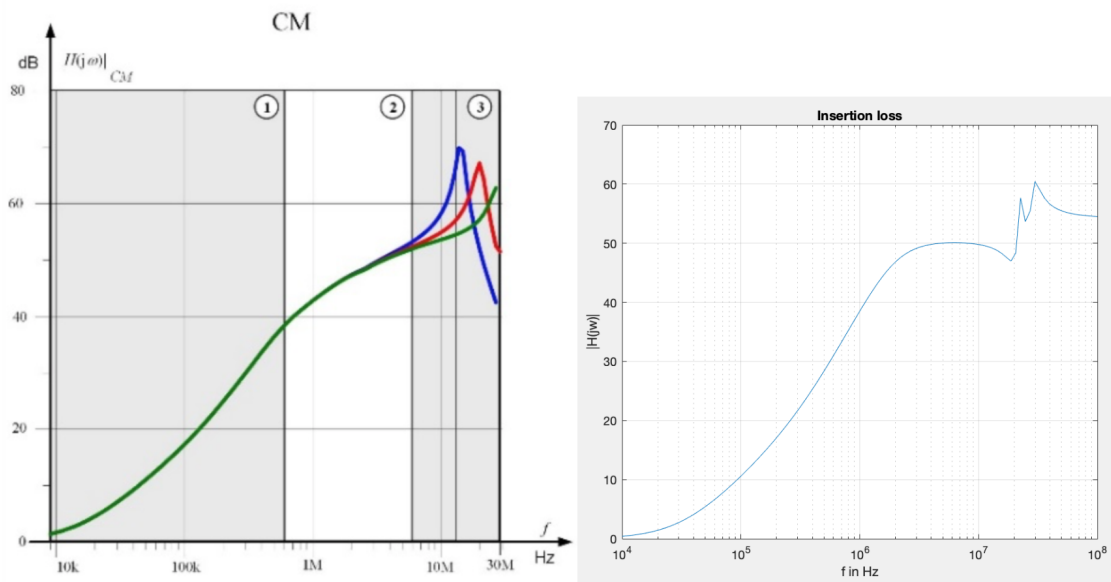


Abbildung 2.6: Vergleich

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 Schaltungsmodelle 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 gut zur maximal anzustrebende Lösung erweiterbar ist.

3.2 GUI

Die GUI wird in 6 Teilbereiche aufgeteilt: Menu, Filtertabelle, CM/DM Plot, Buttonfenster, Eingabefenster und Statusleiste. In der Abbildung 3.1 GUI ist die Benutzerfläche des Programms dargestellt.

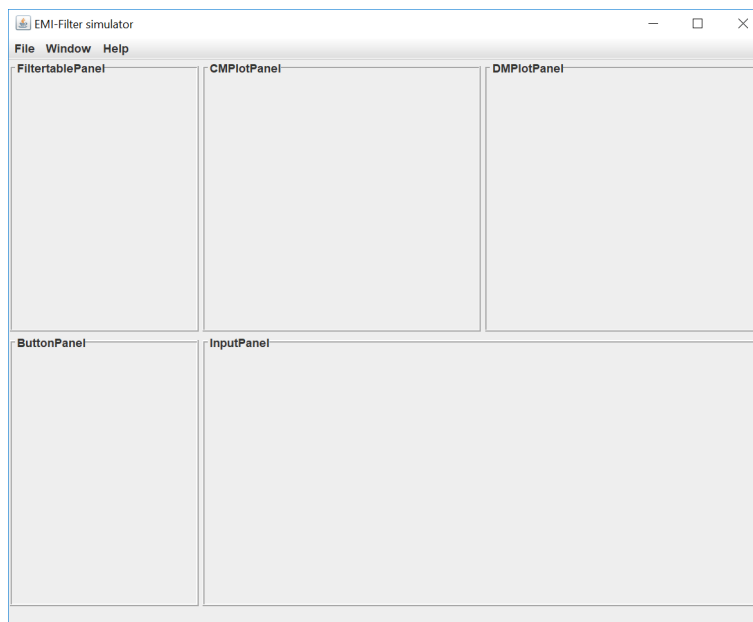


Abbildung 3.1: GUI

3.2.1 Menu

Im Menu können verschiedene Optionen ausgewählt werden.

