



EMV-konformes Design

Transient Simulation(MECH-M-3-SVE-TSI-ILV)

Master Mechatronic & Smart Technologies

3. Semester

Lehrveranstaltungsleiter: Thomas Gadner

Jahrgang: MA-MECH-23-BB

Verfasser: Lukas Sieß

10. Februar 2025

Inhaltsverzeichnis

1 Einleitung

In diesem Bericht wird die Umsetzung der zweiten Aufgabe im Rahmen der Regelungstechnik und Leistungselektronik analysiert. Ziel dieser Untersuchung ist die praxisnahe Anwendung theoretischer Konzepte aus den ersten beiden Semestern der Regelungstechnik durch die Modellierung und Regelung eines Leistungselektroniksystems. Hierbei wird ein digitaler Regler entworfen und auf einer geeigneten Plattform implementiert.

Die Aufgabenstellung gliedert sich in zwei zentrale Arbeitsschritte:

- Modellierung eines Leistungselektronik-Wandlers in PLECSIm ersten Schritt wird ein Modell eines Leistungselektronik-Wandlers unter Verwendung von PLECS erstellt. Dieses Modell bildet die Grundlage für die nachfolgende Regelungsstrategie. Um die Komplexität der Regelung zu reduzieren, kann das System so ausgelegt werden, dass es eine vergleichsweise träge Dynamik aufweist. Dadurch wird es ermöglicht, klassische kontinuierliche Regler zu verwenden, sofern eine ausreichend hohe Abtastrate gewährleistet ist. (Siehe Abbildung:)
- Implementierung des Modells auf einer RTBox mit einem STM-Mikrocontroller als ReglerNach der erfolgreichen Modellierung erfolgt die Implementierung des Wandlers auf einer RTBox. Die Regelung des Systems wird durch einen Mikrocontroller (STM) realisiert, welcher die Steueralgorithmen ausführt und die Systemleistung in Echtzeit analysiert.

Implementierung des Modells auf einer RTBox mit einem STM-Mikrocontroller als ReglerNach der erfolgreichen Modellierung erfolgt die Implementierung des Wandlers auf einer RTBox. Die Regelung des Systems wird durch einen Mikrocontroller (STM) realisiert, welcher die Steueralgorithmen ausführt und die Systemleistung in Echtzeit analysiert.

Dieser Bericht dokumentiert die einzelnen Entwicklungs- und Implementierungsschritte unter Berücksichtigung der theoretischen Grundlagen sowie der praktischen Herausforderungen. Der Fokus liegt auf der präzisen Modellbildung, der Auswahl geeigneter Regelungsparameter sowie der erfolgreichen Umsetzung auf der RTBox mit einem STM-Mikrocontroller als Steuergerät.

2 Task 1

Die Modellierung eines Leistungselektronik-Wandlers ist ein essenzieller Schritt zur Analyse und Regelung des Systems. In diesem Kapitel wird zunächst die theoretische Grundlage für die Modellbildung erörtert. Anschließend wird das erstellte Modell detailliert beschrieben, einschließlich der verwendeten Komponenten und Parameter. Besondere Aufmerksamkeit wird auf die Wahl der Modellstruktur und deren Einfluss auf das dynamische Verhalten des Systems gelegt. Darüber hinaus werden mögliche Vereinfachungen diskutiert, um eine effiziente Implementierung des Reglers zu ermöglichen. (Siehe Abbildung: ?? und ??)

Ein besonderer Fokus lag auf der Auslegung eines PID-Reglers für das System. Verschiedene Methoden zur Bestimmung der Reglerparameter wurden getestet, darunter System Identification, State-Space-Modelle, Pole Placement und PID-Tuning. Trotz dieser Ansätze konnte der PID-Regler nicht erfolgreich ausgelegt werden. Die Übernahme von Parametern aus der Fachliteratur, beispielsweise aus dem Paper "Optimally Designed PID Controller for a DC-DC Buck Converter via a Hybrid Whale Optimization Algorithm with Simulated Annealing", lieferte ebenfalls nicht das gewünschte Ergebnis. Daher wurden die PID-Parameter letztendlich empirisch angepasst, um die gewünschte Regelgüte zu erreichen.

