

Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

KIRA – KI-Methoden zur optimierten Regelung elektrischer Traktionsantriebe

Partnerspezifischer Schlussbericht

Förderkennzeichen: 19I21030H
 Teilvorhaben: Regelung optimiert durch KI
 Laufzeit des Vorhabens: 01.08.2021 – 31.07.2024
 Berichtszeitraum: 01.08.2021 – 31.07.2024
 Vorgelegt durch: MACCON GmbH und Co. KG
 Autoren: Simon Wiedemann
 Datum: 17.09.2025

Projektkonsortium:

- Robert Bosch GmbH (Koordinator)
- AVL Software and Functions GmbH
- Infineon Technologies AG
- AixControl GmbH
- Fraunhofer Institut für Integrierte Systeme und Bauelementetechnologie
- AVL SET GmbH
- Siemens AG
- metaphacts GmbH
- MACCON GmbH und Co. KG
- Institut für Stromrichtertechnik und Elektrische Antriebe, RWTH Aachen

Inhaltsverzeichnis

1.	Teil 1 - Kurzdarstellung	2
1.1	Aufgabenstellung.....	2
1.1.1	Vorhabensbeschreibung Verbund	2
1.1.2	Vorhabensbeschreibung MACCON spezifisch.....	2
1.2	Voraussetzungen, unter denen das Vorhaben durchgeführt wurde	3
1.3	Planung und Ablauf des Vorhabens	3
1.4	Wissenschaftlicher und technischer Stand, an denen angeknüpft wurde.....	4
1.5	Zusammenarbeit mit anderen Stellen.....	4
2.	Teil 2 – Eingehende Darstellung	5
2.1	Verwendung der Zuwendung und der erzielten Ergebnisse im Einzelnen, mit Gegenüberstellung der vorgegebenen Ziele	5
2.1.1	AP 1: Bewertung von KI-Methoden im Traktionsantrieb.....	6
2.1.2	AP 2: Regelung.....	9
2.1.3	AP 3: Modellierung.....	9
2.1.4	AP 4: Prototypische Umsetzung.....	10
2.1.5	Motor Control Environment in Matlab/Simulink für Prüfstand 2	17
2.1.6	Selbstinbetriebnahme - Veröffentlichung I.....	28
2.1.7	Kompensation von Subharmonischen für die geberlose Regelung.....	44
2.1.8	Onlineoptimierung der Positionsfehler-ANNs	48
2.1.9	Ergebnisse der Optimierung in Simulink.....	48
2.1.10	Zusammenfassung.....	56
2.2	Wichtigste Positionen des zahlenmäßigen Nachweises	57
2.3	Notwendigkeit und Angemessenheit der geleisteten Arbeit.....	57
2.4	Eingehende Darstellung des voraussichtlichen Nutzens, insbesondere der Verwendbarkeit des Ergebnisses im Sinne des fortgeschriebenen Verwertungsplans.....	58
2.5	Eingehende Darstellung des während der Durchführung des Vorhabens dem Zuwendungsempfänger bekannt gewordenen Fortschritts auf diesem Gebiet bei anderen Stellen	59
2.6	Erfolgte oder geplante Veröffentlichungen der Ergebnisse.....	59
3.	Literaturverzeichnis	60

1. Teil 1 - Kurzdarstellung

1.1 Aufgabenstellung

1.1.1 Vorhabensbeschreibung Verbund

Das Förderprojekt KIRA adressiert den stark wachsenden Markt von automatisierten und teilautomatisierten Fahrzeugen mit Fokus auf intelligente Antriebslösungen. Fokus des Projekts ist die Verwendung von KI-basierten Verfahren zur Optimierung und optimalen Regelung elektrischer Antriebe. Diese Verfahren sollen im Projekt so weiterentwickelt werden, dass sie in der Lage sind wichtige Probleme elektrischer Antriebe insbesondere in autonomen Fahrzeugen zu lösen. Neben den erhöhten Anforderungen an die Drehmomentgenauigkeit ist vor allem die Unterdrückung akustischer Emissionen und die Erhöhung der Reichweite durch optimierte Ansteuerung und verbesserte Ausnutzung der Komponenten Ziel des Projekts KIRA. Hinsichtlich der technischen Umsetzung bieten KI-basierte Verfahren gegenüber traditionellen „starr“ Ansätzen neue Freiheitsgrade in Regelung und Modellierung sowohl auf Komponenten- als auch Systemebene. Mit KI-basierten Betriebsstrategien ist die Beherrschung hochkomplexer Systeme möglich. Auf Komponentenebene können auch komplexe multiphysikalische Problemstellungen mit KI-basierten Verfahren echtzeitfähig angegangen werden. Für den Projekterfolg ist die enge und domänenübergreifende Verzahnung der Partnerbeiträge mit einem breit gefächerten Kompetenzspektrum entscheidend. Die Zusammenarbeit der Partner wird durch eine eng verlinkte Arbeitspaketstruktur sichergestellt. Durch die zielgerichtete Anforderungsanalyse wird zu Beginn des Projekts sichergestellt, dass die Bewertung der im Projekt entwickelten Methoden nach belastbaren und nachvollziehbaren Kriterien geschieht. Durch die Einbeziehung der Partner aus verschiedenen Stufen der Lieferkette wird sichergestellt, dass der Bezug zur Anwendung immer im Fokus steht. Die Validierung der Projektergebnisse geschieht auf mehreren Ebenen, von der simulativen Validierung bis zum Test an verschiedenen Teildemonstratoren und einem skalierten Fahrzeugdemonstrator, der als Anwendungsbeispiel für ein autonomes Fahrzeug dient. [6]

1.1.2 Vorhabensbeschreibung MACCON spezifisch

Das Ziel dieser Arbeit war die Entwicklung von praxistauglichen KI-basierten Algorithmen und Systemmodellen, welche für die Inbetriebnahme und den Betrieb von unbekanntem isotropen und anisotropen Synchronmaschinen verwendet werden können. Die KI-basierten Algorithmen und Systemmodelle bieten die Möglichkeit eines Einsatzes für einen offline Betrieb (Maschine nicht im Betrieb; im Stillstand) sowie online (im Betrieb). Anhand dieser Techniken soll das Betriebsverhalten, wie die Dynamik der Regler und die Energieeffizienz der Maschine optimiert werden sowie die Genauigkeit von Positionsgeberlosen Regelungen. Da in der Praxis moderne Synchronmaschinen nichtlineares magnetisches Verhalten aufweisen, sind KI-basierte Maschinen-Modelle von isotropen und anisotropen Synchronmaschinen entwickelt worden, welche nichtlineares Sättigungsverhalten und Kreuzkopplung berücksichtigen können sowie Subharmonische der geberlosen Regelung. Weiterhin sind

praxistaugliche Modelle erstellt worden, welche die nichtlinearen und parasitären Effekte des Wechselrichters in Abhängigkeit mehrerer Eingangsvariablen z.B. Strom, Spannung, Temperatur, Schaltfrequenz, berücksichtigen können. Diese Modelle können während einer offline Inbetriebnahme durch Identifizierungs-Algorithmen optimiert werden. In einem weiteren Schritt ist die Möglichkeit gegeben, während des laufenden Betriebs (online), diese Modelle durch KI-Techniken zu optimieren, um Einflüsse wie Temperaturverhalten und Lasten zu berücksichtigen. Basierend auf den KI-Modellen, kann während des Betriebs, eine Optimierung hinsichtlich der Energieeffizienz und Dynamik erfolgen. Um die Robustheit zu evaluieren und um die Einsetzbarkeit in der Praxis zu erproben, sind unterschiedliche Motortechnologien an einem Prüfstand für unterschiedliche Lasten geprüft worden.

1.2 Voraussetzungen, unter denen das Vorhaben durchgeführt wurde

Als Voraussetzung dienen die theoretischen Arbeiten aus [2] sowie der aufgeführte Prüfstand aus Kapitel 2.1.4.

1.3 Planung und Ablauf des Vorhabens

Die Grobzeitplanung des Projektes KIRA ist in folgender Abbildung dargestellt. Die Laufzeit des Projekts war drei Jahre: [6]

Im Projekt sind Meilensteine (M) mit folgenden Ergebnissen vorgesehen:

Nr.	Beschreibung	Projektmonat
M1	Anwendungsfälle der im Projekt untersuchten KI-Methoden sowie Anforderungen an die Komponenten und die Verfahren sind definiert. Die Interaktion von	6
M2	Bewertungsmatrizen für Regelungsverfahren und Modelle sind identifiziert und aufgestellt. Der Fokus liegt auf der Bewertung der in Abschnitt 3.4 definierten Zielen.	15
M3	Prüfstands- und Validierungskonzepte sind erarbeitet und fixiert. Ein Versuchsplan wurde erstellt.	18
M4	Hochpräzise Modelle zum Training von KI sind erstellt. Die Anforderungen zum Training der KI-basierten Modelle wurden erfüllt.	25
M5	Wechselwirkungen zwischen Fahrzeug und KI-basierten Traktionsantrieb sind untersucht. Die Fahrzeugparameterschätzung ist evaluiert.	33
M6	KI-basierte Regelungsverfahren und Modelle sind konzeptioniert. Die Implementierung auf Rapid-Control-Prototyping Systemen ist abgeschlossen.	33
M7	KI-Methoden zur Regelung elektrischer Antriebe sind simulativ evaluiert. Die in AP1 definierten Anforderungen wurden erfüllt.	36
M8	KI-basierte Regelungsverfahren und Modelle sind am Prüfstand umgesetzt und validiert. Die gesetzten Ziele zur Effizienzsteigerung, Reduktion akustischer Emissionen und Erhöhung der Ausnutzung wurden erreicht.	36

Jahr		2022				2023				2024		
Quartal	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3
1.1 Definition von Anforderungen		M1										
1.2 Betrachtungen auf Systemebene											M5	
1.3 Evaluation von Regelungsverfahren						M3						M7
1.4 Evaluation der Modellierung												
2.1 Regelung optimiert durch KI												
2.2 KI-basierte Regelung												M6
3.1 Präzise Modelle zum Training von KI								M4				
3.2 KI-basierte Modelle												
4.1 Aufbau Demonstrator / Prüfstand						M2						
4.2 KI-gesteuerter Prüfstand												
4.3 Skalierter Fahrzeugdemonstrator												
4.4 Validierung und Evaluation am Prüfstand												M8

1.4 Wissenschaftlicher und technischer Stand, an denen angeknüpft wurde

Der wissenschaftliche und technische Stand, an denen angeknüpft wurde, ist in den Kapiteln 2.1.6 und 2.1.7 dargestellt.

1.5 Zusammenarbeit mit anderen Stellen

MACCON stand im stetigen Austausch mit den Konsortialpartnern. Weitere Stellen wurden nicht inkludiert.

2. Teil 2 – Eingehende Darstellung

2.1 Verwendung der Zuwendung und der erzielten Ergebnisse im Einzelnen, mit Gegenüberstellung der vorgegebenen Ziele

Festlegungen sowie wissenschaftliche und technische Ergebnisse der Arbeiten in den einzelnen Arbeitspaketen (AP) werden im Folgenden vorgestellt. Es werden ausschließlich die Arbeitspakete beschrieben, an welche MACCON laut der Vorhabensbeschreibung beteiligt war.

MACCON war an den nachfolgend aufgeführten Arbeitspaketen beteiligt.

AP 1: Bewertung von KI-Methoden im Traktionsantrieb

AP 1.1: Definition von Anforderungen

AP 1.1.1 Definition von Anwendungsfällen und Nutzungsbereich der KI-Ansätze

AP 1.1.2 Voruntersuchungen und Literaturstudien: KI-Methoden, Modelkomplexität, funktionale Anforderungen

AP 1.1.3 Definition von Bewertungsmatrizen

AP 1.3: Evaluation von Regelungsverfahren

AP 1.3.1 Definition von Bewertungskriterien

AP 1.3.2 Evaluation

AP 1.4: Evaluation der Modellierung

AP 1.4.1 Definition von Bewertungsmatrizen

AP 1.4.2 Erklärbarkeit der KI-Ergebnisse

AP 1.4.3 Evaluation

AP 2: Regelung

AP 2.1: Regelung optimiert durch KI

AP 2.1.2 Betriebspunktabhängige Regelungsparameter

AP 2.1.3 Beschleunigte Charakterisierung

AP 3: Modellierung

AP 3.2: KI-basierte Modelle

AP 3.2.5 Formale Wissensrepräsentation zur verbesserten Modellbildung

AP 4: Prototypische Umsetzung

AP 4.1: Aufbau Demonstrator/Prüfstand

AP 4.4: Validierung und Evaluation am Prüfstand

2.1.1 AP 1: Bewertung von KI-Methoden im Traktionsantrieb

Teilarbeitspakete innerhalb AP 1 bestehen aus der **Definition von Anforderungen** (AP 1.1), der Evaluation der Regelungsverfahren (AP 1.3) und der Evaluation der Modellierung (AP 1.4)

2.1.1.1 AP 1.1: Definition von Anforderungen

AP 1.1.1 Definition von Anwendungsfällen und Nutzungsbereich der KI-Ansätze

Folgend werden die Anwendungsfälle innerhalb dieses Vorhabens vorgestellt, welche auch auf elektrische Traktionsantriebe übertragbar sind. Als Motortechnologie hat MACCON dreiphasige Synchronmaschinen verwendet mit unterschiedlichen Rotortechnologien, da diese hauptsächlich in elektrischen Fahrzeugen eingesetzt werden. Als Umrichter-Topologie wurden „Drei-Phasen 2-Level Spannungs-Umrichter“ verwendet. Die entwickelten Verfahren wurden auf drei verschiedenen Motorprüfständen und Recheneinheiten implementiert.

Anwendung: Die entwickelte KI kann in folgenden Bereichen eingesetzt werden:

- 1. Vereinfachung der Inbetriebnahme elektrischer Synchronmaschinen. Dies bedeutet, dass Teile oder die gesamten Regelschleifen, einer klassischen kaskadierten Regelung automatisch in Betrieb genommen werden kann (siehe Kapitel 4). Dies hat im Vergleich zu einer manuellen Inbetriebnahme folgende Vorteile:
 - Ersparnis an Zeit.
 - Ersparnis an finanziellen Mitteln durch weniger Personal
 - Genauigkeit der Regelung
 - Die Inbetriebnahme wird sicherer
 - Genauigkeit der Inbetriebnahme verbessert sich und daher
 - Der Antrieb wird Energieoptimaler verwendet

Es ergibt sich daher letzten Endes ein optimierter Betrieb dieser Maschinen in Hinblick auf Energieeffizienz und Regler-Genauigkeit sowie der Dynamik.

- 2. Geberlose Verfahren zur Bestimmung der Rotorlage sind durch nichtlineare Effekte des Elektromotors und des Spannungsumrichters mit Schätzungsfehlern verbunden. Die entwickelte KI ermöglicht eine Modellierung, Identifikation und Kompensation dieser nichtlinearen Positionsschätzfehlers (siehe Kapitel 5 und 6).

AP 1.1.2 Voruntersuchungen und Literaturstudien: KI-Methoden, Modelkomplexität, funktionale Anforderungen

Für die oben beschriebenen Anwendungen wurden folgende KI-Methoden in Betracht gezogen:

- Einfache neuronale Netzwerke („shallow neural network“) mit einer „supervised optimisation“ zur Modellierung der nichtlinearen Spannungsfehlereffekte des Umrichters sowie der nichtlinearen Sättigungseffekte der Synchronmaschinen. Weiterhin wurden Modelle entwickelt, welche die subharmonischen Schätzfehler einer geberlosen Regelung (injektionsbasierte Schätzverfahren) abbilden können.

Die Modelkomplexität wurde so gering wie möglich gehalten, um eine industriennahe Implementierung zu ermöglichen. Daher wurden bei den entwickelten Modellen möglichst auf rechenintensive Funktionen verzichtet (wie die exponential- und Log-Funktion) und eine geringe Anzahl von Modellparametern eingesetzt. Weiterhin sind die entwickelten Modelle echtzeitfähig.

- Optimierungsalgorithmen zur Identifikation von Modellparametern wie z.B.
 - o „Genetic Algorithm“
 - o „Particle Swarm Optimisation“
 - o „Bee Colony Algorithm“
 - o Levenberg-Marquardt
 - o Recursive-least-squares
 - o ADAM

Einzelne dieser Optimierungsalgorithmen wurden neben einer zufriedenstellenden Konvergenz auch in Bezug auf dessen Konvergenzgeschwindigkeit untersucht.

Die oben beschriebenen Anwendungen wurden mit Standard-Sensoren eines handelsüblichen Umrichters umgesetzt. Dazu wurden lediglich die Messgrößen der Phasenströme verwendet.

Als Regler Architektur wurde eine kaskadierte Regelung, bestehend aus Strom- und Geschwindigkeitsregler verwendet.

Folgende Literatur oder Medien wurden zur Erreichung der Ziele untersucht:

- o Selbstinbetriebnahme von elektrischen Maschinen
- o Parameteridentifikation von elektrischen Maschinen

- o Charakterisierung von elektrischen Maschinen und Umrichtern
- o Neuronale Netzwerke
- o „Supervised Learning“
- o „Genetic Algorithm“
- o „Bee Colony Algorithm“
- o „Fuzzy Logic“
- o Matlab Deep Learning Toolbox
- o Tensor Flow
- o Symbolic Regression Tools
- o Online Learning
- o Stochastic Gradient Decent

AP 1.1.3 Definition von Bewertungskriterien

Folgende Tabelle hat bei der Auswahl der AN-Netzwerke und Optimierungs-Algorithmen geholfen:

Benötigte Logische Elemente (FPGA)	
Benötigte Multiplikatoren	
Benötigter Speicher (uC)	
Durchführungsgeschwindigkeit [s]	
Anzahl Modellparameter	
Maximal Error [%]	
Mean Error [%]	
Anzahl Eingangsvariablen	
Linear oder Nichtlinear	
Physikalisch interpretierbar	
Praxistauglichkeit	
Echtzeitfähig	

„****“: Modell Umrichter, Modell Maschine (Psi_d, Psi_q), Modell Subharmonische für die geberlose Regelung, Optimierungs-Algorithmus online, Optimierungs-Algorithmus offline

2.1.1.2 AP 1.3: Evaluation der Regelungsverfahren

AP 1.3.1 Definition von Bewertungskriterien

Siehe AP 1.1.3 Definition von Bewertungskriterien

AP 1.3.2 Evaluation

Dieses Thema wird in den Kapiteln 2.1.6 bis 2.1.9 behandelt.

2.1.1.3 AP 1.4: Evaluation der Modellierung

AP 1.3.1 Definition von Bewertungskriterien

Siehe AP 1.1.3 Definition von Bewertungskriterien

AP 1.3.2 Evaluation

Dieses Thema wird in den Kapiteln 2.1.6 bis 2.1.9 behandelt.

2.1.2 AP 2: Regelung

2.1.2.1 AP 2.1: Regelung optimiert durch KI

AP 2.1.2 Betriebspunktabhängige Regelungsparameter

Dieses Thema wird in dem Kapitel 2.1.6 behandelt und wird aus übersichtgründen hier nicht weiter aufgeführt. In der dort aufgeführten Arbeit wird die Selbstidentifikation von Synchronmaschinen und dessen Umrichter beschrieben. Die identifizierten Modelle können direkt zur Berechnung von Regelungsparametern während des Betriebs herangezogen werden (siehe z.B. [2]) oder die Trajektorie der Strom-Referenzsignale (zum energieoptimalen Betrieb). Im Rahmen dieses Projekts wurde nur die Selbstidentifikation umgesetzt weshalb eine Optimierung der Regler-Parameter während des Betriebs aus Zeitgründen nicht umgesetzt werden konnte.

AP 2.1.3 Beschleunigte Charakterisierung

Dieses Thema wird in den Kapiteln 2.1.6 bis 2.1.9 behandelt

2.1.3 AP 3: Modellierung

2.1.3.1 AP 3.2: KI-basierte Modelle

AP 3.2.5 Formale Wissensrepräsentation zur verbesserten Modelbildung

Dieses Thema wird in dem Kapiteln 4, 5 und 6 behandelt.

2.1.4 AP 4: Prototypische Umsetzung

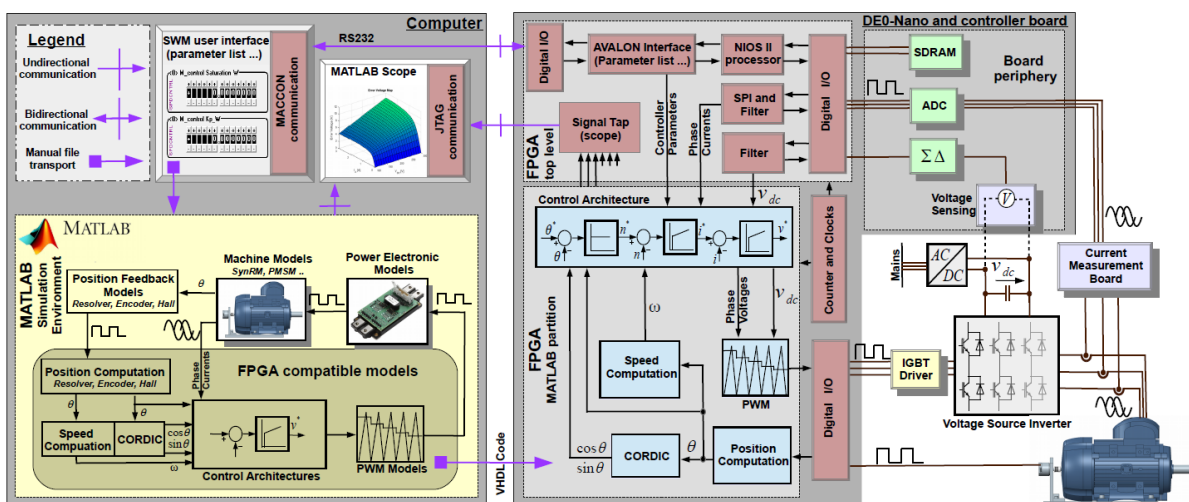
Zur Validierung der im Projekt entwickelten Verfahren auf „embedded“ Systemen wurden im Rahmen von AP 4 mehrere Demonstratoren aufgebaut (AP 4.1). Im letzten Arbeitspaket (AP 4.4) werden die unterschiedlichen KI-basierten Verfahren und Modelle an den Prüfständen validiert.

