**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, 04.04.2019

Inhaltsverzeichnis

[1 Übersicht 3](#_Toc5089211)

[1.1 Ausgangslage 3](#_Toc5089212)

[1.2 Projektziele 4](#_Toc5089213)

[1.3 Wunschziele 4](#_Toc5089214)

[1.4 Nicht-Ziele 6](#_Toc5089215)

[1.5 Lieferobjekte 6](#_Toc5089216)

[2 Softwarekonzept 7](#_Toc5089217)

[2.1 Anforderungen 7](#_Toc5089218)

[2.2 Software/Hardware 7](#_Toc5089219)

[2.3 Mock-Up 7](#_Toc5089220)

[2.4 Menüleiste (noch nicht realisiert) 7](#_Toc5089221)

[2.5 Anzeigefenster DM/CM 8](#_Toc5089222)

[2.6 Haupt- und parasitäre Parameter 8](#_Toc5089223)

[2.7 Beschreibung Programmablauf 8](#_Toc5089224)

[3 Softwarestruktur 10](#_Toc5089225)

[3.1 Klassendiagramm 10](#_Toc5089226)

[4 Theoretische Grundlagen 11](#_Toc5089227)

[4.1 Einleitung EMI Filter 11](#_Toc5089228)

[4.2 Vorgehensweise Berechnung Einfügungsverluste 11](#_Toc5089229)

[5 Testkonzept 12](#_Toc5089230)

[5.1 Kontrolle bei der Entwicklung 12](#_Toc5089231)

[5.2 Überprüfung mit MATLAB 12](#_Toc5089232)

[5.3 Integrationstest 12](#_Toc5089233)

[5.4 Softwaretest 12](#_Toc5089234)

[6 Literaturverzeichnis 13](#_Toc5089235)

# Übersicht

## Ausgangslage

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

Damit diese Störungen (Differential Mode und Common Mode) auf der Netzseite keine negativen 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.

Das Ziel dieser Arbeit ist die Entwicklung eines bedienerfreundlichen Programms, das in der Lage ist, die Einfügungsverluste des Filters zu berechnen, sowie graphisch darzustellen. Das Programm soll ausserdem aufzeigen, wie sich bestimmte Parameter, auf die zwei Störungsarten im ganzen Frequenzspektrum auswirken. Diese Informationen sollten dem Auftraggeber das Dimensionieren von Filtern vereinfachen.

## Projektziele

Die Anforderungen in Tabelle 1 wurden als Pflichtziele definiert. Diese sind notwendig und müssen beim Endprodukt komplett erfüllt werden.

|  |  |  |
| --- | --- | --- |
| **Nr.** | **Bereich** | **Beschreibung** |
| 1 | Software | Die Software basiert auf der Programmiersprache Java, Version JDK11. |
| Die Software läuft auf den aktuellen Betriebssystemen von MacOS (ab Version 10.11) und Windows (7 oder neuer) fehlerfrei. |
| Die Software ist auf der MVC-Architektur realisiert. |
| Die Bedienoberfläche wird mit JavaFX realisiert. |
| 2 | Bedienoberfläche | Die Benutzeroberfläche soll für 80% der Benutzenden als bedienungsfreundlich und selbsterklärend empfunden werden. |
| Die Bedienoberfläche wird mit JavaFX realisiert. |
| Die Werte der parasitären Parameter sind um ±30 % veränderbar. |
| Die Einfügungsverluste werden in Abhängigkeit der Frequenz [CM & DM] graphisch dargestellt. |
| 3 | Berechnungen | Die Berechnungen der Einfügungsverluste sind korrekt. |
| Die Berechnungen der Einfügungsverluste dauern weniger als eine Sekunde. |
| Die Schaltung wird mit den relevanten Bauteilen entsprechend der Aufgabenstellung simuliert. |

Tabelle 1: Projektziele

## Wunschziele

Je nach zeitlichem Verlauf des Projektes können (müssen aber nicht) Zusatzfunktionen implementiert werden. Voraussetzung ist, dass das Projekt keine Verzögerungen erfährt. In Tabelle 2 werden die Wunschziele definiert.

|  |  |  |
| --- | --- | --- |
| **Nr.** | **Bereich** | **Beschreibung** |
| 1 | Ausgabedateien | Die Ergebnisse (Graphen & Parameter) können als PDF Datei gespeichert werden. |
| 2 | Eingabedateien | Auf gespeicherte Ergebnisse kann zugegriffen werden (Save-Load-Option). |
| 3 | Programm | Bei Parameteränderungen mittels Slider wird eine zusätzliche Kurve der Verluständerung dargestellt. |
| Die Sensibilitätsanalyse kann dargestellt werden. |
| Es können zusätzlich eigene Schaltungen erstellt und simuliert werden. |

Tabelle 2: Wunschziele

## 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. Diese Nicht-Ziele werden in Tabelle 3 aufgelistet.

|  |  |  |
| --- | --- | --- |
| **Nr.** | **Bereich** | **Beschreibung** |
| 1 | Darstellung | Es wird eine 3D Darstellung realisiert. |
| 2 | Eingabefelder | Es wird eine Funktion für einen Bauelementvorschlagrechner zur Erreichung einer Verlustkurve nach Wunsch realisiert. |
| 3 | Webapplikation | Es wird eine App (für Smart-Devices) erstellt. |
| 4 | Simulation | Es wird eine Funktion zur Berechnung der Permeabilität der Spulen in Abhängigkeit der Frequenz zur Dämpfung erstellt. |