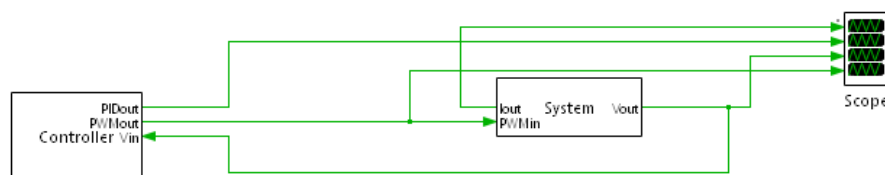


Abbildung 2.1: FFT LTC3639 mit Filter

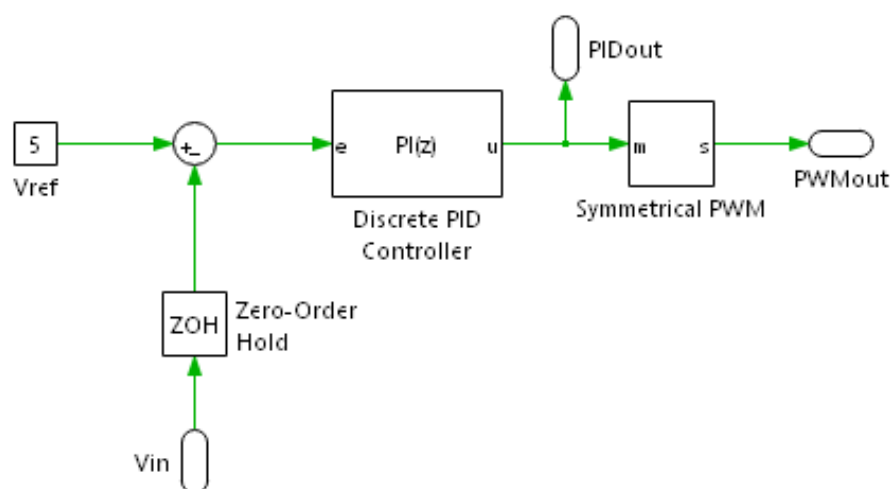


Abbildung 2.2: FFT LTC3639 mit Filter

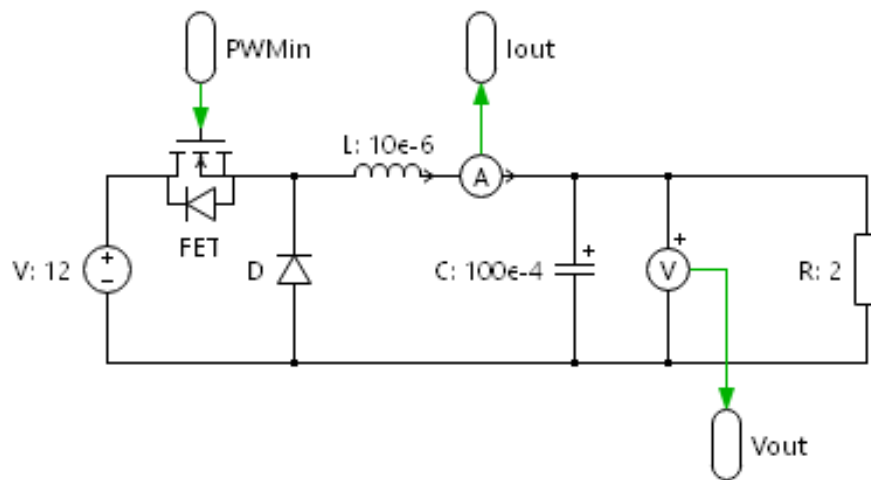


Abbildung 2.3: FFT LTC3639 mit Filter

2.1 PID Auslegung

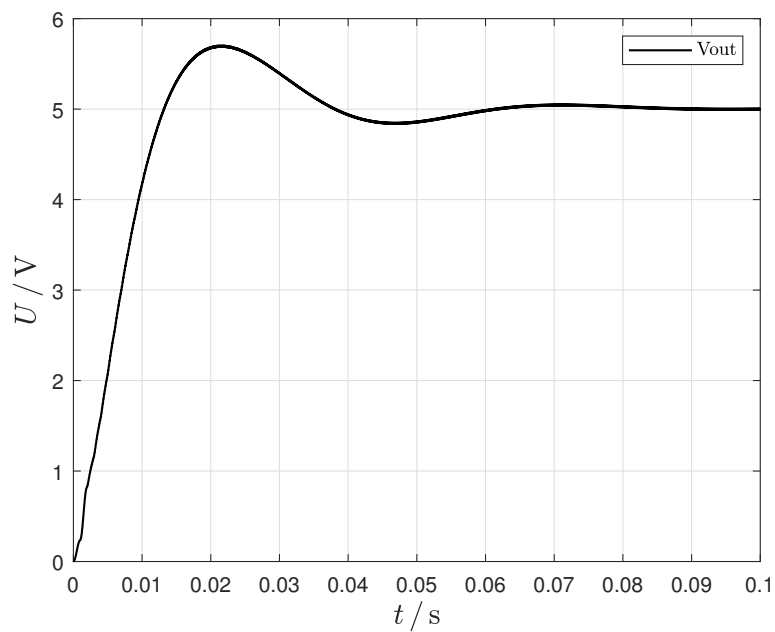


Abbildung 2.4: FFT LTC3639 mit Filter

3 Task 2

Nach der erfolgreichen Modellierung des Wandlers in PLECS erfolgt die Umsetzung auf einer RTBox. In diesem Kapitel werden die notwendigen Schritte zur Implementierung detailliert erläutert. Dies umfasst die Echtzeitfähigkeit der RTBox, die Anbindung an den Mikrocontroller sowie die Entwicklung der Steueralgorithmen. Zudem wird auf die Herausforderungen bei der Echtzeitregelung und die erforderlichen Anpassungen des Modells eingegangen. Abschließend erfolgt eine Analyse der Performance des implementierten Systems anhand von Simulationen und experimentellen Ergebnissen.

Zur Umsetzung wurden zwei Subsysteme definiert: eines für den Controller und eines für das zu regelnde System. Diese Trennung ermöglicht eine separate Parametrierung der Coder-Optionen für eine optimierte Code-Generierung. In der Hardware-Implementierung wurden die Ein- und Ausgänge wie folgt konfiguriert und miteinander verschaltet:

- System-Subsystem: Enthält PLECS PWM Capture sowie PLECS Analog Out für Strom- und Spannungsmessung. (Siehe Abbildung: *ModelTask2system.png*)
- Controller-Subsystem: Enthält STM PWM Out zur Steuerung sowie STM ADC (Analog In Triggered) zur Erfassung der Systemrückmeldung. Zur korrekten Triggierung des ADC wurde ein Control Trigger Block implementiert. (Siehe Abbildung: *ModelTask2Controller.png*)

Die Simulation wurde mit zwei verschiedenen Lasten ($R_{Load} = 2\Omega$ und) durchgeführt, um die Performance des Reglers unter unterschiedlichen Bedingungen zu bewerten. Dabei wurde das PLECS-Modell schrittweise für die Hardware-Simulation mit der RT-Box und dem STM32 Nucleo angepasst. Die in den Coder-Optionen festgelegten Parameter waren:

- Controller: Target = STM32G4x; Scheduling Step Size = (Siehe Abbildung: *Settings CoderOptionsModelTask2ControllerTarget.png*)
- System: Target = PLECS RT Box 2; Scheduling Step Size = (Siehe Abbildung: *Settings CoderOptionsModelTask2systemTarget.png*)
- Gesamtsystem-Ansicht: Überblick der vollständigen Implementierung (Siehe Abbildung: *ModelTask2.png*)