2.1.4.1 AP 4.1: Aufbau Demonstrator/Prüfstand

Für dieses Projekt wurden drei Motorprüfstände verwendet.

Motorprüfstand 1: Dieser FPGA basierte „rapid-prototyping“ Prüfstand ist im Rahmen der Arbeit [2] entwickelt worden und hat die folgenden Merkmale (abgeändert aus [2]):

Die Entwicklung neuer Regelungsstrategien in einem FPGA (Field Programmable Gate Array) kann wegen langer Kompilierzeiten zeitaufwändig und kompliziert sein. Um diese Situation zu verbessern, bietet das gezeigte System die Möglichkeit, hardwarespezifische Antriebskomponenten und Algorithmen unter realistischen Bedingungen zu simulieren. Dadurch können in frühen Entwicklungsstadien Konstruktionsfehler erkannt und behoben werden und Umweltstörungen können abgemildert werden. Zeigt das Design zuverlässige Simulationsergebnisse, kann entweder die gesamte Steuerungsarchitektur oder einzelne Untermodule über den Matlab HDL coder in die Hardwarebeschreibungssprache (HDL) konvertiert werden. Anschließend wird der generierte Code, mittels ausgewiesener Entwicklungstools wie Intel Quartus Prime (ehemalige Altera Quartus II), in einen FPGA Umgebung eingebettet.



Die Abbildung zeigt auf der linken Seite die Matlab Simulationsumgebung in Gelb.

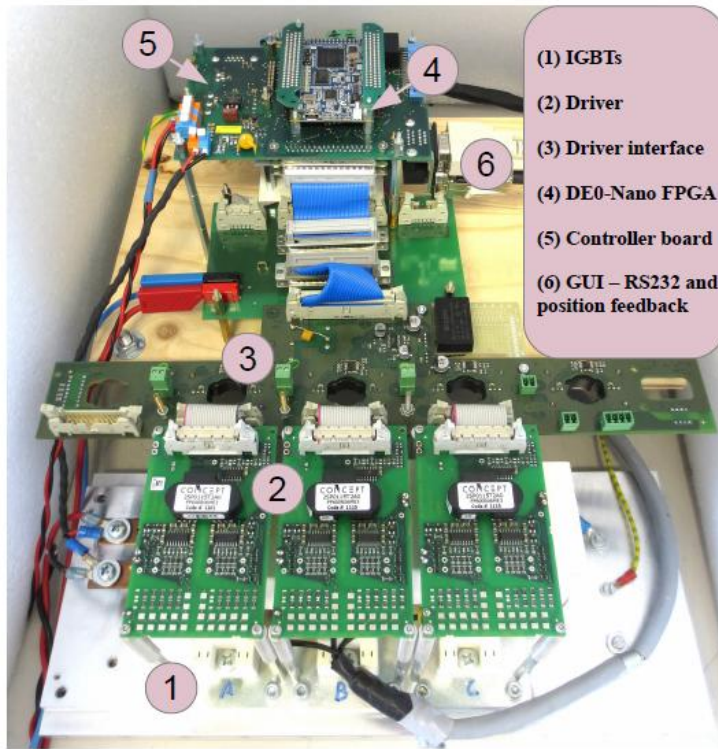
Die Simulation ist unterteilt in Modelle, welche FPGA-kompatibel sind (Softwareteil) und in Modelle welche die Hardwarekomponenten emulieren. Letzteres umfasst Modelle verschiedener Maschinen, leistungselektronische Komponenten und Messtechnik wie Positions-, Strom- und Spannungssensoren sowie FPGA Signalschnittstellen und Umgebungsstörungen. Der Softwarebereich ist in „fixed-point-arithmetic“ ausgelegt und die Eingabe bzw. Ausgangssignale werden auf die Hardwareanforderungen des FPGAs skaliert. Die rechte Seite der Abbildung veranschaulicht die allgemeine Struktur des FPGA DEO-Nanoboard von Intel die sich auf der Steuerplatine befindet und welche die folgenden Komponenten aufweist:

- Zyklon IV FPGA (EP4CE22F17C6N) mit 22.320 Logikelementen, 132 9Bit Multiplikatoren, 4 PLLs und integriertem 50-MHz-Taktoszillator sind Schlüsselemente der Steuerplatine. Auf dem FPGA sind zwei Hauptpartitionen realisiert. Auf der obersten Ebene/Partition sind die Ein- und Ausgangspins (I/O) und dessen Synchronisierung, Phasenregelkreise (PLL) und die Kommunikation mit dem Anwender sowie der externen Peripherie. Darüber hinaus ist die serielle Peripherieschnittstelle (SPI) des analog zu digital Konverters (ADC) realisiert und verschiedene Filter implementiert. Die zweite Partition bestehen aus dem generierten Matlab Code welche die Hauptfunktionalität und Signalerzeugung enthält, die für die Steuerung der Endstufe und Maschinen erforderlich ist.

- Der 32-Bit-NIOS Prozessor ist aus Logikgattern direkt auf dem FPGA aufgebaut. Der NIOS core ist in C programmiert und von Eclipse kompiliert. Die Konfigurationen können in dem 32 MB SDRAM gespeichert werden. Die NIOS-Soft Core kommuniziert über die Avalon Schnittstelle mit der FPGA-Logik. Außerdem ist eine Kommunikation zwischen der FPGA Architektur und einer Benutzeroberfläche (GUI) mittels einer RS232-Verbindung ermöglicht.

- SignalTap II Logicanalyser ist ein Debugging-Tool, das interne Daten erfasst und anzeigt. Signale und Variablen des FPGA-Designs wie Strom, Position und Geschwindigkeit können so veranschaulicht werden. Der Logikanalysator macht eine unidirektionale Kommunikation mit Matlab möglich, welches ein wichtiges Feature für die Visualisierung und Aufzeichnung von Messdaten ist.

Die Hauptkomponenten der Endstufe sind der Wechselrichter mit Treiberplatine (Power Integrations™ 2SP0115T), die elektrischen Maschinen, Spannung und Strommessplatinen und eine DC-Quelle. Drei verschiedenen Synchronmaschinen werden verwendet, um die Zuverlässigkeit und Leistungsfähigkeit der vorgestellten Methoden für verschiedene Maschinentypen zu zeigen. In diesem Zusammenhang wird ein zweistufiger dreiphasiger Spannungsquellen-Wechselrichter verwendet, bestehend aus Bipolartransistoren mit isoliertem Gate (IGBT) von Semikron (SKM50GB12T4). Die Maschinenphasenströme werden digitalisiert durch eine 12-Bit 60 kHz Analog-Digital-Wandlerplatine mit LEM HXS 20-NP Hall-Sensoren.



 Collector emitter voltage

$$V_{CES} = 1200 \text{ V}$$

 Nominal collector current

$$I_{C_{nom}} = 50 \text{ A}$$

 Nominal rise time

$$T_{r_{min}} = 29 \text{ ns}$$

 Nominal fall time

$$T_{f_{min}} = 75 \text{ ns}$$

 Turn on delay

$$T_{d_{on}} = 98 \text{ ns}$$

 Turn off delay

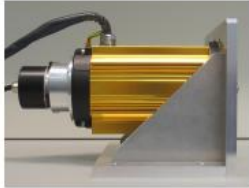

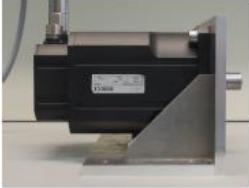
$$T_{d_{off}} = 325 \text{ ns}$$

 Dead-time

$$T_d = 3 \mu\text{s}$$

 Output capacitance

$$C_{oes} = 0.2 \text{ nF}$$

Machine Type				
	IPMSM	SynRM	PMSM	PMSM(load)
Manufacturer	EMP	Stellenbosch	BMP	BMP
Designation	Prototype	Prototype	MS5	MS5
Model	-	-	1050-30-320	1050-30-560
Position Feedback	1000 Line TTL	3600 Line TTL	Hall	Resolver
Phase Current	4.07 A	3.54 A	10.50 A	6.30 A
Torque	2.06 N m	9.60 N m	8.80 N m	8.80 N m
Poles	8	4	6	6
Mechanical Speed	6000 rpm	1500 rpm	3000 rpm	3000 rpm

Motorprüfstand 2: Dieser Mikrocontroller basierte „Rapid-Prototyping“ Prüfstand wurde im Rahmen dieses Projekts entwickelt. Die Recheneinheit ist folgend dargestellt

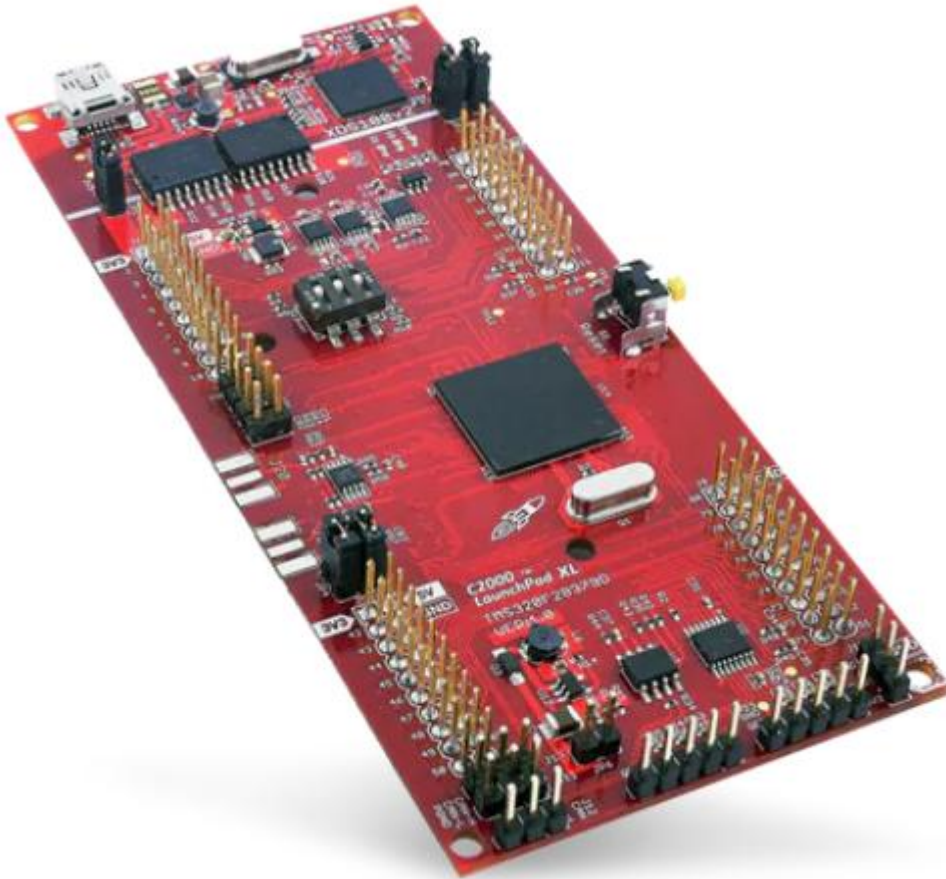


Figure 1 C2000™ LAUNCHXL-F28379D LaunchPad™

und besteht aus dem LAUNCHXL-F28379D evaluation board von TI. Auf diesem board sind zwei Dual Core C2000 Prozessoren verbaut, welcher für real-time Anwendungen geeignet ist. Zudem ermöglicht diese Architektur eine Verbindung zu Matlab/Simulink bzw. erlaubt das direkte Kompilieren von vorher simulierten Simulink Modellen auf die Hardware (vergleichbar zur Testbench 1). Weiterhin ermöglicht dieses Evalboard ein direktes Interface zu dem TI Endstufen: BOOSTXL-DRV8305EVM.

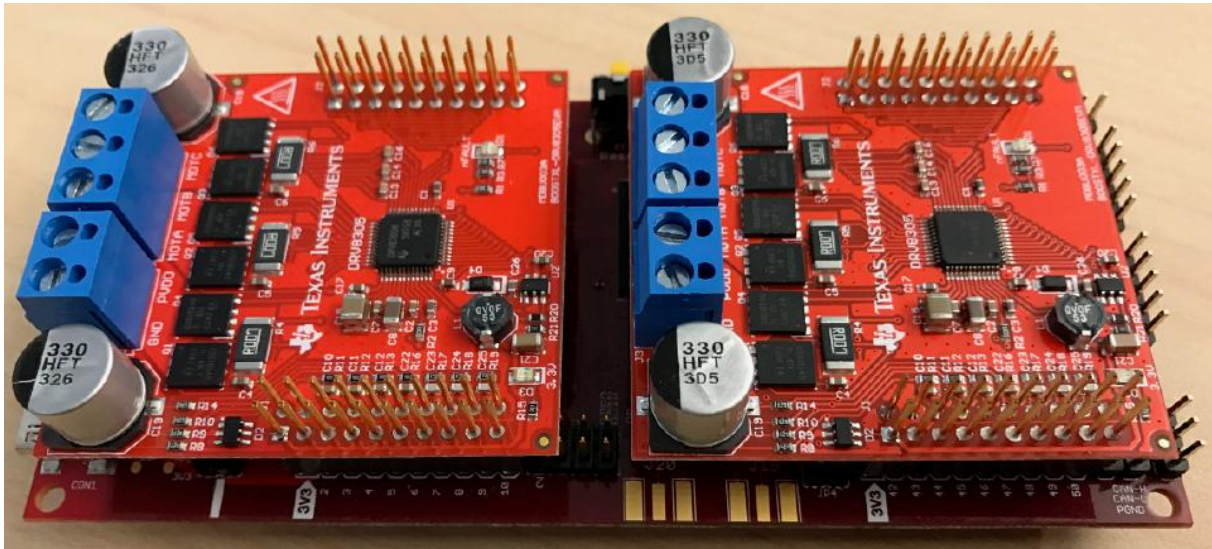


Figure 2 C2000™ LAUNCHXL-F28379D LaunchPad™ mit Endstufe BOOSTXL-DRV8305EVM [3]

Diese Architektur erlaubt einen gleichzeitigen Betrieb von zwei Motoren, weshalb ein Lastmotor und ein zu untersuchender Motor gleichzeitig betrieben werden kann.

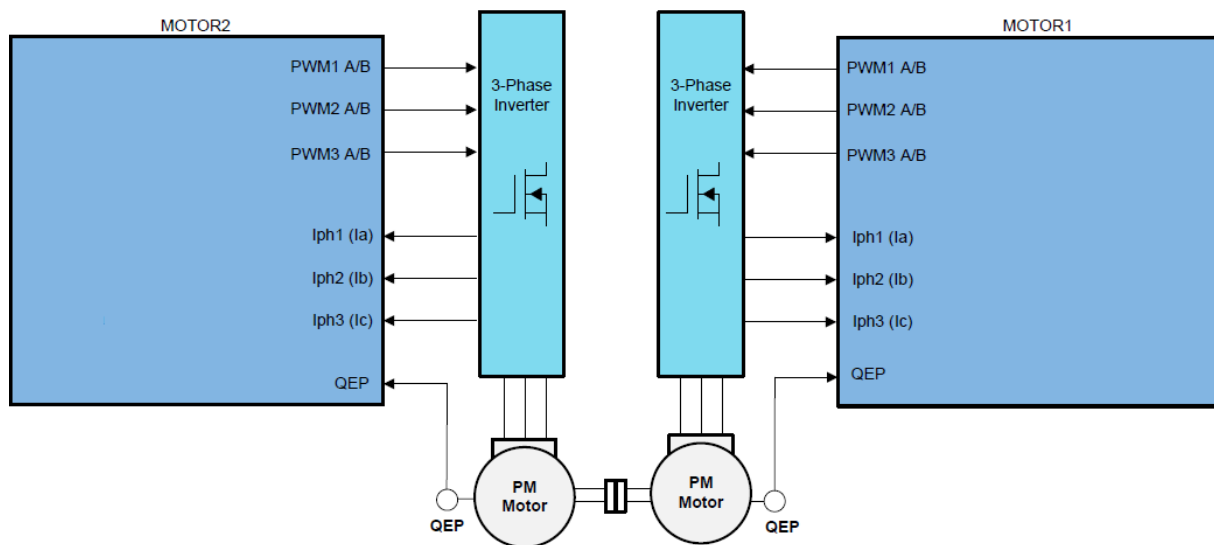


Figure 3 Stark vereinfachte Darstellung der Motoransteuerung mittels des C2000 Evalboards von TI [3] (LAUNCHXL-F28379D)

Die verwendeten Synchronmotoren sind in folgender Abbildung dargestellt. Sie zeigen Wellenabgangsseitig einen TTL-Encoder (Positionssensor). Beide Motoren sind über eine Kupplung miteinander verbunden.



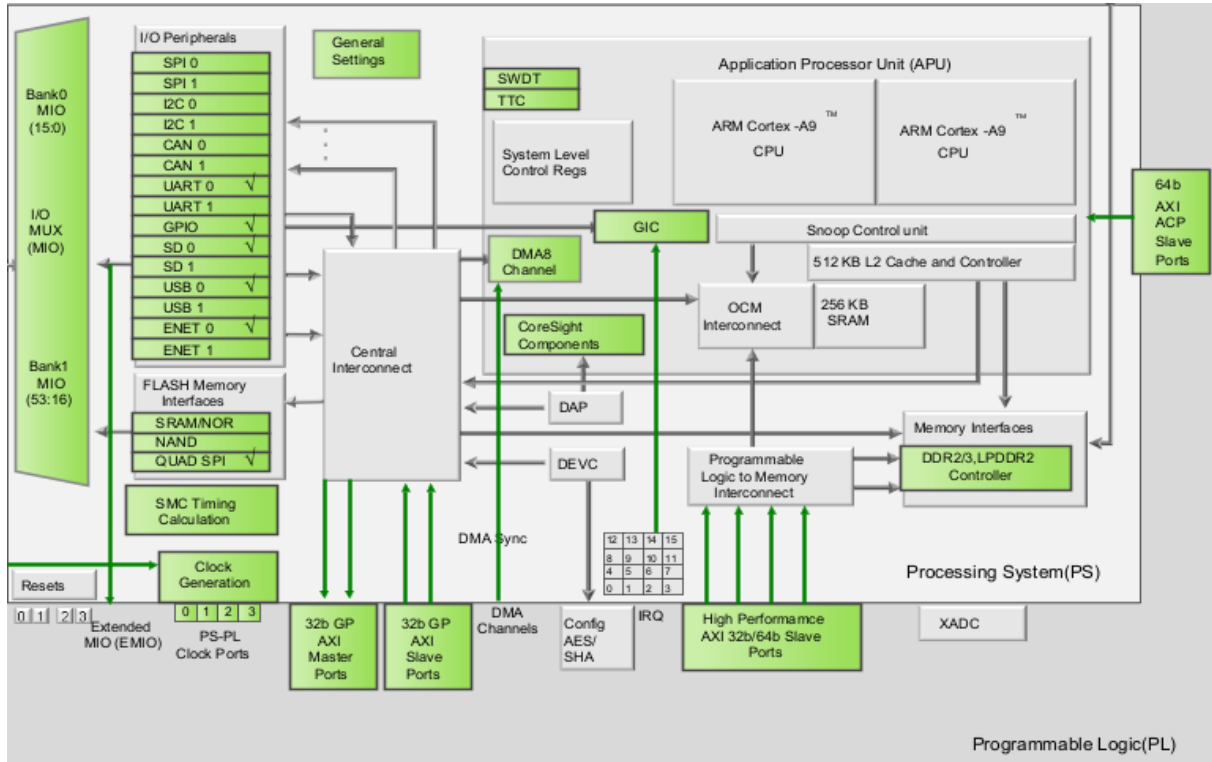
Figure 4 NEMA 23 Motor [3], 48V.

Matlab basierte Software Entwicklungsumgebung für Motorprüfstand 2:

Die für die Motoransteuerung notwendige Software wird wie zuvor erwähnt, mittels Code Konvertierung, auf das C2000 EvalBoard kompiliert. Dabei werden zwei Haupt-Modelle unterschieden. Das Motorregelungsmodell für den zu untersuchenden Motor und für den Lastmotor sowie ein Monitoring Modell welches in „Echtzeit“ Messdaten aufzeichnet und interne Daten des uC manipulieren kann. Das Grundsätzliche Modell zur Ansteuerung des Motors ist im Anhang dargestellt.

Motorprüfstand 3:

Bei diesem Prüfstand wird als Recheneinheit ein Zynq7000 SoC der Firma Xilinx AMD verwendet. Die Architektur ist folgend dargestellt:

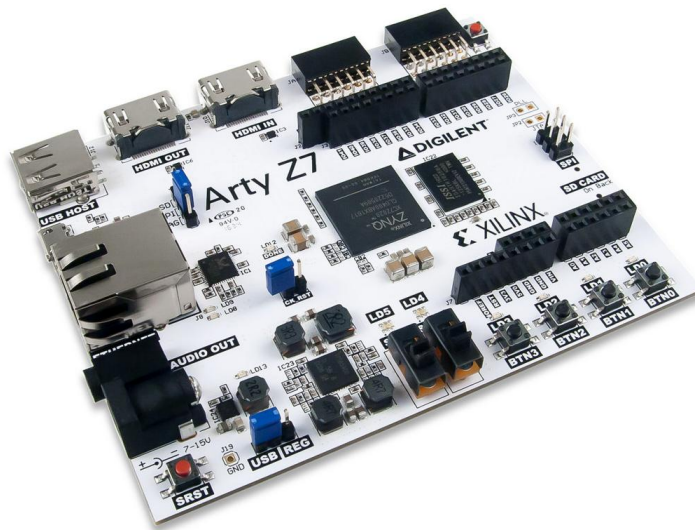


Die Architektur zeigt, dass die Algorithmik auf drei Rechen-Einheiten verteilt wird. Einem FPGA (PL) und zwei ARM Prozessoren (PS).