Tabelle 3: Nicht-Ziele

## Lieferobjekte

Die Lieferobjekte (siehe Tabelle 4) 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 |
| 21.03.2019 | Pflichtenheft Version 1 |
| 07.04.2019 | Pflichtenheft Endversion |
| 11.04.2019 | Zwischenpräsentation |
| 02.05.2019 | Disposition, Einleitung |
| 05.05.2019 | Mock-up Software |
| 06.05.2019 | Berechnungen mit Matlab |
| 13.05.2019 | Betaversion von Software |
| 13.06.2019 | Fachbericht und Präsentation |

Tabelle 4: Lieferobjekte

# Softwarekonzept

## 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.

## Software/Hardware

Die Software ist in der Programmiersprache Java realisiert und besitzt eine MVC Architektur. Sie soll auf den aktuellen Betriebssystemen von MacOS (ab Version 10.11) und Windows (7 oder neuer) Fehlerfrei funktionieren. Für Java haben wir uns entschieden, da es eine weit verbreitete Programmiersprache ist und der Auftraggeber somit flexibel für die Vergabe von Wartungs-/ Änderungs- oder Verbesserungsarbeiten bleibt. Die Software soll so strukturiert sein, dass nachträgliche Änderungen und Upgrades ohne grossen Aufwand möglich sind. Deshalb haben wir uns für die MVC-Architektur entschieden. Dabei wird die Software in drei Bereiche unterteilt: das Model, die View und der Controller. Das Model enthält die Berechnungen für Impedanzen, Kettenmatrizen und Einfügedämpfungen. Die View enthält die Elemente der Benutzeroberfläche. Der Controller verknüpft die View mit dem Model. Die View ist dabei noch weiter unterteilt.

## Mock-Up



Abbildung 10: GUI Übersicht

Die grafische Bedienoberfläche besteht voraussichtlich aus einer Menüleiste, einem Anzeigefenster für jeweils DM und CM, sowie Haupt- und Parasitärparametern (Abbildung 10). Es wird die Bibliothek JavaFX verwendet, um durch Verwendung von Cascading Style Sheets (CSS) den Inhalt noch weiter von der Darstellung trennen zu können. Damit die Bedienoberfläche selbsterklärend und noch benutzerfreundlicher wird, verwenden wir die Tool Tip API. Die Tooltipps werden angezeigt, sobald man mit dem Mauspfeil während zwei Sekunden über einer Oberfläche oder Bedienfeld verweilt.

## 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.

## Anzeigefenster DM/CM

Die Einfügungsverluste werden jeweils separat für CM und DM in einem Bodediagramm mit Frequenz- und Impedanz Achse dargestellt, wobei die Frequenzachse logarithmisch skaliert ist. Die Kurvendiagramme können einzeln mit einem Doppelklick vergrössert werden um gewünschte/spezielle Stellen der Kurve genauer betrachten zu können. Die Einfügungsverluste werden in Abhängigkeit der Frequenz für CM und DM, graphisch im Bodediagramms als Kurve dargestellt, dies begünstigt eine schnelle Gewinnung von Erkenntnissen zum simulierten Filter.

## Haupt- und parasitäre Parameter

****

Abbildung 11: Parametereinstellungen

Jeder Hauptparameter wird durch eine gewisse Anzahl parasitärer Parameter dargestellt. Letztere können einzeln via Schieberegler um maximal ±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 (siehe Abbildung 11).

## 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 Sessionen 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 (siehe Abbildung 12). Nun können die ursprünglichen Werte der parasitären Parameter via Textfelder und anderen Kontroll-Einheiten geändert werden. Bei erneutem Doppelklick geht die Ansicht wieder zurück zu den Schiebereglern (siehe Abbildung 13).



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 (siehe Abbildung 14)



Abbildung 14: Parameterauswirkung

Wenn Nutzende den Mauszeiger über einen Schieberegler bewegen, soll im Diagramm gezeigt werden, welche Auswirkungen dieser Schieberegler bei voller Auslenkung (±30%) auf die Kurve hätte. In der Abbildung 14 als blaue Kurve dargestellt. Somit können die unterschiedlichen Auswirkungen aller parasitären Parameter schnell eingesehen werden, ohne dass die Nutzenden die Schieberegler einzeln verschieben müssen.

# Softwarestruktur

In der Abbildung 15 ist das Klassendiagramm der Software ersichtlich.

## Klassendiagramm



Abbildung 15: Klassendiagramm

# Theoretische Grundlagen

## 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.

## Vorgehensweise Berechnung Einfügungsverluste

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 vereinfachte Schaltung ist in der Abbildung 7 sichtbar



Abbildung 7: Vereinfachte CM-Schaltung

Mit der vereinfachten Schaltung bestimmen wir für alle Bauteile die Impedanzen für eine bestimmte Frequenz. Danach berechnen wir die Impedanz (Z), sowie die Admittanz (Y0). Daraus ergeben sich Zweitore. Daraus ergeben sich die Kettenmatrix (A1) für die Längsimpedanz (Z) sowie (A2) für die Queradmittanz (Y0).

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

Nun können wir die Einfügungsdämpfung a in dB berechnen: [1]

Dieser Parameter wird mit verschiedenen Frequenzen berechnet, damit man die Einfügungsdämpfung über dem ganzen Spektrum darstellen kann.

Für die DM-Schaltung geht man genau gleich vor. Diese Schaltung hat aber noch eine weitere Querimpedanz in der Schaltung. Die Gesamtmatrix für den S21 Parameter berechnet sich dann aus drei Zweitoren. Durch die objektorientierte Programmierung kann diese aber problemlos implementiert und falls nötig, jederzeit angepasst werden.

# 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.

## 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.

## 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.

## 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.

## 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.

# Literaturverzeichnis

|  |  |
| --- | --- |
| [1] | A. Maier und S. Rupp, «Hochfrequenztechnik, Teil 2 - Anwendungen,» DHBW, Stuttgart, 2016. |