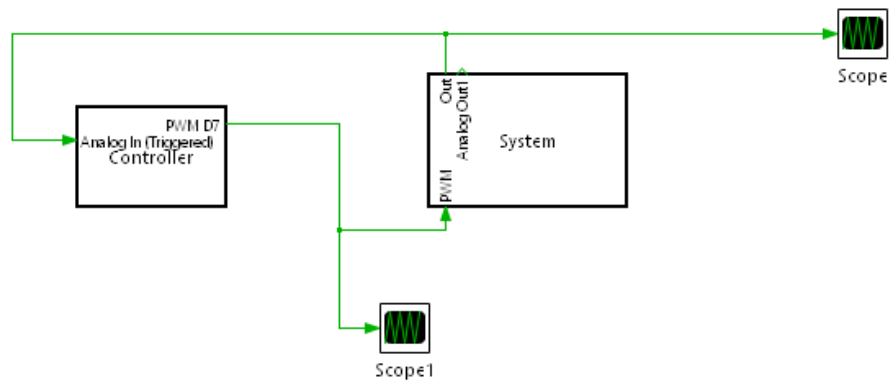


Abbildung 3.1: FFT LTC3639 mit Filter

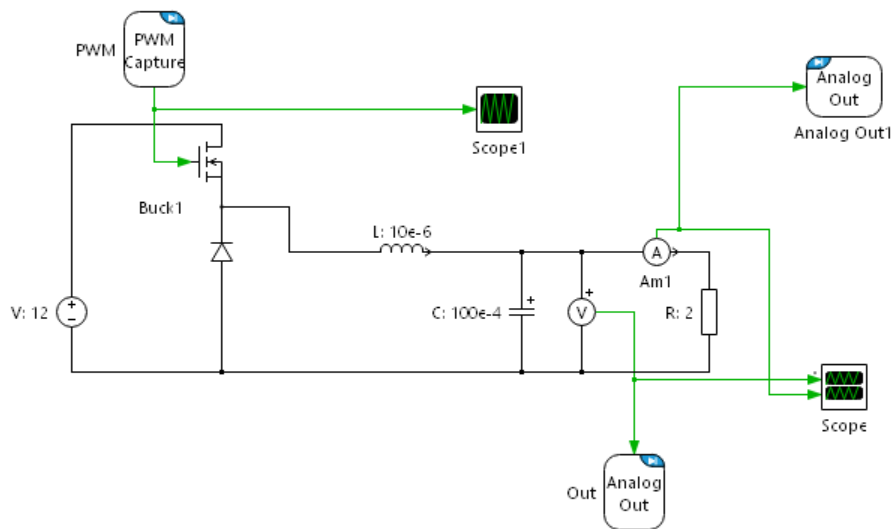


Abbildung 3.2: FFT LTC3639 mit Filter

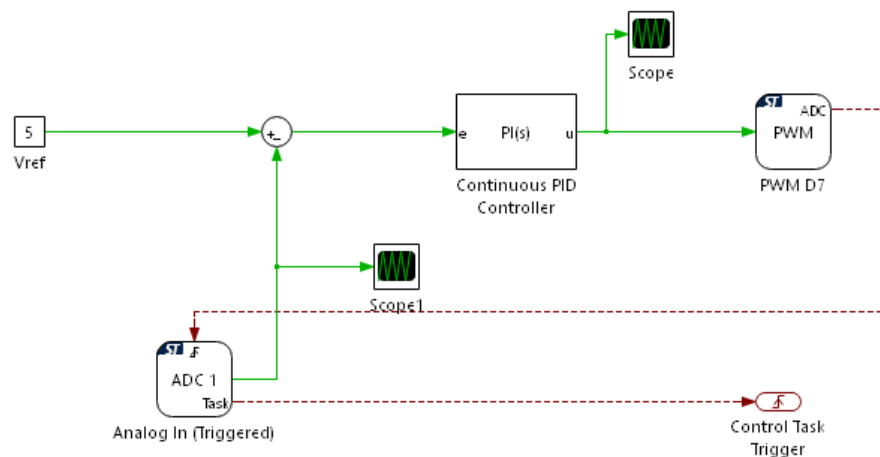


Abbildung 3.3: FFT LTC3639 mit Filter