Als Leistungselektronik wurde folgende Power-Stage „TEC0053-04 EDPS Power Stage“ verwendet (modifiziert mit 6A LEM Strom-Sensoren):



Als Control-Board wurde das Arty Z7-10 Evaluationsboard verwendet:



Welches neben den 650MHz Dual-Core Cortex-A9 Prozessors folgende Leistungsmerkmale hat:

	Arty Z7-10
FPGA part	XC7Z010-1CLG400C
1 MSPS On-chip ADC	Yes
Processor	Dual ARM Cortex A9
Look-up Tables (LUTs)	17,600
Flip-flops	35,200
Block RAM	270 KB
Clock Management Tiles	2
Available Shield I/O	26

2.1.4.2 AP 4.4: Validierung und Evaluation am Prüfstand

Dieses Thema wird in den Kapiteln 2.1.6 bis 2.1.9 behandelt und wird aus übersichtgründen hier nicht weiter aufgeführt.

2.1.5 Motor Control Environment in Matlab/Simulink für Prüfstand 2

Nachfolgend werden beispielhaft ausgewählte Modelle gezeigt, welche für die automatische Code-Generierung des Motorprüfstands zwei verwendet worden sind.

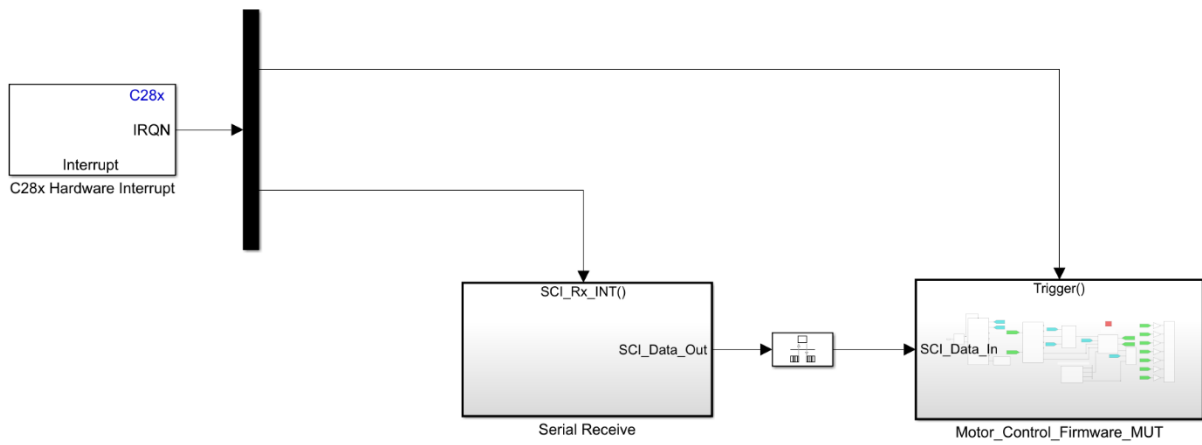


Figure 5 Modell Motoransteuerung – Toplevel

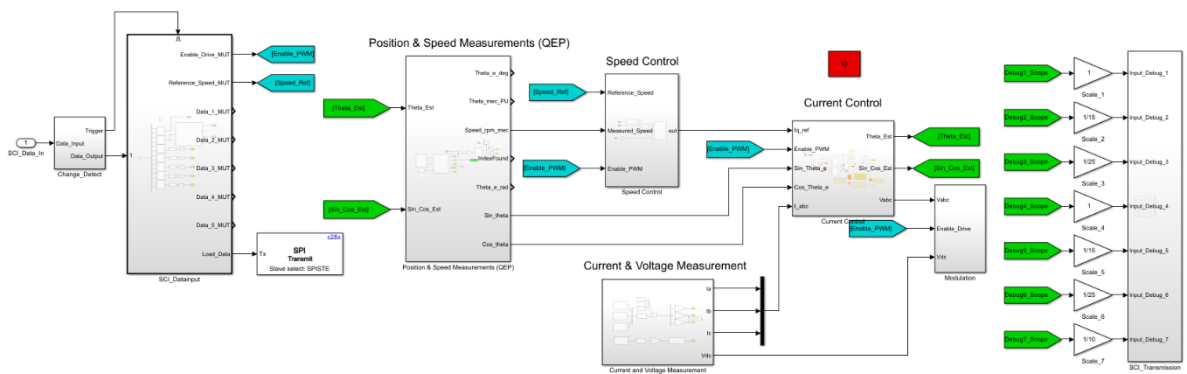


Figure 6 Modell Motoransteuerung – Main Subsystems

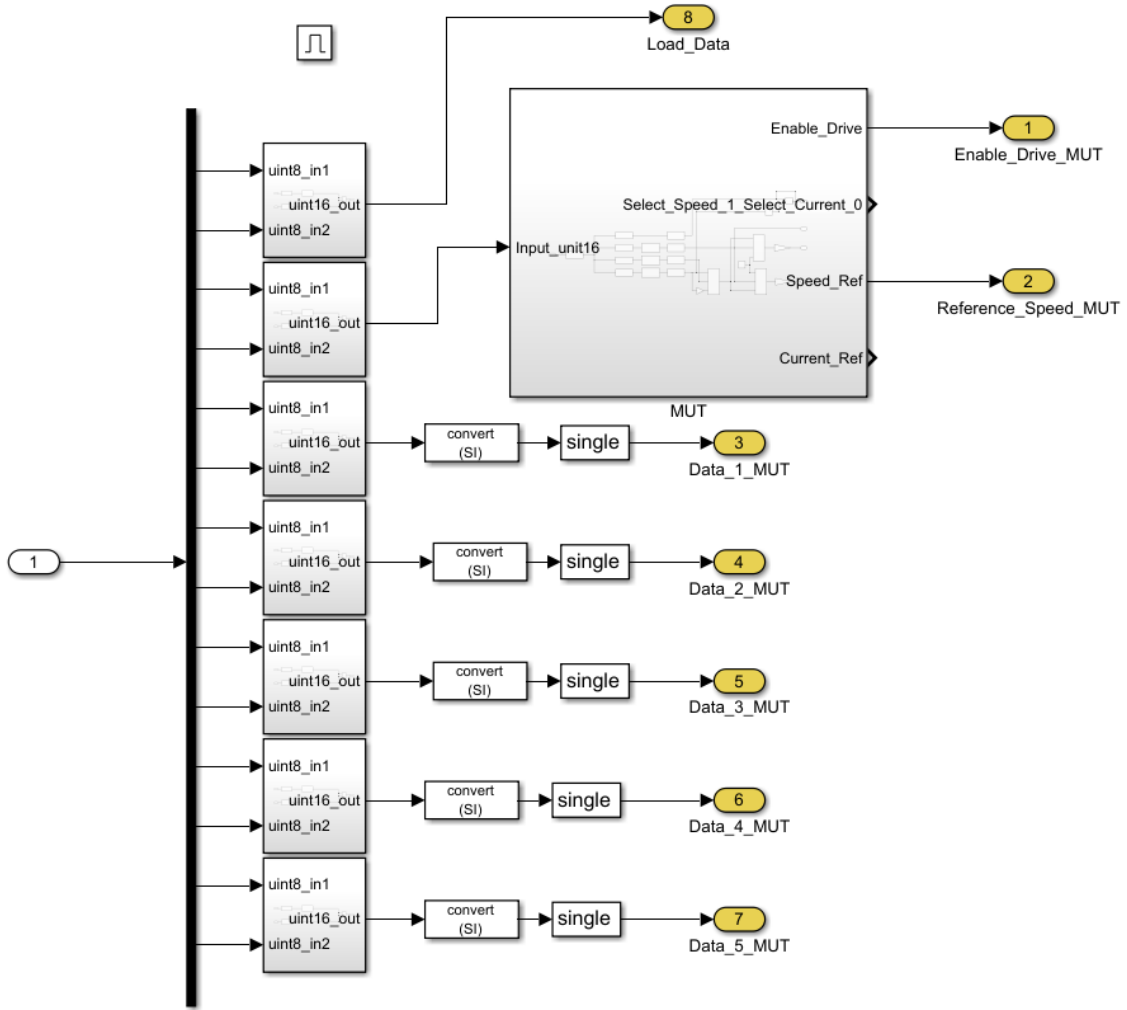


Figure 7 Modell Motoransteuerung – SCI_Datainputs

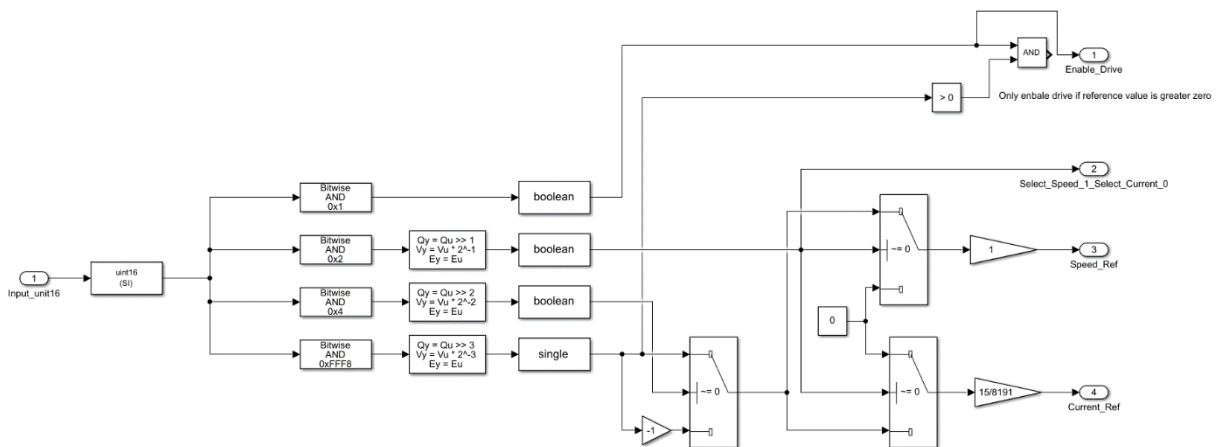


Figure 8 Modell Motoransteuerung – SCI_Datainputs Submodule

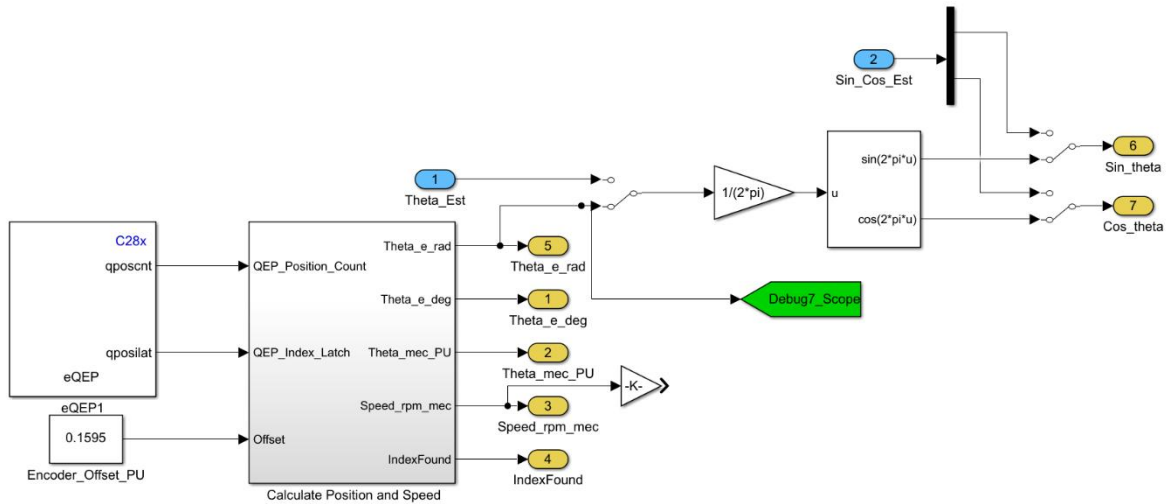


Figure 9 Modell Motoransteuerung – Position & Speed Measurements (QEP)

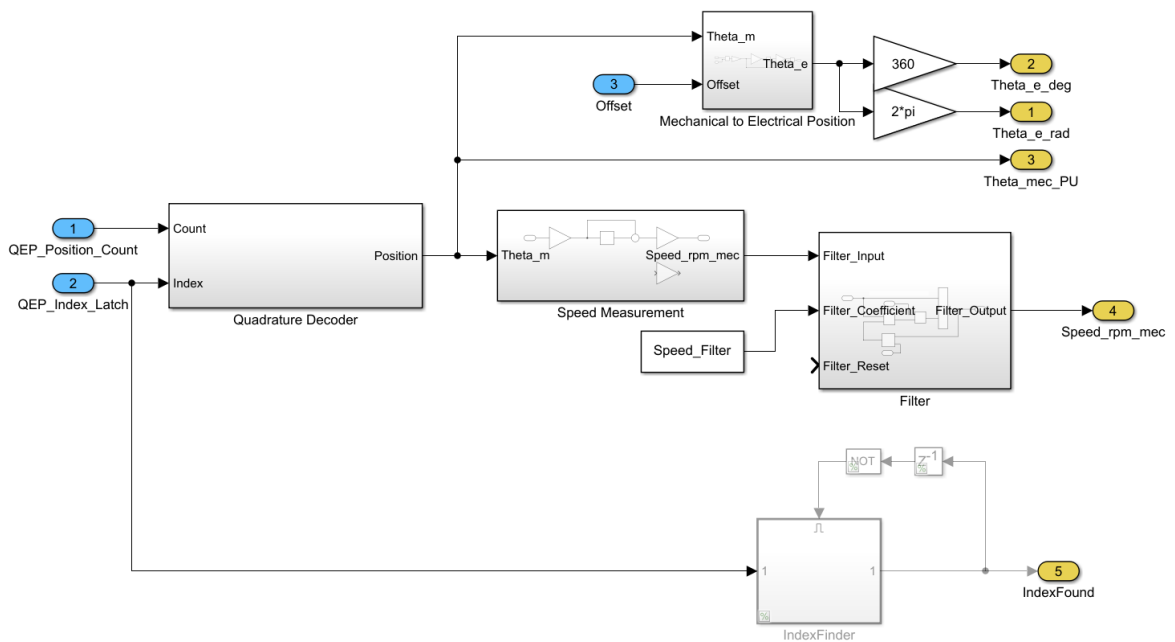


Figure 10 Modell Motoransteuerung – Position & Speed Measurements (QEP) Submodule (weitere Submodule sind aus Übersichtsgründen nicht aufgeführt)

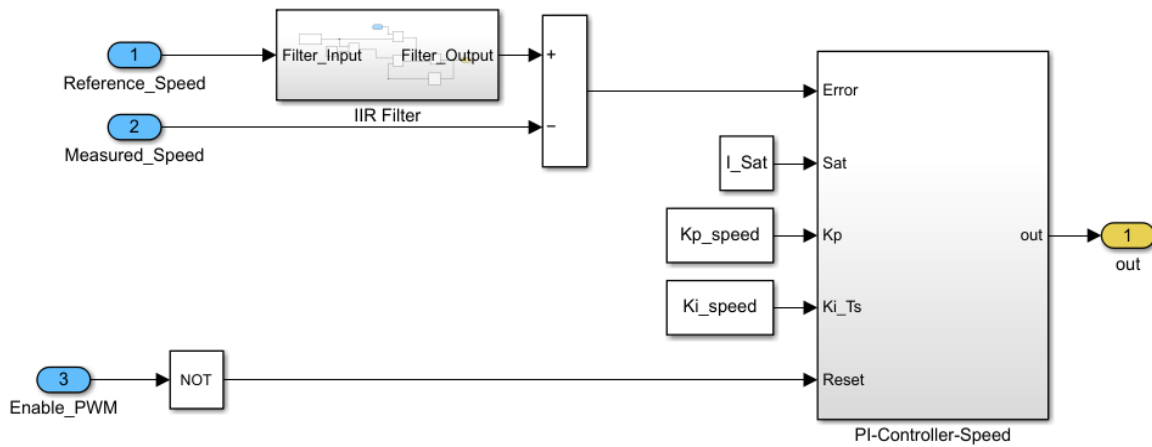


Figure 11 Modell Motoransteuerung – Speed Control (IIR-Filter Submodule ist aus Übersichtsgründen nicht aufgeführt)