File

Im Menüpunkt "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 Menüpunkt ist in der Abbildung Abbildung 3.2 Menuoption File ersichtlich.



Abbildung 3.2: Menuoption File

Window

Im Menüpunkt "Window" können die beiden Plots der CM und DM Berechnungen zur besseren Analyse in einem separaten Fenster geöffnet werden. Dieser Menüpunkt ist in der Abbildung Abbildung 3.3 Menuoption Window ersichtlich.

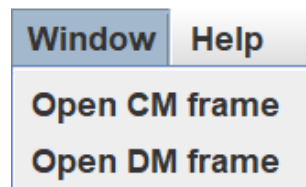


Abbildung 3.3: Menuoption Window

Help

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



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 und eine Plotfarbe hinzugefügt werden. Die parasitären Filterparameter des ausgewählten Filterprofils werden in das Eingabefenster geladen und können dort verändert werden. Mit den Shortcuts Backspace und 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.

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.

3.2.6 Statusleiste

In der Statusleiste werden Informationen und Fehlermeldungen ausgegeben.

3.3 Softwarestruktur

Die Software wird mit dem Model-View-Controller Entwurfsmuster (MVC Design Pattern) **MVCDesignPattern** 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 Teile Modell(engl. model), Präsentation(engl. view) und Steuerung(engl. controller) unterteilt

3.3.1 Model

Im Model werden die Berechnungen ausgeführt. Die Daten werden vom Controller an die View übergeben.

3.3.2 View

In der View werden die Panel programmiert. Die Eingaben des Benutzers werden in der View erfasst und die Berechnungen des Models als plot angezeigt.

3.3.3 Controller

Der Controller übermittelt die Daten der View an das Model indem er die entsprechende Methode aufruft.

3.3.4 Klassendiagramm

3.4 Programmablauf

Beim Start des Programms werden die Panel initialisiert und die default Konfiguration hergestellt. Nun kann entweder ein neuer Filter erzeugt, oder ein bestehender Filter geladen werden. Ein bestehender Filter wird mittels "Menu / load Filter profile" aus einer Textdatei geladen. Ein neuer Filter wird mittels verändern der Schieberegler eingestellt und unter "Menu / save Filter" oder mittels "Ctrl + s" in ein .txt file gespeichert. Mittels button "Add" wird die Filtereinstellung temporär, bis zum Schließen des Programms gespeichert. An der linken Seite des Programms ist das Filterpanel. Darin werden alle eingestellten Filter aufgelistet und können mittels aktivieren der checkbox in verschiedenen Farben angezeigt werden. Der Einfluss eines bestimmten Parameters auf die Funktion kann mittels Monte Carlo Methode ermittelt werden. Dabei wird der Wert in kleinen Schritten verändert zum Beispiel von -10 bis + 10 Prozent und die Kurve für jeden Wert einmal abgebildet. So entstehen mehrere Kurven für die verschiedenen Parameter die man dann vergleichen kann.

4 Testkonzept

Um das fertige Produkt zu testen werden drei Testläufe unternommen. Zuerst wird das Programm durch das Projektteam getestet. Dabei wird das GUI auf Herz und Nieren geprüft. Die Funktionen werden alle mehrmals ausprobiert und die berechnete Kurve, des eingestellten Filters wird mit einer PSpice Simulation mit gleichen Werten verglichen. Im zweiten Testlauf wird das Programm dem Auftraggeber abgegeben. Er testet das Programm und gibt ein Feedback. Er kann noch kleine Änderungen verlangen aber am Grobkonzept wird nichts mehr geändert. In der dritten Testphase wird das Programm an Mitstudenten oder in Elektrotechnik versierte Kollegen abgegeben. Dabei wird getestet, wie benutzerfreundlich das Projekt ist und wie stabil es auf den Rechnern der Tester läuft. Dabei wird auch versucht das Programm einem Stresstest zu unterziehen um es zum Abstürzen zu bringen.

5 Projektvereinbarung

Auftraggeber

Dr. Luca Dalessandro

Ort, Datum

Unterschrift

Projektleiter

Niklaus Schwegler

Ort, Datum

Unterschrift

6 Anhang