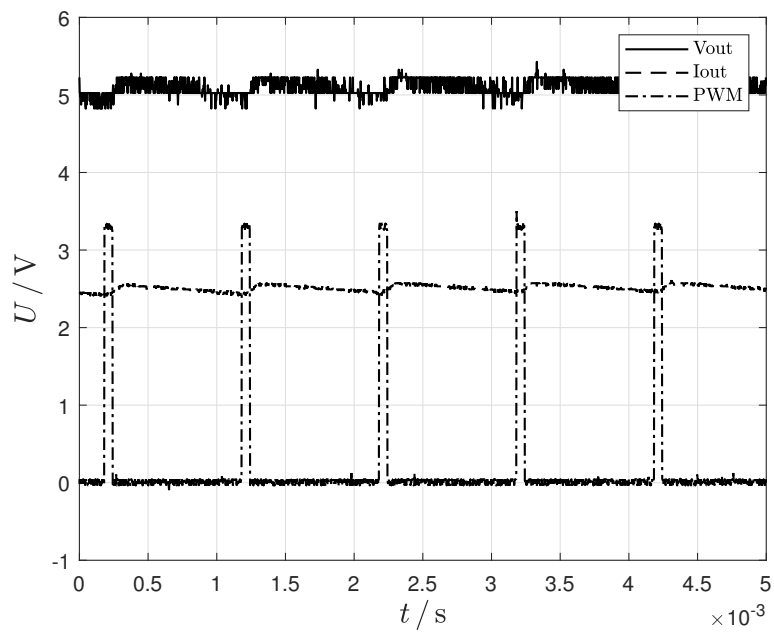


Abbildung 3.4: Messung mit $R_{load} = 2 \Omega$

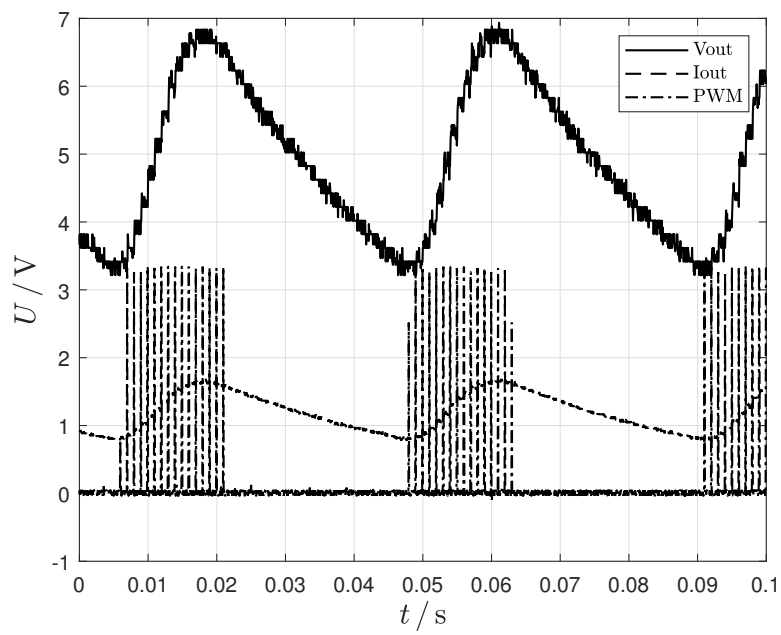


Abbildung 3.5: Messung mit $R_{load} = 4 \Omega$

A Anhang

A.1 MATLAB Skript