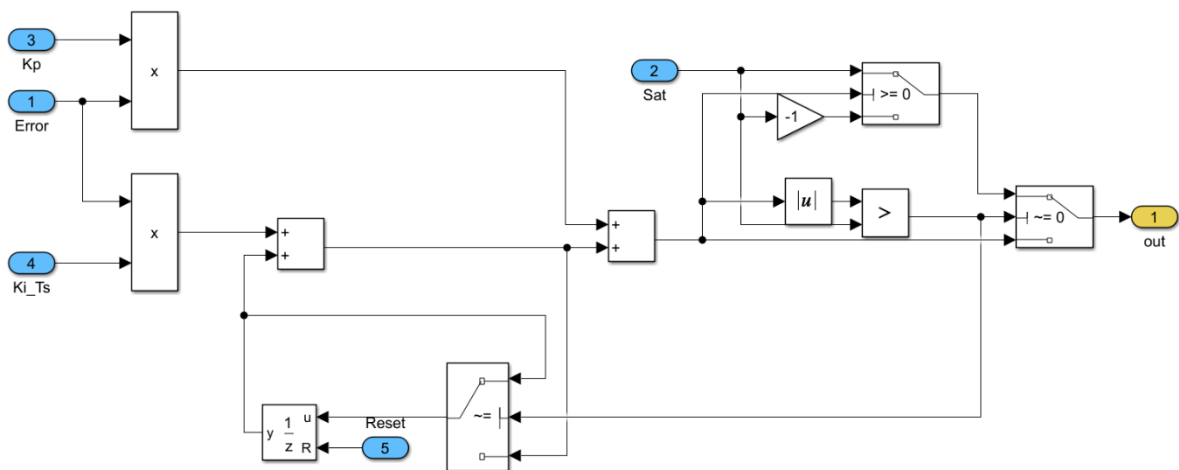


Figure 12 Modell Motoransteuerung – PI Control

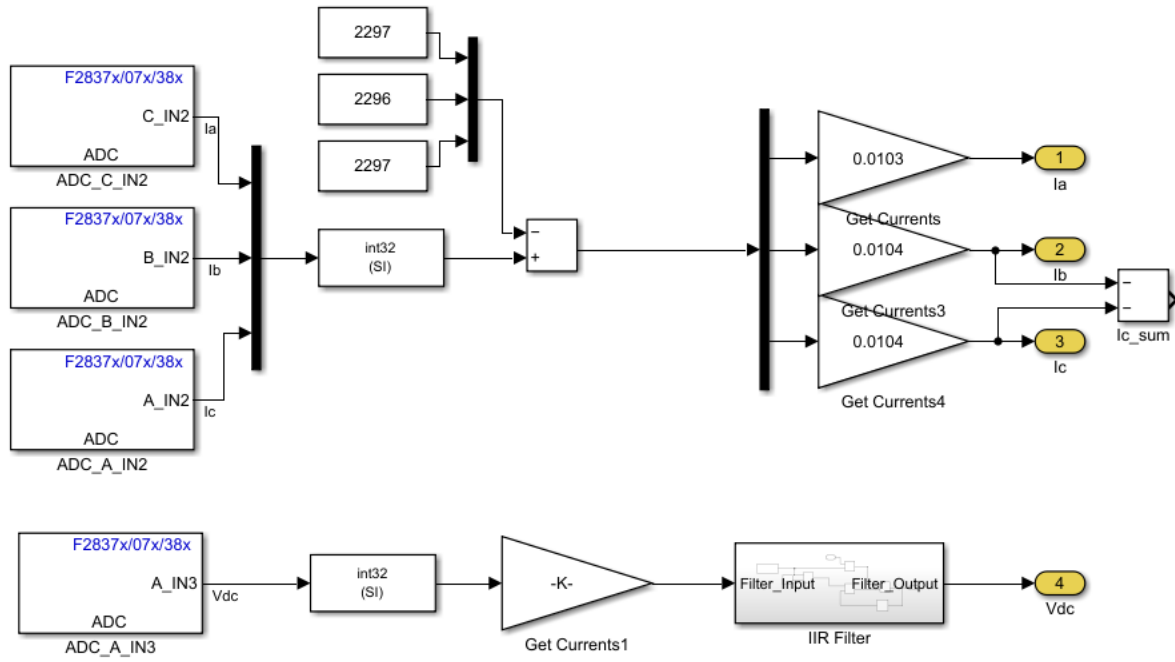


Figure 13 Modell Motoransteuerung – Current and Voltage Measurement

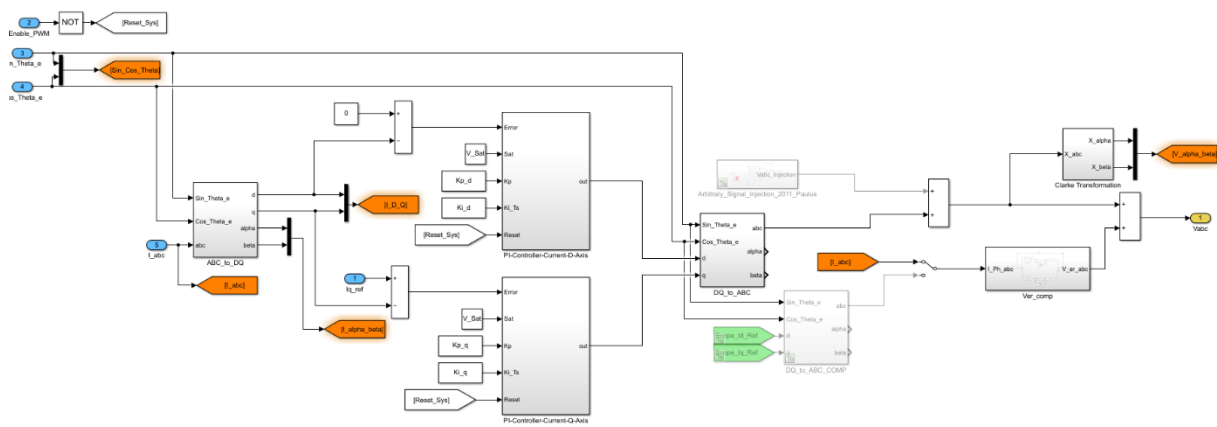


Figure 14 Modell Motoransteuerung – Current Control

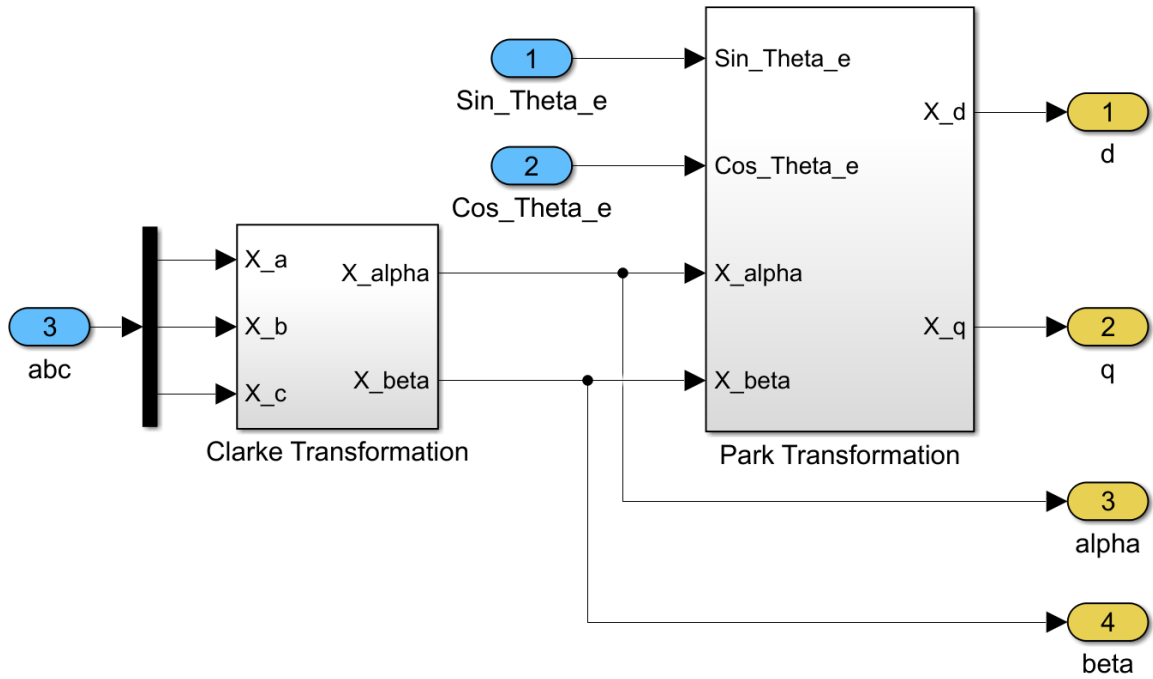


Figure 15 Modell Motoransteuerung – ABC-to-DQ

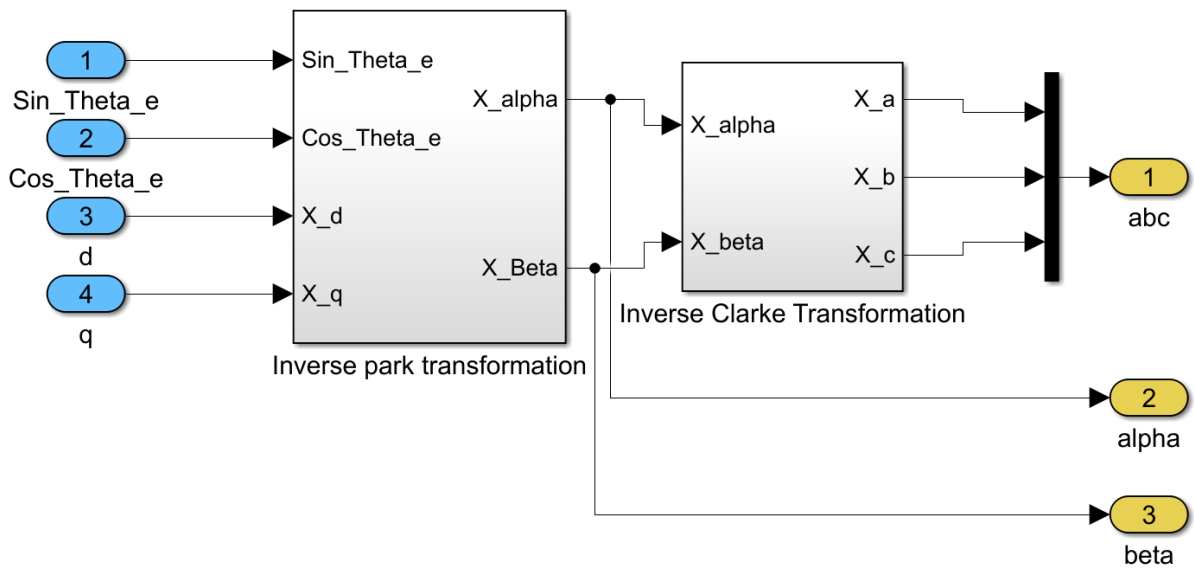


Figure 16 Modell Motoransteuerung –DQ-to-ABC

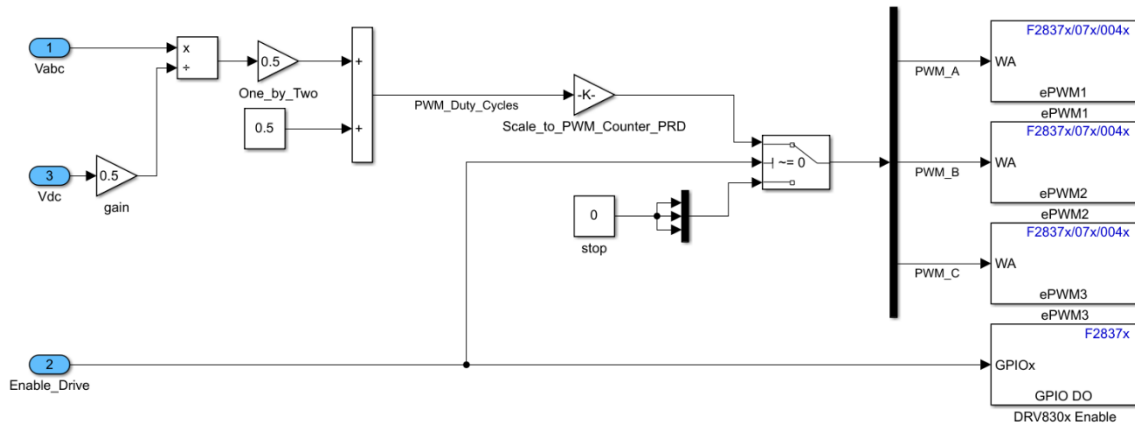


Figure 17 Modell Motoransteuerung – Modulation

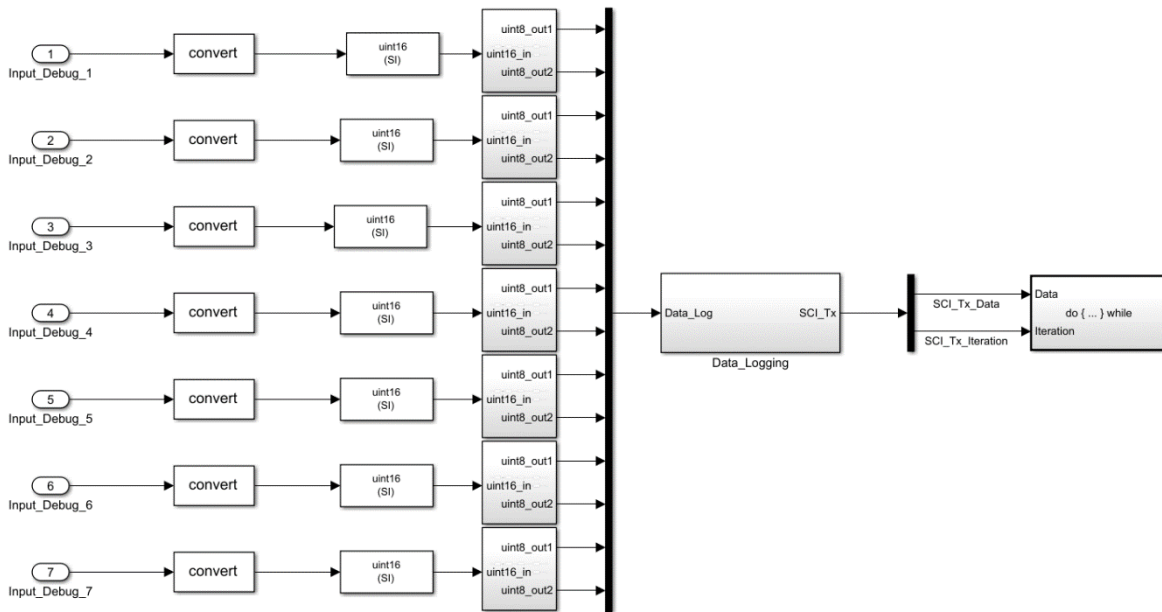


Figure 18 Modell Motoransteuerung – SCI-Transmission

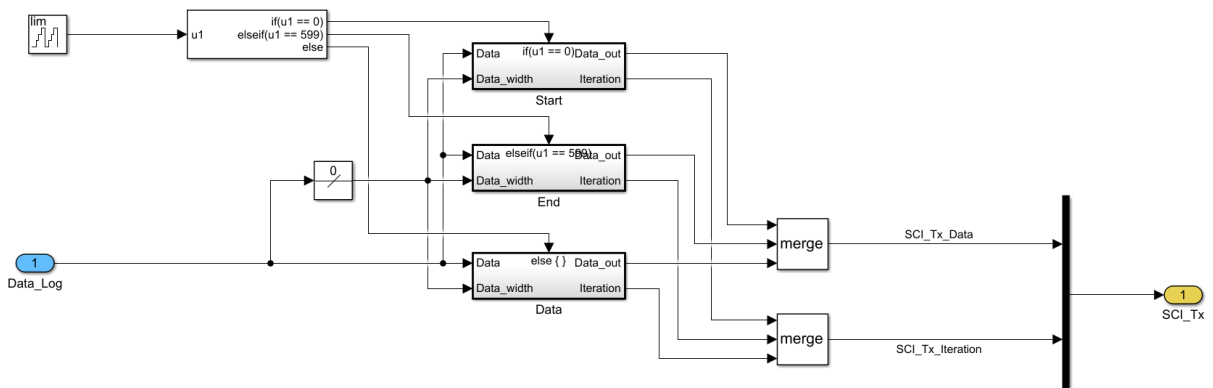


Figure 19 Modell Motoransteuerung – SCI-Transmission Submodule

Das Grundsätzliche Modell zum Monitoring ist folgend abgebildet:

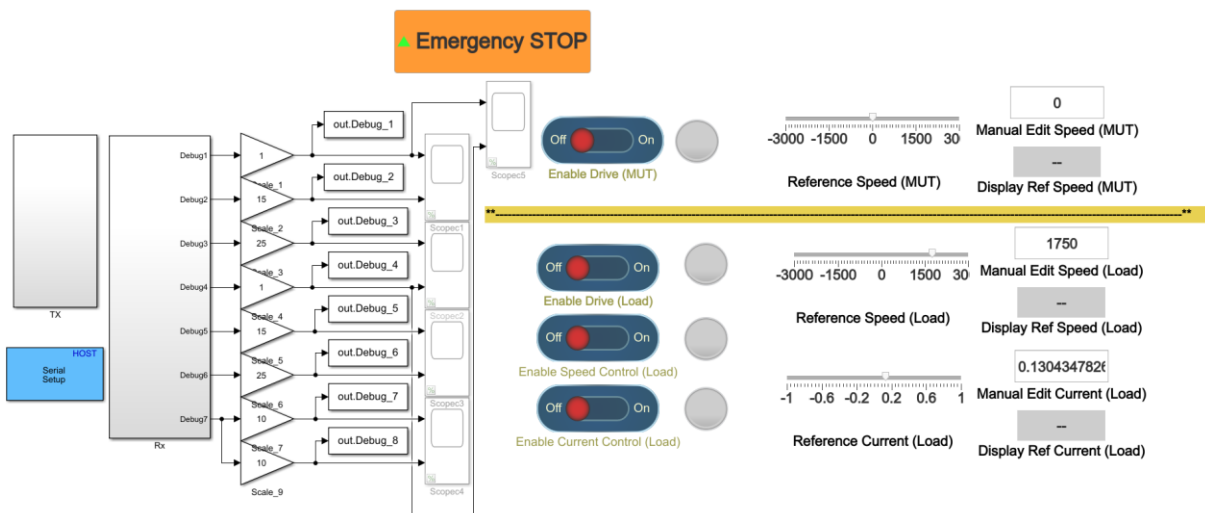


Figure 20 Modell Motor-Monitoring – Toplevel

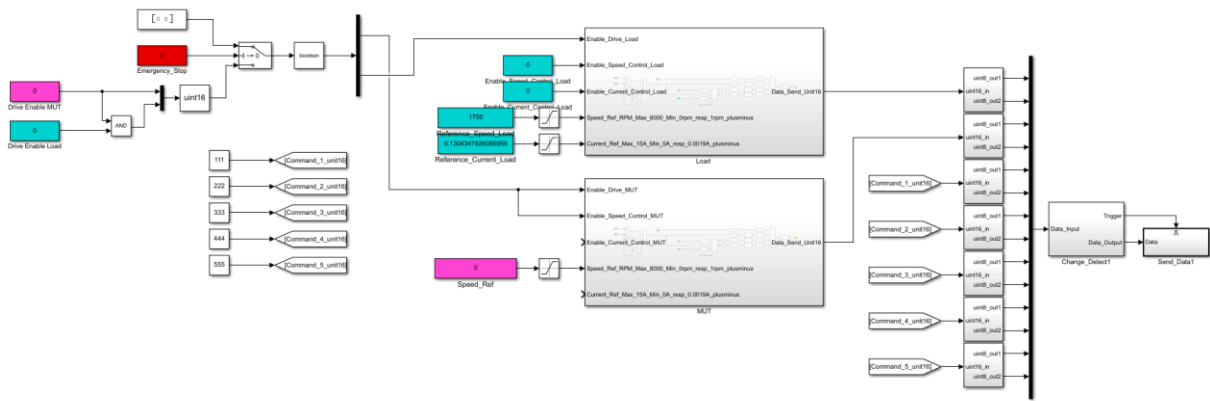


Figure 21 Modell Motor-Monitoring – TX

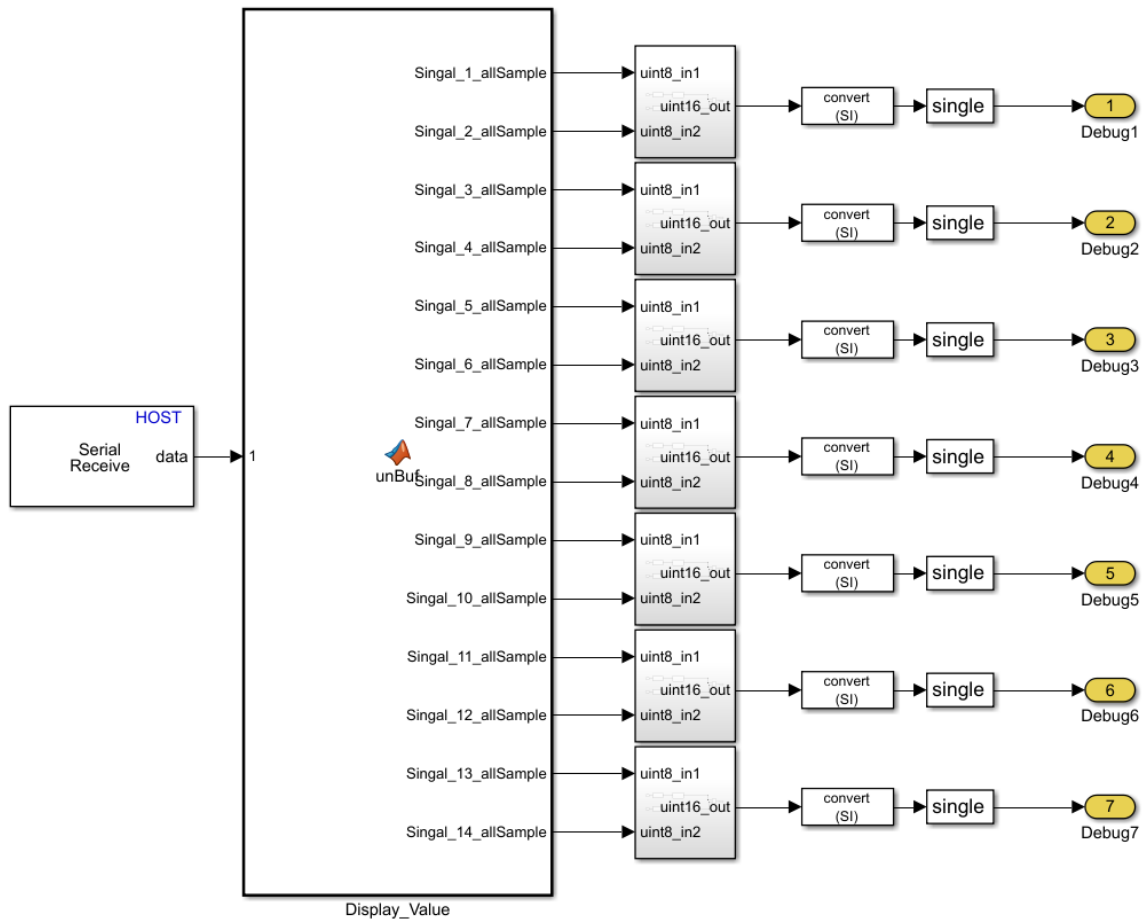


Figure 22 Modell Motor-Monitoring – RX

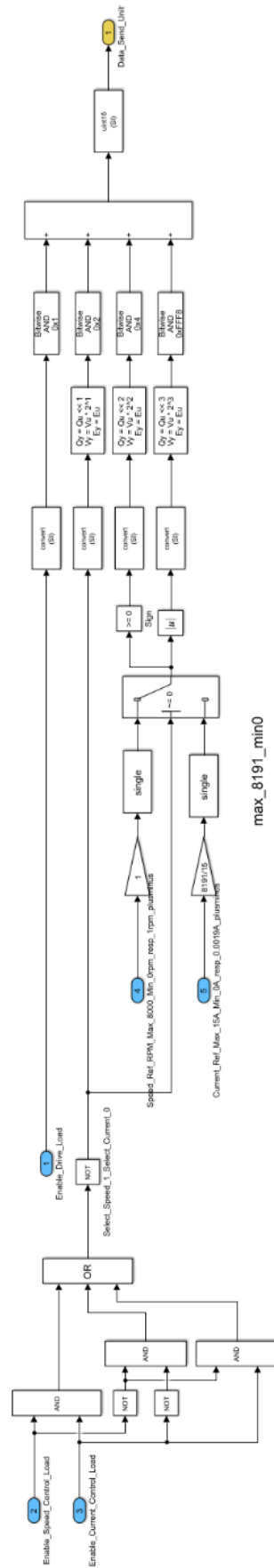


Figure 23 Modell Motor-Monitoring – Load

2.1.6 Selbstinbetriebnahme - Veröffentlichung I

Das vorgeschlagene Identifizierungsverfahren ermöglicht eine gleichzeitige Identifizierung nichtlinearer Ausgangsspannungsabweichungen eines Spannungsquellenwechselrichters (VSIs) sowie des nichtlinearen Modells von Synchronmaschinen welche zur Selbstinbetriebnahme elektrischer Antriebe verwendet werden. Durch die identifizierten Modelle auf der Basis von strukturierten künstlichen neuronalen Netzen (ANNs), ist eine energie-effiziente und dynamische Anpassung der Stromregelung während des Betriebs möglich und die nichtlinearen Spannungsabweichungen des VSI welche durch parasitäre Effekte und Totzeitverzerrungen entstehen, können exakt kompensiert werden. Die Identifizierung wird ohne Positionssensor durchgeführt, während der Rotor mechanisch blockiert ist unter der Verwendung der gemessenen Phasenströme und den Referenzphasenspannungen der Maschine. Experimente für eine Synchronmaschine mit vergrabenen Permanentmagneten (IPMSM) und einer Reluktanz-Synchronmaschine (RSM) zeigen, dass das vorgeschlagene Verfahren dazu in der Lage ist, folgende Charakteristiken zeitgleich zu identifizieren: die stromabhängigen Selbst-Achsen- und Kreuz-Achsen-Flusskarten sowie die Karten der differentiellen Induktivitäten und die nichtlineare VSI-Spannungs-Abweichungen und des Phasenwiderstands. Die vorgeschlagene Methode ist sehr schnell und generisch. Neben den Maschinen-Nennströmen und -Spannungen sind keine Systemkenntnisse erforderlich, welches es für die Selbstinbetriebnahme von allen elektrischen Synchronmaschinen anwendbar macht.

Weitere Details sind in der nachfolgenden Veröffentlichung zu finden („preprint“ Version) [7]:

S. Wiedemann and C. M. Hackl, "Simultaneous Identification of Inverter and Machine Nonlinearities for Self-Commissioning of Electrical Synchronous Machine Drives," in *IEEE Transactions on Energy Conversion*, vol. 38, no. 3, pp. 1767-1780, Sept. 2023, doi: 10.1109/TEC.2023.3263353

Die Veröffentlichung behandeln die Themen der KI-gestützten Modellierung für Synchronmaschinen und Umrichter-Nichtlinearitäten. Diese Modelle können vor der Inbetriebnahme der Maschine durch ein übergeordnetes System selbständig identifiziert werden. Die Modelle bieten die Vorteile, einer adaptiven Anpassung der Stromregler, um eine gleichbleibende Dynamik zu gewährleisten. Weiterhin können die Maschinen, energieoptimaler betrieben werden. Neben der KI-gestützten Modelle wurde auch ein KI-gestütztes Optimierungsverfahren verwendet (Particle-Swarm-Optimisation). Die aufgeführten Methoden wurden durch praktische Messungen an verschiedene Maschinen erfolgreich validiert.

Simultaneous Identification of Inverter and Machine Nonlinearities for Self-Commissioning of Electrical Synchronous Machine Drives

Simon Wiedemann and Christoph M. Hackl *Senior Member, IEEE*

Abstract—The proposed identification method allows for a simultaneous estimation of nonlinear output voltage deviations in voltage source inverters (VSIs) and nonlinear synchronous machine models. Based on the identified characteristics with the help of physically inspired structured artificial neural networks (ANNs), an efficient tuning of the current control system can be performed and the nonlinear voltage deviations caused by parasitic effects and dead-time distortions can be accurately compensated for. The identification is performed without position sensor while the rotor is mechanically locked by utilising measured phase currents and reference machine voltages only. Experiments for an interior permanent magnet synchronous machine (IPMSM) and a reluctance synchronous machine (RSM) show that the proposed method is capable of identifying the current dependent self-axis and cross-axis flux linkages, differential inductances and the nonlinear VSI voltage deviations as well as the phase resistance at the same time. The proposed method is fast and generic. Besides the rated machine current, voltage and frequency, no prior system knowledge is required making it applicable for the self-commissioning of any electrical synchronous machine drive.

Index Terms—Identification, Self-Commissioning, Auto-tuning, Synchronous Machine, Inverter Dead-Time, Machine Characterisation, Flux Linkage Map, Machine Model, Artificial Neural Network, Encoderless.

NOTATION

\mathbb{N}, \mathbb{R} : natural, real numbers; $x := (x_1, \dots, x_n)^\top \in \mathbb{R}^n$: column vector, $n \in \mathbb{N}$ where “ \top ” and “ $:=$ ” mean “transposed” and “is defined as”, resp.; $a^\top b := a_1 b_1 + \dots + a_n b_n$: scalar product of vectors a, b ; $\|x\| := \sqrt{x^\top x} = \sqrt{x_1^2 + \dots + x_n^2}$: Euclidean norm of x ; $X \in \mathbb{R}^{n \times n}$: matrix (n rows & columns); X^{-1} : inverse of X (if exists). *Remark: All physical quantities are introduced and explained in the text.*

I. INTRODUCTION

Modern control methods and observer architectures of electrical machines rely mostly on accurate information of drive characteristics. Identified parameters of the linear machine

Simon Wiedemann is with the Research and Development Department at MACCON GmbH, Aschauer Str. 21, Munich 81549, Germany (e-mail: s.wiedemann@maccon.de).

Christoph M. Hackl is with the Institute for Sustainable Energy Systems (ISES), Hochschule München (HM) University of Applied Sciences, Munich 80335, Germany (e-mail: christoph.hackl@hm.edu).

This work was supported by the Federal Ministry for “Economic Affairs and Climate Action” (19I21030H - KIRA).

model such as the phase resistances and inductances can be utilised to tune the current controllers [1]–[3] or observers and to compute the optimal reference currents for an energy efficient operation [2], [4]. This tuning can be further improved by the identification and utilisation of nonlinear flux linkage maps usually expressed in the synchronously rotating (d, q)-reference frame [2], [4]–[8] which allow to compute the differential inductances (in the following, the superscript d or q stands for the d - or q -axis, resp.). The automatic identification and tuning of the machine drive can be considered as self-commissioning/auto-tuning and is usually performed before machine operation via offline identification [1], [2], [9].

Self-commissioning algorithms heavily decrease the time required to characterise and tune an electrical machine drive compared to a manual process which, due to a lack of expertise and time of the control engineers, might lead to imprecise results and poor control dynamics [2] as well as an energy inefficient operation [9]. The importance of an automatic commissioning becomes obvious if one considers that around 53 % of the globally generated energy is consumed by electrical machines [10]. Despite several improvements on energy efficient control methods in the last 30 years [4], only little effort seems to be taken for a rigorous adoption of these technologies within the industry. A comprehensive review on parameter identification and self-commissioning of ac drives can be found in [1].

Furthermore, not only control architectures but also identification algorithms are prone to perform inaccurately if inverter nonlinear effects expressed through output voltage errors of the voltage source inverter are not accurately compensated for [11]–[13]. Therefore, the characterisation of these VSI voltage errors must be (and usually is) performed before (and separately from) the machine identification process. Moreover, the state of the art voltage error identification methods [11], [12] rely on already tuned current controllers which requires knowledge and, therefore, leads to an iterative and repetitive commissioning process which is still not most efficient.

Magnetic saturation is usually modelled either by the stator flux linkage maps $\psi_s^d(i_s^d, i_s^q)$ and $\psi_s^q(i_s^d, i_s^q)$ depended on the stator d and q currents or vice versa, i.e. $i_s^d(\psi_s^d, \psi_s^q)$ and $i_s^q(\psi_s^d, \psi_s^q)$. The former is usually more complex to model with e.g. neural networks [5], [14], [15], piece-wise nonlinear functions [16], or nonlinear models [17], [18]. The latter can be expressed by polynomials [18]–[20] making it simpler to be identified. Typically, three offline identification methods

can be applied to estimate the saturation characteristics:

Constant speed methods (CSM) [6], [19]–[21] are most accurate but require much time and extensive measurement equipment including a load machine and take rather longer. Dynamic testing methods (DTM) [6], [22] perform the characterisation faster compared to CSM and need less equipment such as a load machine but are limited in the range of characterisation currents due to very fast rotor accelerations during the identification. Standstill methods (SSM) [15], [16], [23], [24] allow for the fastest characterisation within a few seconds or less. However, estimation results may be negatively affected if the rotor starts to move from the rest point due to non-negligible torques produced by the injected currents. In this case, the rotor needs to be locked mechanically. Moreover, for the aforementioned SSMs, the flux linkage maps are estimated by an open loop integrator which may lead to a drift in the estimation results due to measurement inaccuracies and/or noise. For all of the methods above, an identification error exists due to voltage deviations arising from uncompensated VSI nonlinearities or uncertainties in the stator resistance. However, it depends on the severity of both effects if one needs to implement a compensation or not [9]. In particular, SSMs seem to be more robust in this regard [24]. For industrial drives (as experience indicates), it is always beneficial if not even necessary to identify and compensate for VSI nonlinearities to improve the identification results.

Contributions of this paper are: (C1) A simple and very fast offline standstill identification method which can be performed position sensorless (encoderless) where the machine under test (MUT) is excited by low frequency voltage signals resulting in phase current responses in d - or q -axis direction; and (C2) An effective post-processing nonlinear least-squares algorithm which utilises the measured currents and the stator reference voltages only for the simultaneous identification of VSI voltage error characteristics *and* nonlinear synchronous machine models considering self-axis *and* cross-axis flux linkages *and* all differential inductances as well as the stator resistance. VSI voltage error and flux linkages are approximated by structured ANNs which exploit intrinsic physical properties of the approximated quantities. The proposed method has the following advantages compared to state-of-the-art machine identification methods: (A1) The a priori compensation of VSI nonlinearities is *not* required in contrast to CSM, DTM and SSM; (A2) the phase resistance does *not* need to be known a priori in contrast to SSM; (A3) *no* current controllers are needed in contrast to CSM and DTM; (A4) *no* load machine (in contrast to CSM) or flywheel (in contrast to DTM) is necessary; (A5) the identification is performed as fast as with SSM and, therefore, is much faster than for CSM and DTM; (A6) the phase resistance is identified at the same time which is not the case for CSM, DTM and SSM; and (A7) compared to the competitive SSM, no open loop integration effects can occur, which makes this method more robust to measurement inaccuracies and/or noise. Furthermore, the proposed method has the following advantages compared to state-of-the-art VSI identification methods (see e.g. [11], [12]): (A8) the machine phase resistance and equivalent VSI resistance do not have to be known a priori; (A9) no current

controllers and special current reference signals are necessary; and (A10) the identification is performed much faster which leads to a significantly reduced machine heating (a general disadvantage of most available methods). Nevertheless, it is recommended that the rotor is locked during the identification of the proposed method (at least for a q -axis characterisation) which can be seen as a drawback of the proposed method.

The remainder of the paper is organized as follows: Section II introduces the VSI voltage deviations and its implication on drive performance and introduces a modelling technique suitable for compensation and identification. Section III shows how the current dynamics are modelled (in general and at standstill) while nonlinear magnetic effects (such as saturation) of synchronous machines are considered. Section IV describes the overall identification process including system excitation, post-processing and how the electric drive characteristics are identified simultaneously. Finally, Section V validates the proposed identification and modelling techniques via measurements for an IPMSM and a RSM. Section VI concludes the paper.

II. VOLTAGE SOURCE INVERTER NONLINEARITIES

Besides their modulated approximation of sinusoidal voltages, VSIs induce additional errors in their produced output voltages which result in deviations compared to the desired phase reference voltage $u_{s,\text{ref}}^p \in \{u_{s,\text{ref}}^a, u_{s,\text{ref}}^b, u_{s,\text{ref}}^c\}$ obtained from the control system. These deviations in the actually applied phase voltage $u_s^p \in \{u_s^a, u_s^b, u_s^c\}$ to the electrical machine will lead to (a) performance deterioration of the current controllers and/or outer control loops, (b) harmonics in the machine, torque and currents and (c) deteriorated estimation results for observers or identification architectures. The reasons for voltage deviations are mainly: voltage transients of the power semiconductors (diodes and transistors) [9], [13], dead-times between consecutive switching events [25], zero current clamping effects [26] and nonlinear voltage drops over the semiconductors through on-state and resistive voltages [13], [27]. Furthermore, the voltage deviations nonlinearly depend on several system signals and parasitic effects such as the: DC-link voltage u_{dc} , switching frequency $f_{sw} = 1/T_{sw}$ of the power transistors, stator phase current $i_s^p \in \{i_s^a, i_s^b, i_s^c\}$ and its respective sign $\text{sign}(i_s^p)$ [13], [25], VSI output capacitance C_{out} [9], [13], temperature and ageing effects [28].

A. Modeling of voltage deviations due to VSI nonlinearities

The typical phase voltage deviations Δu_s^p of some phase $p \in \{a, b, c\}$ are illustrated in Fig. 1. Two main regions can be distinguished. In the high-current region, Δu_s^p becomes approximately a constant whereas a distinctive current dependency in the shape of a sigmoid function is present in the low-current region. In the high-current region, the magnitude of the voltage deviation can be computed as [25], [29]

$$M_{\Delta u_s} = \left[(u_{dc} - u_{\text{sat}} + u_{d,\text{on}}) \frac{T_a + T_{\text{on}} - T_{\text{off}}}{T_{sw}} + \frac{u_{\text{sat}} + u_{d,\text{on}}}{2} + \frac{(u_{\text{sat}} - u_{d,\text{on}}) u_{s,\text{ref}}^p}{u_{dc}} \right], \quad (1)$$

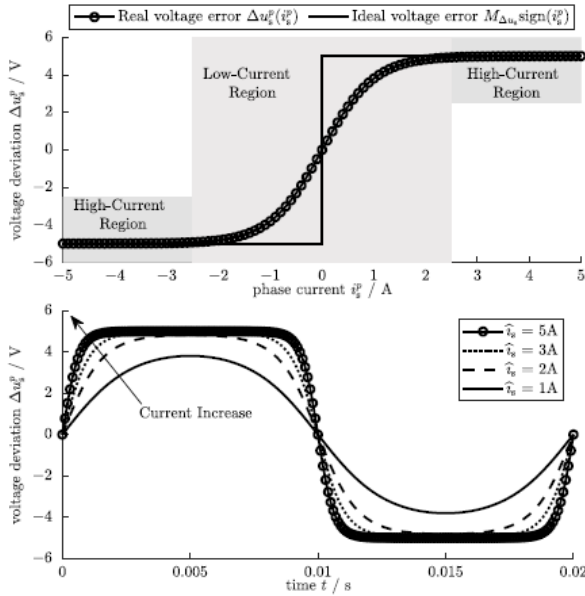


Figure 1: Ideal vs. real voltage deviation over i_s^p (top); real voltage deviation $\Delta u_s^p(i_s^p)$ over one period of a sinusoidal phase current i_s^p with different amplitudes \hat{i}_s (bottom).

which depends on dc-link voltage u_{dc} , transistor threshold voltage u_{sat} and diode threshold voltage $u_{d, on}$ and phase reference voltage $u_{s, ref}^p$, dead time T_d and switching period T_{sw} where on and off times are defined as

$$T_{on} = T_{d, on} + \frac{T_{f, min}}{2} \quad \text{and} \quad T_{off} = T_{d, off} + \frac{T_{r, min}}{2}, \quad (2)$$

i.e. the times of the negative and positive voltage areas as a result of the on and off delay times $T_{d, on}$ and $T_{d, off}$ and the minimal rise and fall times $T_{r, min}$ and $T_{f, min}$, respectively.

The voltage error model becomes more accurate if one considers that T_{off} in (2) is a nonlinear function

$$T_{off} = f(C_{out}, u_{dc}, i_s^p, T_{d, off}, T_{r, min, max}, T_{f, min}), \quad (3)$$

where C_{out} is a composition of different equivalent VSI leg capacitors and the output capacitance of the half bridge (for details see [9], [13]). In conclusion, the voltage deviation Δu_s^p due to the VSI nonlinearities affects each phase voltage, i.e.

$$\forall p \in \{a, b, c\}: \quad u_s^p = u_{s, ref}^p - \Delta u_s^p(M_{\Delta u_s}, i_s^p, \dots). \quad (4)$$

Remark 1. Due to the switching nature of the VSI, all voltages in (4) actually represent averaged voltages over one switching period T_{sw} , i.e., $\bar{u}(t) := \int_{t-T_{sw}}^t u(\tau) d\tau$.

B. ANN-based approximation of VSI voltage deviations

For self-commissioning, an accurate representation of Δu_s^p is required within the control system in order to compensate for the VSI nonlinearities. Using a physical model composed of (1), (2), (3) and (4) is crucial for the understanding of the error source but impractical for a real-time compensation and implementation as most parameters and quantities in (1), (2), (3) and (4) are neither (exactly) known nor measured.

1) *Approximation in (a, b, c)-reference frame:* Therefore, a simple feed-forward artificial neural network (ANN) as shown in Fig. 2 is proposed. This allows to model $\Delta u_s^p = f_{ann}^{vsi}(i_s^p, u_{dc}, T_{sw}, \dots)$ as a nonlinear function of one or more inputs. This makes it very flexible, accurate and efficient in representing experimental data analytically within a digital environment utilising a minimum amount of model parameters and system storage compared to look-up tables (LUTs) [11] or trapezoidal models [30]. Recalling Fig. 1 reveals that one input – the phase current i_s^p – is sufficient. Therefore,

$$\forall p \in \{a, b, c\}: \quad \Delta u_s^p \approx \Delta \hat{u}_s^p(i_s^p) := f_{ann}^{vsi}(i_s^p) \quad (5)$$

is chosen as approximation $\Delta \hat{u}_s^p(i_s^p)$ of each phase VSI nonlinearity for self-commissioning (assuming a constant dc-link voltage). The proposed ANN architecture is illustrated in Fig. 2 and consists of one input layer (Layer 0) and one output layer (Layer 2) each with identity activation function $\Phi_0(y) = \Phi_2(y) = \Phi_{id}(y) = y$ (cf. [31]) and one hidden layer (Layer 1) with two neurons utilizing the soft-sign activation function $\Phi_{1,1}(y) = \Phi_{1,2}(y) = \Phi_{SoSi}(y) = \frac{y}{1+|y|}$ (cf. [32]).

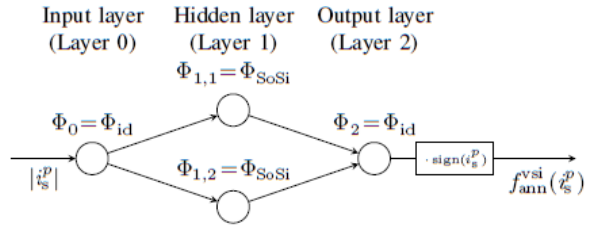


Figure 2: ANN for approximation of VSI voltage deviation Δu_s^p of phase $p \in \{a, b, c\}$ by $\Delta \hat{u}_s^p := f_{ann}^{vsi}(i_s^p)$.

With the ANN architecture as in Fig. 2, the VSI voltage deviation Δu_s^p of phase $p \in \{a, b, c\}$ will be approximated by

$$\Delta \hat{u}_s^p(i_s^p) := f_{ann}^{vsi}(i_s^p) = \left(\frac{w_{2,1}^{vsi} (w_{1,1}^{vsi} |i_s^p| + b_{1,1}^{vsi})}{1 + |w_{1,1}^{vsi} |i_s^p| + b_{1,1}^{vsi}|} + \frac{w_{2,2}^{vsi} (w_{1,2}^{vsi} |i_s^p| + b_{1,2}^{vsi})}{1 + |w_{1,2}^{vsi} |i_s^p| + b_{1,2}^{vsi}|} \right) \text{sign}(i_s^p) \quad (6)$$

with overall six ANN parameters¹, i.e. $w_1^{vsi} := (w_{1,1}^{vsi}, w_{1,2}^{vsi})^T$, $b_1^{vsi} := (b_{1,1}^{vsi}, b_{1,2}^{vsi})^T$ of Layer 1 and $w_2^{vsi} := (w_{2,1}^{vsi}, w_{2,2}^{vsi})^T$ of Layer 2 which need to be identified.

Remark 2 (ANN-training for one phase). *It is sufficient to train the ANN as in (6) for the VSI voltage deviation $\Delta u_s^p(i_s^p)$ for one phase only (e.g., $p = a$). For real-time compensation, it must be implemented for each phase $p \in \{a, b, c\}$ separately and fed by the respective phase current i_s^p .*

Remark 3 (Dependency on dc-link voltage). *In general, the VSI nonlinearity may also depend on the dc-link voltage u_{dc} . To consider a varying u_{dc} , the ANN structure in (6) has to be extended by replacing the terms $x_1 := w_{1,1}^{vsi} |i_s^p| + b_{1,1}^{vsi}$ and $x_2 := w_{1,2}^{vsi} |i_s^p| + b_{1,2}^{vsi}$ by $x_1 := w_{1,1}^{vsi} |i_s^p| + w_{1,3}^{vsi} u_{dc} + b_{1,1}^{vsi}$ and*

¹Explanation of ANN notation (cf. [31]): Weight $w_{i,j}^z$ and bias $b_{i,j}^z$ are identified by their subscripts i and j indicating layer and neuron, respectively, and by their superscript z indicating the system to be estimated by the ANN.

$x_2 := w_{1,2}^{\text{vsi}} |i_s^p| + w_{1,4}^{\text{vsi}} u_{\text{dc}} + b_{1,2}^{\text{vsi}}$, respectively. Then, the VSI-ANN depends not only on the phase current i_s^p but also on the dc-link voltage u_{dc} and has two more weights $w_{1,3}^{\text{vsi}}$ and $w_{1,4}^{\text{vsi}}$.

2) Approximation in (d, q) -reference frame (required for later ANN training): In the (d, q) -reference frame, the VSI voltage deviations for $\phi_p = 0^\circ$ (standstill²) become

$$\Delta u_s^{dq}(i_s^{dq}) := \begin{pmatrix} \Delta u_s^d(i_s^{dq}) \\ \Delta u_s^q(i_s^{dq}) \end{pmatrix} = T_c \begin{pmatrix} \Delta u_s^a(i_s^a) \\ \Delta u_s^b(i_s^b) \\ \Delta u_s^c(i_s^c) \end{pmatrix} \quad (7)$$

with Clarke transformation matrix [33, Chap. 14]

$$T_c := \kappa \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \Leftrightarrow T_c^{-1} := \frac{1}{\kappa} \begin{bmatrix} \frac{2}{3} & 0 \\ -\frac{1}{3} & \frac{\sqrt{3}}{3} \\ -\frac{1}{3} & -\frac{\sqrt{3}}{3} \end{bmatrix}$$

where $\kappa \in \{2/3; \sqrt{2}/3\}$ allows for an amplitude or power invariant transformation, respectively. In view of

$$\begin{pmatrix} i_s^a \\ i_s^b \\ i_s^c \end{pmatrix} = T_c^{-1} i_s^{dq} = \begin{pmatrix} -\frac{1}{3\kappa} i_s^d + \frac{\sqrt{3}}{3\kappa} i_s^q \\ -\frac{1}{3\kappa} i_s^d + \frac{\sqrt{3}}{3\kappa} i_s^q \\ -\frac{1}{3\kappa} i_s^d - \frac{\sqrt{3}}{3\kappa} i_s^q \end{pmatrix},$$

the transformation (7) is also applicable for the approximation $\Delta \hat{u}_s^p(i_s^p) \approx J_{\text{ann}}^{\text{vsi}}(i_s^p)$ for all $p \in \{a, b, c\}$ as in (6) resulting in

$$\begin{aligned} \Delta u_s^{dq}(i_s^{dq}) &\approx \Delta \hat{u}_s^{dq}(i_s^{dq}) := \begin{pmatrix} \Delta \hat{u}_s^d(i_s^{dq}) \\ \Delta \hat{u}_s^q(i_s^{dq}) \end{pmatrix} \\ &= \kappa \begin{pmatrix} J_{\text{ann}}^{\text{vsi}}(\frac{2}{3\kappa} i_s^d) - \frac{1}{2} J_{\text{ann}}^{\text{vsi}}(-\frac{1}{3\kappa} i_s^d + \frac{\sqrt{3}}{3\kappa} i_s^q) - \frac{1}{2} J_{\text{ann}}^{\text{vsi}}(-\frac{1}{3\kappa} i_s^d - \frac{\sqrt{3}}{3\kappa} i_s^q) \\ \frac{\sqrt{3}}{2} J_{\text{ann}}^{\text{vsi}}(-\frac{1}{3\kappa} i_s^d + \frac{\sqrt{3}}{3\kappa} i_s^q) - \frac{\sqrt{3}}{2} J_{\text{ann}}^{\text{vsi}}(-\frac{1}{3\kappa} i_s^d - \frac{\sqrt{3}}{3\kappa} i_s^q) \end{pmatrix}, \end{aligned} \quad (8)$$

which implies that the VSI-ANN not only depends on one current in the (d, q) -reference frame but may depend on both.

III. SYNCHRONOUS MACHINE NONLINEARITIES

The electrical dynamics of a synchronous machine in the (d, q) -reference frame are given by [20], [33]

$$u_s^{dq} = R_s i_s^{dq} + \frac{d}{dt} \psi_s^{dq}(i_s^{dq}) + \omega_p J \psi_s^{dq}(i_s^{dq}), \quad (9)$$

with stator voltages $u_s^{dq} = (u_s^d, u_s^q)^\top$, stator (phase) resistance R_s , stator currents $i_s^{dq} = (i_s^d, i_s^q)^\top$, electrical synchronous angular velocity $\omega_p = \frac{d}{dt} \phi_p$ (where $\omega_p = n_p \omega_m$ with pole pair number n_p and mechanical angular velocity ω_m), flux linkages $\psi_s^{dq}(i_s^{dq}) = (\psi_s^d(i_s^d, i_s^q), \psi_s^q(i_s^d, i_s^q))^\top$ and rotation matrix $J := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. The time derivative of the flux linkages

$$\frac{d}{dt} \psi_s^{dq}(i_s^{dq}) = \underbrace{\begin{pmatrix} \frac{\partial \psi_s^d(i_s^d, i_s^q)}{\partial i_s^d} & \frac{\partial \psi_s^d(i_s^d, i_s^q)}{\partial i_s^q} \\ \frac{\partial \psi_s^q(i_s^d, i_s^q)}{\partial i_s^d} & \frac{\partial \psi_s^q(i_s^d, i_s^q)}{\partial i_s^q} \end{pmatrix}}_{=: L_s^{dq}(i_s^{dq}) \in \mathbb{R}^{2 \times 2}} \begin{pmatrix} \frac{d}{dt} i_s^d \\ \frac{d}{dt} i_s^q \end{pmatrix} \quad (10)$$

gives the current dynamics with differential inductance matrix $L_s^{dq}(i_s^{dq})$ which consists of the differential inductances $L_s^{dd}(i_s^d, i_s^q)$ and $L_s^{qq}(i_s^d, i_s^q)$ and the differential mutual

²The Park transformation for $\phi_p = 0^\circ$ simplifies to a unity matrix.

(cross-coupling) inductance $L_s^{dq}(i_s^d, i_s^q) = L_s^{qd}(i_s^d, i_s^q)$. For anisotropic machines which exhibit saturation effects such as the interior permanent magnet machine (IPMSM) or the reluctance synchronous machine (RSM), all differential inductances depend on both currents $i_s^{dq} = (i_s^d, i_s^q)^\top$. For anisotropic machines with purely self-saturation effects, the inductances simplify to $L_s^{dd}(i_s^d)$, $L_s^{qq}(i_s^q)$ (both only depend on their respective axis current) and $L_s^{dq}(i_s^d, i_s^q) = L_s^{qd}(i_s^d, i_s^q) = 0$.

A. Simplified current dynamics at standstill

In this work, the MUT is identified at standstill (i.e. $\omega_p = 0 \frac{\text{rad}}{\text{s}}$) such that (9) reduces to

$$u_s^{dq} = R_s i_s^{dq} + \frac{d}{dt} \psi_s^{dq}(i_s^{dq}) \quad (11)$$

which leads to the simplified current dynamics at standstill

$$\frac{d}{dt} i_s^{dq} = L_s^{dq}(i_s^{dq})^{-1} (u_s^{dq} - R_s i_s^{dq}). \quad (12)$$

Invoking forward Euler discretization (i.e. $\frac{d}{dt} x \approx \frac{x[n+1] - x[n]}{T_s}$) with sampling time T_s and instants n and $n+1$ yields

$$i_s^{dq}[n+1] \approx T_s L_s^{dq}(i_s^{dq}[n])^{-1} (u_s^{dq}[n] - R_s i_s^{dq}[n]) + i_s^{dq}[n],$$

which may be rewritten in a more compact form as follows

$$\begin{aligned} i_s^{dq}[n+1] &\approx \underbrace{[I_2 - R_s T_s L_s^{dq}(i_s^{dq}[n])^{-1}]}_{=: A_s^{dq}[n] \in \mathbb{R}^{2 \times 2}} i_s^{dq}[n] \\ &\quad + \underbrace{T_s L_s^{dq}(i_s^{dq}[n])^{-1}}_{=: B_s^{dq}[n] \in \mathbb{R}^{2 \times 2}} u_s^{dq}[n]. \end{aligned} \quad (13)$$

If the cross-coupling effects (i.e. assuming $L_s^{dq} = L_s^{qd} = 0$) are neglected and only the self-axis differential inductances (i.e. $L_s^{dd}(i_s^d, 0)$ and $L_s^{qq}(0, i_s^q)$) are considered, one obtains the simplified difference equations

$$i_s^d[n+1] \approx \underbrace{\left(1 - \frac{T_s R_s}{L_s^{dd}(i_s^d[n], 0)}\right)}_{=: a_s^d[n]} i_s^d[n] + \underbrace{\frac{T_s}{L_s^{dd}(i_s^d[n], 0)}}_{=: b_s^d[n]} u_s^d[n] \quad (14)$$

for the d -axis current and

$$i_s^q[n+1] \approx \underbrace{\left(1 - \frac{T_s R_s}{L_s^{qq}(0, i_s^q[n])}\right)}_{=: a_s^q[n]} i_s^q[n] + \underbrace{\frac{T_s}{L_s^{qq}(0, i_s^q[n])}}_{=: b_s^q[n]} u_s^q[n] \quad (15)$$

for the q -axis current, respectively.

B. ANN-based approximation of self-axis flux linkages

The flux linkages can be written as [34]

$$\psi_s^{dq}(i_s^{dq}) = \begin{pmatrix} \psi_{s,\text{self}}^d(i_s^d) \\ \psi_{s,\text{self}}^q(i_s^q) \end{pmatrix} + \begin{pmatrix} \psi_{s,\text{cross}}^d(i_s^d, i_s^q) \\ \psi_{s,\text{cross}}^q(i_s^d, i_s^q) \end{pmatrix} \quad (16)$$

where $\psi_{s,\text{self}}^d(i_s^d)$ and $\psi_{s,\text{self}}^q(i_s^q)$ reflect the self-axis flux linkages which only depend on the respective axis currents, whereas $\psi_{s,\text{cross}}^d(i_s^d, i_s^q)$ and $\psi_{s,\text{cross}}^q(i_s^d, i_s^q)$ describe the cross-coupling flux linkages which depend on both currents. For the

initial self-axis identification, the cross-coupling effects are neglected and, therefore, for each axis $x \in \{d, q\}$,

$$\psi_{s,\text{self}}^x(i_s^x) \approx \widehat{\psi}_{s,\text{self}}^x(i_s^x) := f_{\text{ann}}^{\psi_{s,\text{self}}^x}(i_s^x) \quad (17)$$

is chosen as approximation $\widehat{\psi}_{s,\text{self}}^x(i_s^x)$ of the self-axis flux linkages $\psi_{s,\text{self}}^x(i_s^x)$. For each axis $x \in \{d, q\}$, each proposed ANN architecture has one input and one output as illustrated in Fig. 3 and consists of one input layer (Layer 0) and one output layer (Layer 2) each with identity activation function $\Phi_0(y) = \Phi_2(y) = \Phi_{\text{id}}(y) = y$ and one hidden layer (Layer 1) with two neurons utilizing the tanh activation function $\Phi_{1,1}(y) = \Phi_{1,2}(y) = \Phi_{\text{tanh}}(y) = \frac{1-e^{-2y}}{1+e^{-2y}}$ (cf. [31]).

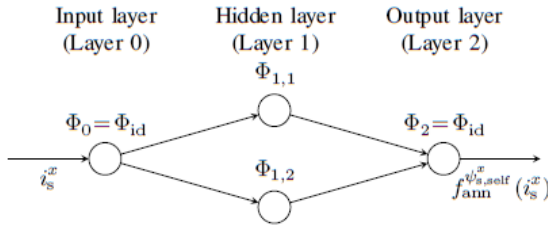


Figure 3: ANN for approximation of self-axis flux linkage $\psi_{s,\text{self}}^x(i_s^x)$ of axis $x \in \{d, q\}$ by $\widehat{\psi}_{s,\text{self}}^x(i_s^x) := f_{\text{ann}}^{\psi_{s,\text{self}}^x}(i_s^x)$.

For the ANN as in Fig. 3 and $x \in \{d, q\}$, each self-axis flux linkage is approximated by

$$\widehat{\psi}_{s,\text{self}}^x(i_s^x) = w_{2,1}^{x,\text{self}} \frac{1 - \exp(-2(w_{1,1}^{x,\text{self}} i_s^x + b_{1,1}^{x,\text{self}}))}{1 + \exp(-2(w_{1,1}^{x,\text{self}} i_s^x + b_{1,1}^{x,\text{self}}))} + w_{2,2}^{x,\text{self}} \frac{1 - \exp(-2(w_{1,2}^{x,\text{self}} i_s^x + b_{1,2}^{x,\text{self}}))}{1 + \exp(-2(w_{1,2}^{x,\text{self}} i_s^x + b_{1,2}^{x,\text{self}}))} + b_2^{x,\text{self}} \quad (18)$$

with (six or) seven ANN parameters $w_1^{x,\text{self}} := (w_{1,1}^{x,\text{self}}, w_{1,2}^{x,\text{self}})^\top$, $b_1^{x,\text{self}} := (b_{1,1}^{x,\text{self}}, b_{1,2}^{x,\text{self}})^\top$ of Layer 1 and $w_2^{x,\text{self}} := (w_{2,1}^{x,\text{self}}, w_{2,2}^{x,\text{self}})^\top$ and $b_2^{x,\text{self}}$ of Layer 2 which need to be identified per axis.

If the considered (IP)MSM is rather linear, one can also use

$$\widehat{\psi}_{s,\text{self}}^x(i_s^x) := w_{2,1}^{x,\text{self}} i_s^x - w_{2,2}^{x,\text{self}} \ln(1 + \exp(-i_s^x)) + b_2^{x,\text{self}} \quad (19)$$

with $\Phi_{1,1}(y) = y$ (identity) and $\Phi_{1,2}(y) = \Phi_{\text{spa}}(y) := \ln(1 + e^y)$ [soft-plus activation (spa) function [35]] which, in Layer 2, are weighted by $w_{2,1}^{x,\text{self}}$ describing the linear property and $w_{2,2}^{x,\text{self}}$ accounting for the saturation characteristic. In (18) and (19), the permanent magnet flux linkage is considered by the bias $b_2^{x,\text{self}}$ for both axis $x \in \{d, q\}$, respectively.

Remark 4. Please note that, (i) for RSMs without permanent magnet, the biases $b_2^{d,\text{self}} = b_2^{q,\text{self}} = 0$ can be neglected; whereas (ii) for IPMSMs, $b_2^{d,\text{self}} = 0$ and $b_2^{q,\text{self}} = \psi_{\text{pm}}(\vartheta_r) > 0$, or (iii) for permanent-magnet assisted RSMs (PMA-RSMs) [36], $b_2^{d,\text{self}} = 0$ and $b_2^{q,\text{self}} = \psi_{\text{pm}}(\vartheta_r) < 0$ must be introduced to model the (rotor) temperature-dependent flux linkage of the permanent magnet. It is usually identified while the machine rotates (see [5]–[7]), but it can also be identified

at standstill [8]. A priori knowledge, e.g., in form of a look-up table (LUT), can be exploited to reduce identification effort.

Remark 5. If apparent inductances $L_{s,\text{app}}^{xx}$ are required those can easily be computed with the help of the flux linkage ANNs as follows $L_{s,\text{app}}^{xx} = \frac{\partial \widehat{\psi}_{s,\text{self}}^x(i_s^x)}{\partial i_s^x}$ for $x \in \{d, q\}$.

C. ANN-based approximation of cross-axis flux linkages

For the cross-axis identification, for each axis $x \in \{d, q\}$,

$$\psi_{s,\text{cross}}^x(i_s^d, i_s^q) \approx \widehat{\psi}_{s,\text{cross}}^x(i_s^d, i_s^q) := f_{\text{ann}}^{\psi_{s,\text{cross}}^x}(i_s^d, i_s^q) \quad (20)$$

is chosen as the approximation $\widehat{\psi}_{s,\text{cross}}^x(i_s^d, i_s^q)$ of the cross-axis flux linkages $\psi_{s,\text{cross}}^x(i_s^d, i_s^q)$ in (16). The proposed ANN architecture for cross-axis identification is shown in Fig. 4. It has two inputs and one output and consists of one input layer (Layer 0) and one output layer (Layer 2) each with identity activation function $\Phi_0(y) = \Phi_2(y) = \Phi_{\text{id}}(y) = y$. The one hidden layer (Layer 1) comes with two (for RSMs) or three (for IPMSMs) neurons and utilizes special activation functions $\Phi_{1,1}(y_1, y_2) = \Phi_{1,2}(y_1, y_2) = \Phi_{1,3}(y_1, y_2) = \Phi'_m(y_1)\Phi_m(y_2)$ for the d -axis and $\Phi_m(y_1)\Phi'_m(y_2)$ for the q -axis of machine $m \in \{\text{ipmsm}, \text{rsm}\}$, respectively. Hence, one obtains

$$\widehat{\psi}_{s,\text{cross}}^d(i_s^d, i_s^q) = \sum_{i=1}^n w_{2,i}^{\text{cross}} \Phi'_m(i_s^d) \Phi_m(i_s^q) \quad (21)$$

$$\widehat{\psi}_{s,\text{cross}}^q(i_s^d, i_s^q) = \sum_{i=1}^n w_{2,i}^{\text{cross}} \Phi_m(i_s^d) \Phi'_m(i_s^q) \quad (22)$$

such that $L_s^{dq} = \frac{\partial \widehat{\psi}_{s,\text{cross}}^d(i_s^d, i_s^q)}{\partial i_s^q} = \frac{\partial \widehat{\psi}_{s,\text{cross}}^q(i_s^d, i_s^q)}{\partial i_s^d} = L_s^{qd}$ and conservation of energy [34] is also assured by the approximation of the cross-axis flux linkages, i.e.

$$\frac{\partial \widehat{\psi}_{s,\text{cross}}^d(i_s^d, i_s^q)}{\partial i_s^q} = \frac{\partial \widehat{\psi}_{s,\text{cross}}^q(i_s^d, i_s^q)}{\partial i_s^d} = \sum_{i=1}^n w_{2,i}^{\text{cross}} \Phi'_m(i_s^d) \Phi'_m(i_s^q).$$

For IPMSMs (i.e., $m = \text{ipmsm}$), in (21) and (22), we set $n = 3$ and choose $\Phi_m(y) = \Phi_{\text{ipmsm}}(y) := y + y^2 + \ln(1 + \exp(y))$ with $\Phi'_m(y) = \Phi'_{\text{ipmsm}}(y) = 1 + 2y + \frac{1}{1 + \exp(-y)}$ (weights and biases omitted) which leads to (23).

For RSMs (i.e., $m = \text{rsm}$), in (21) and (22), we set $n = 2$ and choose $\Phi_m(y) = \Phi_{\text{rsm}}(y) := 1 - \exp(-y^2)$ with $\Phi'_m(y) = \Phi'_{\text{rsm}}(y) = 2y \exp(-y^2)$ (motivated by [34]; again weights and biases omitted) which yields (24).

D. Approximation of differential inductances

According to (10), the differential inductances are the partial derivatives of the flux linkages with respect to the currents. In view of (16), those can be directly approximated by

$$\begin{pmatrix} \widehat{L}_s^{dd}(i_s^d, i_s^q) \\ \widehat{L}_s^{dq}(i_s^d, i_s^q) \end{pmatrix} = \begin{pmatrix} \frac{\partial \widehat{\psi}_{s,\text{self}}^d(i_s^d)}{\partial i_s^d} \\ \frac{\partial \widehat{\psi}_{s,\text{self}}^q(i_s^q)}{\partial i_s^q} \end{pmatrix} + \begin{pmatrix} \frac{\partial \widehat{\psi}_{s,\text{cross}}^d(i_s^d, i_s^q)}{\partial i_s^d} \\ \frac{\partial \widehat{\psi}_{s,\text{cross}}^q(i_s^d, i_s^q)}{\partial i_s^q} \end{pmatrix} \quad (25)$$

and

$$\widehat{L}_s^{dq}(i_s^d, i_s^q) = \frac{\partial \widehat{\psi}_{s,\text{cross}}^d(i_s^d, i_s^q)}{\partial i_s^q} = \frac{\partial \widehat{\psi}_{s,\text{cross}}^q(i_s^d, i_s^q)}{\partial i_s^d} = \widehat{L}_s^{qd}(i_s^d, i_s^q) \quad (26)$$

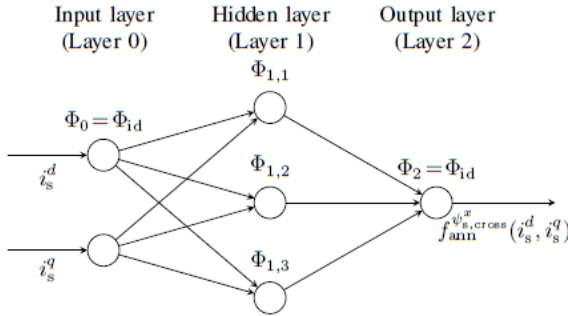


Figure 4: ANN for approximation of cross-axis flux linkage $\psi_{s,\text{cross}}^x(i_s^d, i_s^q)$ of axis $x \in \{d, q\}$ by $\hat{\psi}_{s,\text{cross}}^x(i_s^d, i_s^q) := f_{\text{ann}}^{\psi_{s,\text{cross}}^x}(i_s^d, i_s^q)$.

with the help of the ANN-based flux linkage estimates in (17) and (20). As the computation of the partial derivatives of (18), (19), (23) and (24) is lengthy but straightforward, the explicit expressions of the differential inductance approximations are omitted due to space limitations.

IV. OVERALL IDENTIFICATION ALGORITHM

The proposed identification consists of three parts: (P1) a rotor alignment and locking procedure (at least for a q -axis characterisation); (P2) a special open-loop voltage excitation of the machine under test (MUT) at standstill for a fraction of a second or a few seconds; and (P3) a post-processing identification method which solves a recursive least squares optimization problem offline (for details see [37, Section 4.2 & 10.4.5] or [31]). The three parts will be discussed in more detail in the following.

A. Rotor alignment and standstill operation (P1)

At the beginning of the identification process, the machine is excited in α -axis direction with a constant reference voltage magnitude to align the machine rotor with the d -axis in order to achieve $\phi_p \approx 0^\circ$. Next, the reference voltage is slowly increased until a pre-defined maximum (e.g., rated) current is obtained in order to guarantee correct alignment (i.e. $\phi_p = 0^\circ$). Then, the rotor is mechanically locked. Alternatively, one can directly lock the rotor and apply a high-frequency position estimation technique [38]. Afterwards, the self-axis and cross-axis identification is conducted with the following two steps:

- (S1) Open-loop excitation (for self-axis identification, d and q -axis are excited separately; whereas for cross-coupling identification, both axes are excited simultaneously).
- (S2) Post-processing identification (the obtained weights during self-axis identification serve as initial weights for cross-axis identification).

Both steps are explained in more detail in the following.

B. Open-loop excitation (P2)

For a simultaneous estimation of all ANN parameters of the VSI and machine model, a “sufficiently rich” excitation signal

must be applied to produce current responses within the high- and low-current region (see Fig. 1). From a system theoretical point of view, one must assure “persistence of excitation” [39], which, according to the applied voltage steps due to the switching nature of the inverter, is guaranteed as steps contain (theoretically) infinitely many frequencies. Moreover, properly chosen and varying excitation (reference) voltages (see (27)) in combination with the nonlinear phase voltage deviations Δu_s^p as in (4) excite the current dynamics additionally. To improve the richness of the excitation signal further and to distinguish during steady-state operation of the current response between a voltage drop caused by Δu_s^p and a voltage drop over the stator resistance R_s , the excitation magnitude should vary e.g. after a full excitation period $\frac{1}{f_{\text{ext}}}$ (see Sec. V-B), such that a steady-state current response is guaranteed within the high-current and the low-current region. In conclusion, a proper excitation signal in the (d, q) -reference frame must be chosen

$$u_{s,\text{ref}}^{d/q}(t) = \begin{cases} \bar{u}_{\text{sat}}(t), & \text{if } u_{\text{ext}}(t) \geq \bar{u}_{\text{sat}}(t) \\ \underline{u}_{\text{sat}}(t), & \text{if } u_{\text{ext}}(t) \leq \underline{u}_{\text{sat}}(t) \\ \underbrace{a_{\text{ext}}(t) \sin(2\pi f_{\text{ext}} t)}_{=: u_{\text{ext}}(t)}, & \text{else,} \end{cases} \quad (27)$$

where $a_{\text{ext}}(t)$, $\underline{u}_{\text{sat}}(t)$ and $\bar{u}_{\text{sat}}(t)$ may change after each excitation period to allow for different amplitudes and lower and upper saturation levels of the test signal to avoid over-currents and to force different steady-state responses.

As a rule of thumb, (at least) two cycles of (27) should be applied: One cycle for the high-current region and one cycle for the low-current region. Each cycle should contain at least 100 samples where at least $N_{\text{min}} = 50$ samples for steady state and transient response should be recorded, respectively. The following steps to obtain a proper parametrisation of (27) are recommended: (i) The α/d -axis phase voltage is ramped up, e.g. by $+1 \frac{V}{s}$, until a predefined maximum current is reached, e.g., $I_{\text{max}} = I_R$ (rated current). The corresponding maximum voltage for I_{max} is stored as \bar{u}_{sat} and prevents further over-currents. The same procedure can be repeated for the low-current region with e.g. $I_{\text{max}} = 2.5 \text{ A}$ (see Fig. 1); (ii) the excitation signal in (27) now can be pre-parametrised for the high-current and the low-current region with \bar{u}_{sat} , $a_{\text{ext}} = \bar{u}_{\text{sat}}$ and $f_{\text{ext}} = f_R$ (rated machine frequency), respectively; (iii) afterwards f_{ext} is reduced, e.g. by $-1 \frac{\text{Hz}}{s}$, until the measured currents reach their respective limit I_{max} in both current regions; and (iv) a_{ext} is increased again until at least N_{min} samples are obtained for steady state and transient current response, respectively. Finally, all parameters for high-current and low-current region are stored and two cycles of the excitation signal (27) are applied.

C. Post-processing identification (P3)

Goal is to find an expression which allows to identify all parameters of the ANNs simultaneously only based on the current measurements $i_s^{dq}[n]$ and the applied (reference) voltages $u_{s,\text{ref}}^{dq}[n]$ for a set of samples $n \in \{0, 1, \dots, N\}$ (where 0 represents the initial time step and $N \gg 1$ is large). To do so, the current dynamics in (13) in combination with

$$\begin{aligned}
 \text{IPMSMs: } \left\{ \begin{aligned}
 \hat{\psi}_{s,\text{cross}}^d(i_s^d, i_s^q) &= w_{2,1}^{\text{cross}} \left(w_{1,1,1}^{d,\text{cross}} + 2w_{1,1,2}^{d,\text{cross}} i_s^d + \frac{w_{1,1,3}^{d,\text{cross}}}{1+\exp(-(i_s^d+b_{1,1}^{d,\text{cross}}))} \right) \\
 &\quad \left(w_{1,1,1}^{q,\text{cross}} i_s^q + w_{1,1,2}^{q,\text{cross}} (i_s^q)^2 + w_{1,1,3}^{q,\text{cross}} \ln(\exp(i_s^q + b_{1,1}^{q,\text{cross}}) + 1) \right) \\
 &+ w_{2,2}^{\text{cross}} \left(w_{1,2,1}^{d,\text{cross}} + 2w_{1,2,2}^{d,\text{cross}} i_s^d + \frac{w_{1,2,3}^{d,\text{cross}}}{1+\exp(-(i_s^d+b_{1,2}^{d,\text{cross}}))} \right) \\
 &\quad \left(w_{1,2,1}^{q,\text{cross}} i_s^q + w_{1,2,2}^{q,\text{cross}} (i_s^q)^2 + w_{1,2,3}^{q,\text{cross}} \ln(\exp(i_s^q + b_{1,2}^{q,\text{cross}}) + 1) \right) \\
 &+ w_{2,3}^{\text{cross}} \left(w_{1,3,1}^{d,\text{cross}} + 2w_{1,3,2}^{d,\text{cross}} i_s^d + \frac{w_{1,3,3}^{d,\text{cross}}}{1+\exp(-(i_s^d+b_{1,3}^{d,\text{cross}}))} \right) \\
 &\quad \left(w_{1,3,1}^{q,\text{cross}} i_s^q + w_{1,3,2}^{q,\text{cross}} (i_s^q)^2 + w_{1,3,3}^{q,\text{cross}} \ln(\exp(i_s^q + b_{1,3}^{q,\text{cross}}) + 1) \right)
 \end{aligned} \right. \quad (23) \\
 \\
 \text{RSMs: } \left\{ \begin{aligned}
 \hat{\psi}_{s,\text{cross}}^d(i_s^d, i_s^q) &= 2w_{2,1}^{\text{cross}} (w_{1,1}^{d,\text{cross}})^2 i_s^d \exp(-(w_{1,1}^{d,\text{cross}} i_s^d)^2) \left(1 - \exp(-(w_{1,1}^{q,\text{cross}} i_s^q)^2) \right) \\
 &\quad + 2w_{2,2}^{\text{cross}} (w_{1,2}^{d,\text{cross}})^2 i_s^d \exp(-(w_{1,2}^{d,\text{cross}} i_s^d)^2) \left(1 - \exp(-(w_{1,2}^{q,\text{cross}} i_s^q)^2) \right) \\
 \hat{\psi}_{s,\text{cross}}^q(i_s^d, i_s^q) &= 2w_{2,1}^{\text{cross}} (w_{1,1}^{q,\text{cross}})^2 i_s^q \exp(-(w_{1,1}^{q,\text{cross}} i_s^q)^2) \left(1 - \exp(-(w_{1,1}^{d,\text{cross}} i_s^d)^2) \right) \\
 &\quad + 2w_{2,2}^{\text{cross}} (w_{1,2}^{q,\text{cross}})^2 i_s^q \exp(-(w_{1,2}^{q,\text{cross}} i_s^q)^2) \left(1 - \exp(-(w_{1,2}^{d,\text{cross}} i_s^d)^2) \right)
 \end{aligned} \right. \quad (24)
 \end{aligned}$$

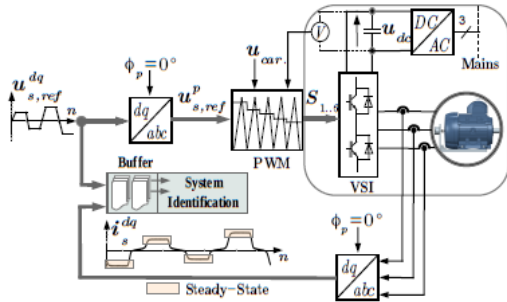


Figure 5: Simplified block diagram of the excitation and identification process.

the proposed ANNs can be used to compute the estimated currents $\hat{i}_s^{dq}[n]$ for each sampled time instant of the recorded measurements. As only currents and voltage references are available for identification, the actual voltages $u_s^{dq}[n]$ in (13) must be replaced by $u_{s,\text{ref}}^{dq}[n] - \Delta u_s^{dq}[n]$ as in (4), which yields

$$\hat{i}_s^{dq}[n+1] \approx A_s^{dq}[n] \hat{i}_s^{dq}[n] + B_s^{dq}[n] \left(u_{s,\text{ref}}^{dq}[n] - \Delta u_s^{dq}[n] \right).$$

Moreover, the entries of the matrices $A_s^{dq}[n]$ and $B_s^{dq}[n]$ are not a priori known and, actually, depend on T_s , $L_s^{dd}(i_s^{dq}[n])$, $L_s^{qq}(i_s^{dq}[n])$, $L_s^{dq}(i_s^{dq}[n])$ and R_s . Hence, those and also the unknown inverter nonlinearities Δu_s^{dq} need to be replaced by

their estimates $\hat{A}_s^{dq}[n] := I_2 - \hat{R}_s T_s \hat{L}_s^{dq}(i_s^{dq}[n])^{-1}$, $\hat{B}_s^{dq}[n] := T_s \hat{L}_s^{dq}(i_s^{dq}[n])^{-1}$ and $\Delta \hat{u}_s^{dq}$, respectively, to predict

$$\hat{i}_s^{dq}[n+1] \approx \hat{A}_s^{dq}[n] \hat{i}_s^{dq}[n] + \hat{B}_s^{dq}[n] \left(u_{s,\text{ref}}^{dq}[n] - \Delta \hat{u}_s^{dq}[n] \right) \quad (28)$$

at the next sampling instant where the estimated VSI nonlinearities as in (8) and estimated differential inductance matrix

$$\hat{L}_s^{dq}(i_s^{dq}[n]) \stackrel{(18)}{=} \begin{bmatrix} =: \hat{L}_s^{dd}(i_s^{dq}[n]) & =: \hat{L}_s^{dq}(i_s^{dq}[n]) \\ \frac{\partial f_{\text{ANN}}^{\psi,d}(i_s^{dq}[n])}{\partial i_s^d} & \frac{\partial f_{\text{ANN}}^{\psi,d}(i_s^{dq}[n])}{\partial i_s^q} \\ \frac{\partial f_{\text{ANN}}^{\psi,q}(i_s^{dq}[n])}{\partial i_s^d} & \frac{\partial f_{\text{ANN}}^{\psi,q}(i_s^{dq}[n])}{\partial i_s^q} \\ =: \hat{L}_s^{qd}(i_s^{dq}[n]) & =: \hat{L}_s^{qq}(i_s^{dq}[n]) \end{bmatrix}$$

(and, hence, also its inverse $\hat{L}_s^{dq}(i_s^{dq}[n])^{-1}$) can be expressed with the help of the proposed ANNs as in (6), (18), (19), (23), (24), respectively. Finally, collecting (i) the identification parameters of the (to be estimated) stator resistance \hat{R}_s^x and the weights and biases of the ANNs in one parameter vector

$$\theta_s := \left(\hat{R}_s^x, w_{1,1}^{\text{vsi}}, \dots, b_{1,2}^{\text{vsi}}, \dots, w_{1,1}^{d/q,\text{self}}, \dots, w_{1,1,1/1,1}^{d/q,\text{cross}}, \dots \right)^T$$

and (ii) the $N+1$ samples of the currents $i_s^{dq}[0], \dots, i_s^{dq}[N]$ and voltage references $u_{s,\text{ref}}^{dq}[0], \dots, u_{s,\text{ref}}^{dq}[N]$ during the open-loop excitation allows to rewrite, for all $n \in \{0, \dots, N-1\}$, the estimated current dynamics as

$$\hat{i}_s^{dq}[n+1] = f(i_s^{dq}[n], u_{s,\text{ref}}^{dq}[n], \theta_s),$$

which only depends on $i_s^{dq}[n]$, $u_{s,\text{ref}}^{dq}[n]$ and θ_s and can directly be computed with (28). Hence, an optimization problem

$$\theta_s^* := \arg \min_{\theta_s} \left\| \begin{pmatrix} i_s^{dq}[1] - \hat{i}_s^{dq}[1] \\ \vdots \\ i_s^{dq}[N] - \hat{i}_s^{dq}[N] \end{pmatrix} \right\|^2 \quad (29)$$

can be formulated and solved by the Levenberg-Marquardt algorithm or any Particle Swarm Optimization (PSO) algorithm which yields the optimal parameter vector θ_s^* including the estimated stator resistance and the weights and biases of both ANNs for d and q axes. Evaluating the trained ANNs for different currents allows to approximate (i) the VSI voltage deviations $\hat{u}_s^p(i_s^p) = f_{\text{ann}}^{\text{vsi}}(i_s^p)$ as in (6) with $p \in \{a, b, c\}$, (ii) the self-axis flux linkages $\hat{\psi}_{s,\text{self}}^x(i_s^x) = f_{\text{ann}}^{\psi_{s,\text{self}}^x}(i_s^x)$ as in (18) or (19), (iii) the cross-axis flux linkages $\hat{\psi}_{s,\text{cross}}^x(i_s^d, i_s^q) = f_{\text{ann}}^{\psi_{s,\text{cross}}^x}(i_s^d, i_s^q)$ as in (23) for IPMSMs and (24) for RSMs, and (iv) the differential inductances $\hat{L}_s^{dd}(i_s^d, i_s^q)$, $\hat{L}_s^{qq}(i_s^d, i_s^q)$ and $\hat{L}_s^{dq}(i_s^d, i_s^q) = \hat{L}_s^{qd}(i_s^d, i_s^q)$ as in (25) and (26).

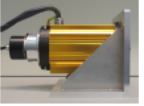

Remark 6. Usually the electrical drive system exhibits a time delay T_Σ comprising delays due to sample and hold circuits of the digital control environment, pulse width modulation (PWM), inverter and analogue-to-digital conversion [2]. For $\frac{1}{f_{\text{ext}}} \gg T_\Sigma$ (see Table II), the system delay is negligible. If this is not the case, one needs to estimate T_Σ during the identification as additional parameter or, if T_Σ is known, $u_{s,\text{ref}}^{dq}[n]$ must be shifted accordingly during post-processing [2].

V. EXPERIMENTAL VALIDATION

A. Description of experimental setup

The proposed identification method, as illustrated in Fig. 5, was implemented on a Cyclone IV (EP4CE22F17C6N) field programmable gate array (FPGA) with the help of a rapid prototyping system for model-based controller design. The test bench consists of a three-phase two-level inverter with Semikron SKM50GB12T4 insulated-gate bipolar transistors (IGBTs) rated for 50 A and 1200 V operating at a switching frequency of $f_{\text{sw}} = 10$ kHz with a dead time of $3\mu\text{s}$. The phase currents were measured by a 12-bit 60 kHz analogue to digital converter using LEM HXS 20-NP Hall sensors.

Table I: Considered machines and key data.

Machine Type		
	M1: IPMSM	M2: RSM
Manufacturer	EMP	Stellenbosch [40]
Designation	Prototype	Prototype
Rated Phase Current I_{R}	4.07 A _{rms}	3.54 A _{rms}
Rated Torque $m_{\text{m,R}}$	2.06 N m	9.60 N m
Poles	8 ($n_p = 4$)	4 ($n_p = 2$)
Rated Speed $n_{\text{m,R}}$	6000 rpm	1500 rpm

The proposed identification is demonstrated for an IPMSM and an RSM with nominal ratings as shown in Table I. The

DC-link voltage was permanently set to $u_{\text{dc}} = 300$ V. Due to that, the VSI voltage deviations $\Delta u_s^p(i_s^p)$ and, therefore, also their approximations $\Delta \hat{u}_s^p(i_s^p)$ in (6) are expected to be very similar for IPMSM and RSM identification. Small deviation in the low current region are possible due to the dependence on the output capacitance of the VSI legs resulting in a varying off time as in (3) [9], [12].

The experimental settings, such as parameters of the excitation signal (27) composed of $\underline{u}_{\text{sat}}(t)$, $\bar{u}_{\text{sat}}(t)$, $a_{\text{ext}}(t)$ and f_{ext} and other implementation data, are collected in Table II. The excitation voltage can be obtained by gradually increasing $a_{\text{ext}}(t)$, $-\underline{u}_{\text{sat}}(t)$ and $\bar{u}_{\text{sat}}(t)$ until a desired current (e.g., three times the rated current) is reached in steady state. Solely, the rated current, voltage and frequency must be known for the identification but can usually be extracted from the nameplate. For the following experiments, the parameters of the excitation signal were obtained by trial and error but could also be obtained by an automated process within a high-level logic in the operating system of the electrical drive. The automated choice of proper excitation signals is still an open research question (e.g., also for other stand-still methods, see [24]) and is not considered in this paper (future research).

Table II: Experimental settings for identification.

	IPMSM		RSM	
	d -axis	q -axis	d -axis	q -axis
Self-axis identification				
Excitation amplitude a_{ext}	25 V	25 V	90 V	90 V
Saturation $\bar{u}_{\text{sat}} = -\underline{u}_{\text{sat}}$	[17, 12] V	[17, 12] V	[54, 27] V	[54, 27] V
Excitation frequency f_{ext}	5 Hz	5 Hz	1 Hz	5 Hz
Sampling frequency f_s	10 kHz	10 kHz	1 kHz	5 kHz
Switching frequency f_{sw}	10 kHz	10 kHz	10 kHz	10 kHz
Samples N per signal (29)	4096	4096	4096	4096
Cross-axis identification	d -axis	q -axis	d -axis	q -axis
Excitation amplitude a_{ext}	25 V	25 V	90 V	90 V
Saturation $\bar{u}_{\text{sat}} = -\underline{u}_{\text{sat}}$	[17, 12] V	[17, 11] V	[54, 27] V	[54, 27] V
Excitation frequency f_{ext}	5 Hz	20 Hz	1 Hz	10 Hz
Sampling frequency f_s	5 kHz	5 kHz	1 kHz	1 kHz
Switching frequency f_{sw}	10 kHz	10 kHz	10 kHz	10 kHz
Samples N per signal (29)	2048	2048	2048	2048

B. Discussion of experimental results

The identification results are shown in Fig. 6 and 7 for the IPMSM, in Fig. 8 and 9 for the RSM and in Fig. 10 for the VSI. The identified model parameters and phase resistance estimation results and references are listed in Tab. III.

In Fig. 6, the self-axis identification results and, in Fig. 7, the cross-axis (full) identification results for the IPMSM are presented. The first subplot (a) of Fig. 6 and Fig. 7 show the time series of excitation signal $u_{s,\text{ref}}^{d/q}$ with respective actual $i_s^{d/q}$ and predicted current $\hat{i}_s^{d/q}$ responses according to (28). Subplots (b) & (c) of Fig. 6 and of Fig. 7 show actual and identified self-axis flux linkage curves and flux linkage maps, respectively. Subplots (d) & (e) of Fig. 6 show actual $L_s^{dd/qq}$ and identified $\hat{L}_s^{dd/qq}$ (self-axis) differential inductances of the IPMSM, whereas subplots (d) & (e) of Fig. 7 show the relative flux linkage map errors of the cross-axis (full) identification.

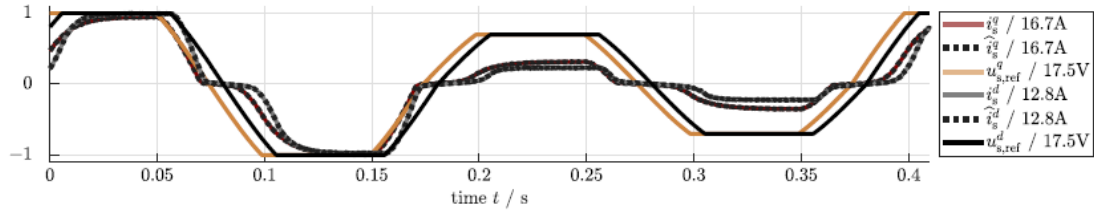
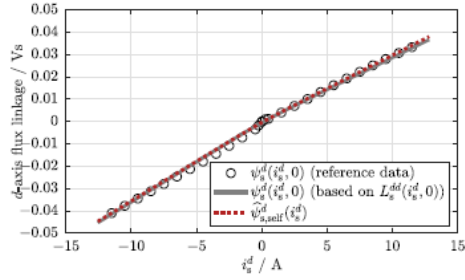
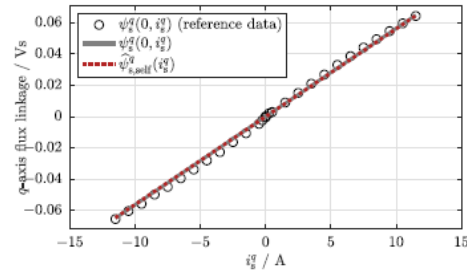
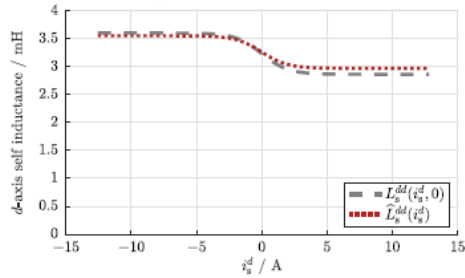
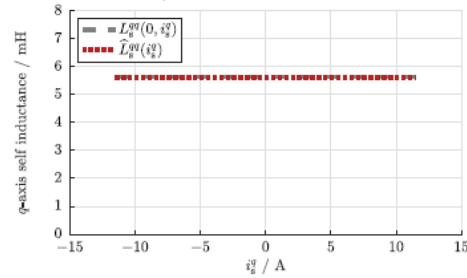

 (a) Time series of the normalized stator d - and q -axis current and its estimate and stator voltage references.

 (b) Comparison of d -axis flux linkage $\psi_s^d(i_s^d, 0)$ and its self-axis estimate $\hat{\psi}_{s,\text{self}}^d(i_s^d) = f_{\text{ann}}^{\psi,d}(i_s^d)$.

 (c) Comparison of q -axis flux linkage $\psi_s^q(0, i_s^q)$ and its self-axis estimate $\hat{\psi}_{s,\text{self}}^q(i_s^q) = f_{\text{ann}}^{\psi,q}(i_s^q)$.

 (d) Comparison of d -axis self inductance $L_s^{dd}(i_s^d, 0)$ and its estimate $\hat{L}_s^{dd}(i_s^d) = \frac{\partial \hat{\psi}_{s,\text{self}}^d(i_s^d)}{\partial i_s^d}$.

 (e) Comparison of q -axis self inductance $L_s^{qq}(0, i_s^q)$ and its estimate $\hat{L}_s^{qq}(i_s^q) = \frac{\partial \hat{\psi}_{s,\text{self}}^q(i_s^q)}{\partial i_s^q}$.

Figure 6: IPMSM identification results: (a) Time series; (b) & (c) self-axis flux linkages; (d) & (e) differential inductances.

The actual (reference) measurements were obtained from the CSM [6]. It can be seen, that the self-axis flux linkages and inductances and the flux linkage maps were identified with good accuracy and small relative errors of -8% to 2% for $\hat{\psi}_s^d(i_s^d, i_s^q)$ and -6% to 5% for $\hat{\psi}_s^q(i_s^d, i_s^q)$. A small deviation can also be seen in the d -axis differential inductance which is due to the asymmetry of the d -axis flux linkage which is particularly dominant within the zero current region of an IPMSM as a result of the permanent magnet [34]. A more accurate representation within this region requires usually a significantly increased model complexity as for which it is often omitted (such as in [16]). The bias $b_2^{q,\text{self}}$ as shown in Tab. III is zero due to the absence of a q -axis permanent magnet flux linkage. Fig. 8 and Fig. 9 illustrate the self-axis and cross-axis (full) identification results for the RSM, respectively. Again, time series, self-axis flux linkage curves or flux linkage maps and self-axis differential inductances or relative flux linkage errors for the d - and q -axis are shown in

subplots (a), (b) & (c) and (d) & (e), respectively. The actual (reference) data of flux linkages and differential inductances was taken from [6]. It can be seen, that the self-axis flux linkage curves and self-axis differential inductance curves were identified with high accuracy. Also, the identified flux linkage maps exhibit (very) small relative errors of -4% to 3% for $\hat{\psi}_s^d(i_s^d, i_s^q)$ and -4% to 6% for $\hat{\psi}_s^q(i_s^d, i_s^q)$. Fig. 10 shows the identification results of the VSI voltage deviations for the IPMSM and RSM. It can be seen that reference and estimation do match very accurately, i.e., $\Delta u_s^{d/q}(i_s^d/i_s^q) \approx \Delta \hat{u}_s^{d/q}(i_s^d/i_s^q)$ with errors less than 0.179 V (2.2%) to 0.320 V (4.0%) with respect to the high current region voltage error. The reference VSI measurements were conducted in [12].

The experiments with the presented identification results for the IPMSM in Fig. 6 (using ANN (19)), Fig. 7 (using ANN (19), (23)), and for the RSM in Fig. 8 (using ANN (18)), Fig. 9 (using ANN (18), (24)) as well as the VSI voltage error in Fig. 10 (using ANN (6)) highlight the capability of the proposed method to *simultaneously* identify

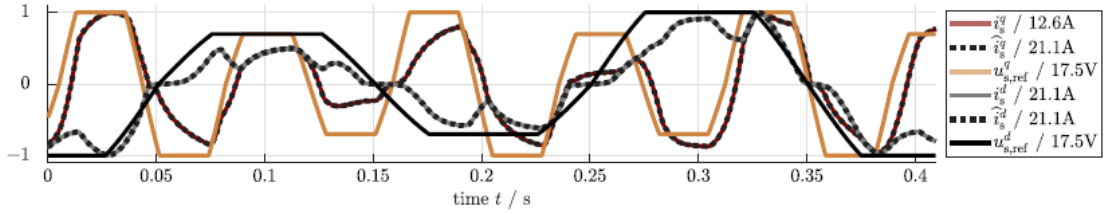
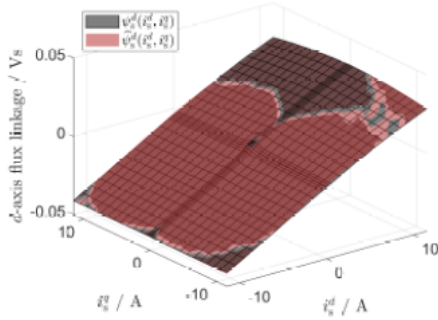
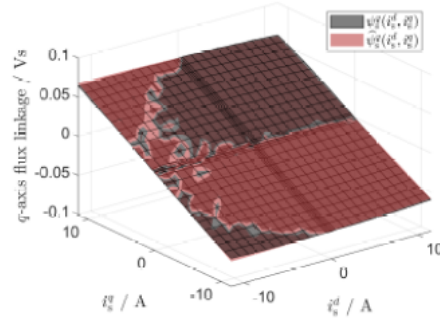
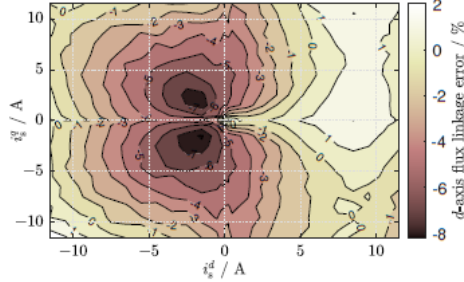
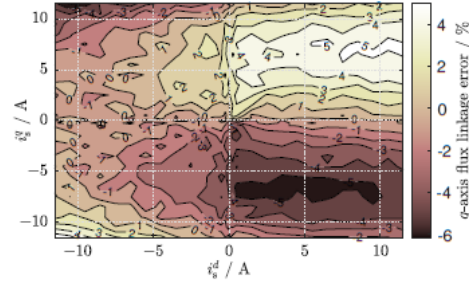
(a) Time series of the normalized stator d - and q -axis current and its estimate and stator voltage references.(b) Comparison of d -axis flux linkage $\psi_s^d(i_s^d, i_s^q)$ and its estimate $\hat{\psi}_s^d(i_s^d, i_s^q)$.(c) Comparison of q -axis flux linkage $\psi_s^q(i_s^d, i_s^q)$ and its estimate $\hat{\psi}_s^q(i_s^d, i_s^q)$.(d) Relative d -flux linkage error $\frac{\psi_s^d(i_s^d, i_s^q) - \hat{\psi}_s^d(i_s^d, i_s^q)}{\psi_{s,max}^d}$.(e) Relative q -flux linkage error $\frac{\psi_s^q(i_s^d, i_s^q) - \hat{\psi}_s^q(i_s^d, i_s^q)}{\psi_{s,max}^q}$.

Figure 7: IPMSM identification results: (a) Time series; (b) & (c) flux linkage maps; (e) & (f) relative flux linkage errors.

(i) the VSI voltage deviations, (ii) the self-axis and cross-axis flux linkages and, therefore, the differential inductances as partial derivatives of the flux linkages as well and (iii) the phase resistance of IPMSM and RSM, respectively. The identification achieves high estimation accuracies compared to the actual VSI/machine nonlinearities and it is very fast as it takes a fraction of a second or a few seconds compared to DTMs with ≈ 45 min or CSMs with ≈ 90 min [9].

VI. CONCLUSION

In this paper, a simple and effective identification method for simultaneous estimation of voltage source inverter and electrical machine nonlinearities has been proposed and validated for an interior permanent magnet machine (IPMSM) and a synchronous reluctance machine (RSM). The estimation is based on structured artificial neural networks (ANNs).

Experimental results have shown, that, at the same time, (i) the voltage deviations due to the VSI nonlinearities, (ii) the current-dependent self- and cross-axis flux linkages and differential inductances as well as (iii) the phase resistance can be estimated with good accuracy. The proposed identification concept is fast and does not require any a priori knowledge of the electrical drive in contrast to other available state-of-the-art methods, except for the rated current, voltage and frequency. Therefore, it can easily be applied in industrial applications (e.g. during end-of-line tests). Future work will focus on (i) other choices of ANN topologies and activation functions and (ii) online self-identification during normal operation; both in order to improve and refine the identification results further.

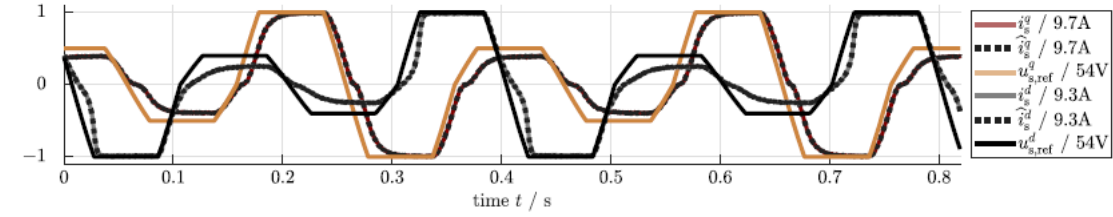
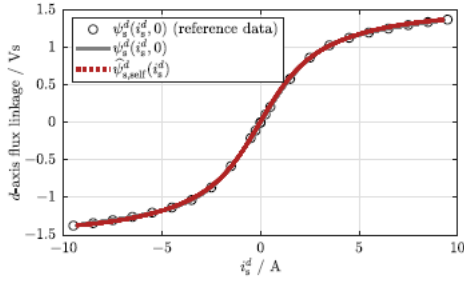
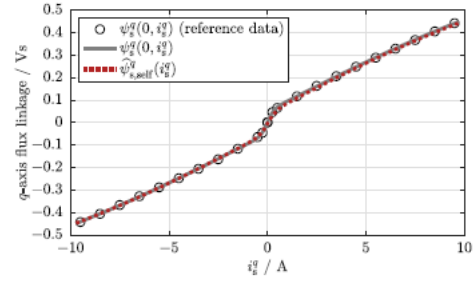
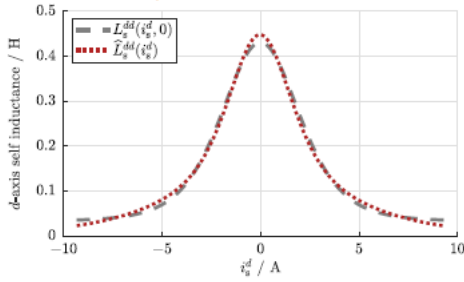
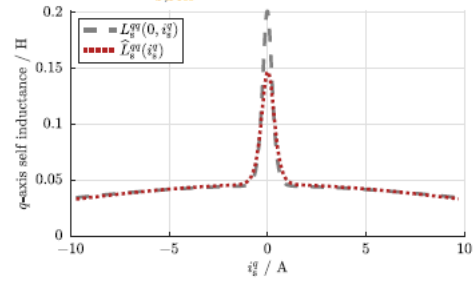
(a) Time series of the normalized stator d - and q -axis current and its estimate and stator voltage references.(b) Comparison of d -axis flux linkage $\psi_s^d(i_s^d, 0)$ and its self-axis estimate $\hat{\psi}_{s,\text{self}}^d(i_s^d) = f_{\text{ann}}^{\psi,d}(i_s^d)$.(c) Comparison of q -axis flux linkage $\psi_s^q(0, i_s^q)$ and its self-axis estimate $\hat{\psi}_{s,\text{self}}^q(i_s^q) = f_{\text{ann}}^{\psi,q}(i_s^q)$.(d) Comparison of d -axis self inductance $L_s^{dd}(i_s^d, 0)$ and its estimate $\hat{L}_s^{dd}(i_s^d) = \frac{\partial \hat{\psi}_{s,\text{self}}^d(i_s^d)}{\partial i_s^d}$.(e) Comparison of q -axis self inductance $L_s^{qq}(0, i_s^q)$ and its estimate $\hat{L}_s^{qq}(i_s^q) = \frac{\partial \hat{\psi}_{s,\text{self}}^q(i_s^q)}{\partial i_s^q}$.

Figure 8: RSM identification results: (a) Time series; (b) & (c) self-axis flux linkages; (d) & (e) differential inductances.

ACKNOWLEDGMENTS

The authors would like to thank the Federal Ministry for “Economic Affairs and Climate Action” for their funding of this research (19I21030H - KIRA).

REFERENCES

- [1] S.A. Odhano, P. Pescetto, H.A.A. Awan, M. Hinkkanen, G. Pellegrino, and R. Bojoi. Parameter identification and self-commissioning in ac motor drives: A technology status review. *IEEE Transactions on Power Electronics*, 34(4):3603–3614, 2019.
- [2] S. Wiedemann and R. Kennel. Self-Commissioning of the Current Control Loop in AC Drives. *International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management, PCIM Europe*, 2018.
- [3] Christoph M. Hackl, Maarten J. Kamper, Julian Kullick, and Joshua Mitchell. Current control of reluctance synchronous machines with online adjustment of the controller parameters. In *Proceedings of the 2016 IEEE International Symposium on Industrial Electronics (ISIE 2016)*, pages 153–160, Santa Clara, CA, USA, jun 2016. Institute of Electrical and Electronics Engineers (IEEE).
- [4] H. Eldeeb, C. Hackl, L. Horlbeck, and J. Kullick. A unified theory for optimal feedforward torque control of anisotropic synchronous machines. *International Journal on Control, Taylor and Francis Group*, 2017.
- [5] L. Ortombina, D. Pasqualotto, F. Tinazzi, and M. Zigliotto. Magnetic model identification of synchronous motors considering speed and load transients. *IEEE Transactions on Industry Applications*, 56(5):4945–4954, 2020.
- [6] S. Wiedemann, S. Hall, R. Kennel, and M. Alaküla. Dynamic testing characterization of a synchronous reluctance machine. *IEEE Transactions on Industry Applications*, 54(2):1370–1378, 2018.
- [7] G. Pellegrino, B. Boazzo, and T. M. Jahns. Magnetic Model Self-Identification for PM Synchronous Machine Drives. *IEEE Transactions on Industry Applications*, 51(3):2246–2254, May 2015.
- [8] P. Pescetto and G. Pellegrino. Sensorless magnetic model and pm flux identification of synchronous drives at standstill. *IEEE International Symposium on Sensorless Control for Electrical Drives (SLED)*, 2017.
- [9] S. Wiedemann. *Self-Identification and Automatic Tuning of Synchronous Machine Drives*. PhD thesis, Technical University of Munich, 2020.
- [10] M. Van Werkhoven and R. Brunner. C.U. 4E EMSA Policy guidelines for motor driven units - Pumps, fans and compressors. *10th International Conference on Energy Efficiency in Motor Driven System (EEMODS)*, Rome, Italy, 2017.
- [11] I. Bojoi, E. Armando, G. Pellegrino, and S. G. Rosu. Self-commissioning

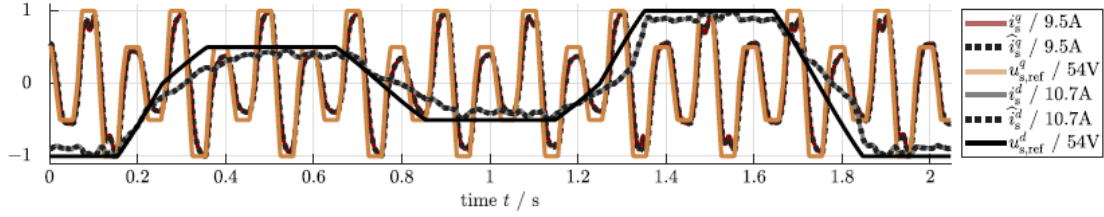
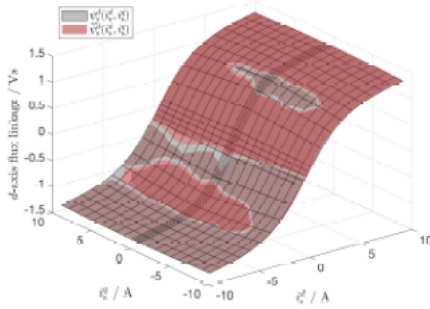
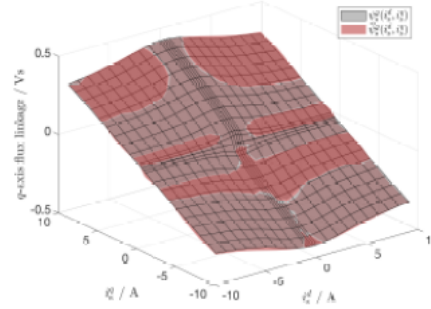
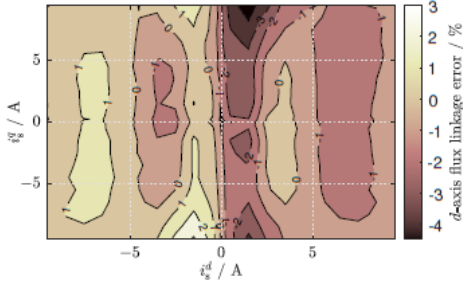
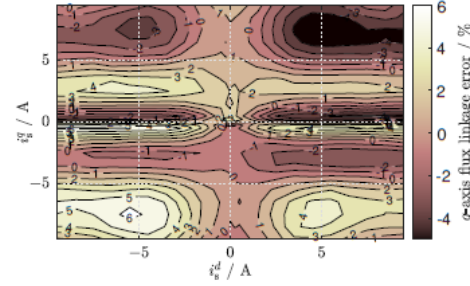
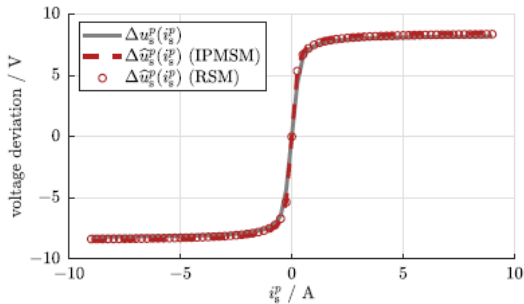
(a) Time series of the normalized stator d - and q -axis current and its estimate and stator voltage references.(b) Comparison of d -axis flux linkage $\psi_s^d(i_s^d, i_s^q)$ and its estimate $\hat{\psi}_s^d(i_s^d, i_s^q)$.(c) Comparison of q -axis flux linkage $\psi_s^q(i_s^d, i_s^q)$ and its estimate $\hat{\psi}_s^q(i_s^d, i_s^q)$.(d) Relative d -flux linkage error $\frac{\psi_s^d(i_s^d, i_s^q) - \hat{\psi}_s^d(i_s^d, i_s^q)}{\psi_{s,max}^d}$.(e) Relative q -flux linkage error $\frac{\psi_s^q(i_s^d, i_s^q) - \hat{\psi}_s^q(i_s^d, i_s^q)}{\psi_{s,max}^q}$.

Figure 9: RSM identification results: (a) Time series; (b) & (c) flux linkages; (e) & (f) relative flux linkage errors.

Figure 10: IPMSM and RSM identification results: Comparison of VSI voltage deviations $\Delta u_s^p(i_s^p)$ and their estimates $\Delta \hat{u}_s^p(i_s^p)$ (d and q axis give very similar results).

- of inverter nonlinear effects in AC drives. *IEEE International Energy Conference and Exhibition (ENERGYCON)*, pages 213–218, Sept 2012.
- [12] S. Wiedemann and R. Kennel. Accurate Self-Identification of Inverter Nonlinear Effects in AC Drives. *International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management, PCIM Europe*, 2018.
- [13] D. Salt, D. Drury, D. Holliday, A. Griffo, P. Sangha, and A. Dinu. Compensation of Inverter Nonlinear Distortion Effects for Signal-Injection-Based Sensorless Control. *IEEE Transactions on Industry Applications*, 47(5), 2011.
- [14] L. Ortombina, F. Tinazzi, and M. Zigliotto. Magnetic Modelling of Synchronous Reluctance and Internal Permanent Magnet Motors Using Radial Basis Function Networks. *IEEE Transactions on Industrial Electronics*, 2017.
- [15] S. Wiedemann and R. Kennel. Encoderless Self-Commissioning and Identification of Synchronous Reluctance Machines at Standstill. *IEEE 26th International Symposium on Industrial Electronics (ISIE)*, June 2017.
- [16] N. Bedetti, S. Calligaro, and R. Petrella. Stand-Still Self-Identification of Flux Characteristics for Synchronous Reluctance Machines Using Novel Saturation Approximating Function and Multiple Linear Regression. *IEEE Transactions on Industry Applications*, 52(4):3083–3092, 2016.

Table III: IPMSM and RSM identification results including the phase resistance R_s (obtained from [41] Sec. 5, [9]), its estimate \hat{R}_s & relative error in % ($b_2^{d,\text{self}}$ obtained from [6]).

	IPMSM		RSM	
	<i>d</i> -axis	<i>q</i> -axis	<i>d</i> -axis	<i>q</i> -axis
R_s	0.45 Ω	0.45 Ω	4.72 Ω	4.72 Ω
\hat{R}_s ; $\frac{\hat{R}_s - R_s}{R_s}$	0.49Ω; 9%	0.48Ω; 6%	4.68Ω; 1%	4.64Ω; 2%
$w_{1,1}^{\text{vsi}}$	7.658	6.488	-6.675	8.142
$w_{1,2}^{\text{vsi}}$	11.54	10.62	-7.386	3.957
$b_{1,1}^{\text{vsi}}$	0.4859	0.2905	0.8266	-0.1667
$w_{2,1}^{\text{vsi}}$	5.993	6.426	-4.613	6.447
$w_{2,2}^{\text{vsi}}$	2.583	2.084	-3.883	1.881
$b_{1,2}^{\text{vsi}}$	-2.115	-2.151	-3.245	7.824
$w_{1,1}^{x,\text{self}}$	—	—	0.4849	0.06072
$w_{1,2}^{x,\text{self}}$	—	—	-0.1625	2.345
$b_{1,1}^{x,\text{self}}$	—	—	0.01482	-0.003926
$w_{2,1}^{x,\text{self}}$	0.003554	0.005608	0.6599	0.7583
$w_{2,2}^{x,\text{self}}$	0.0005852	0	-0.7862	0.04309
$b_{1,2}^{x,\text{self}}$	—	—	-0.00678	-0.0544
$b_2^{x,\text{self}}$	0.0685	—	—	—
$w_{2,1}^{x,\text{cross}}$	0.038796	—	-0.210865	—
$w_{2,2}^{x,\text{cross}}$	0.036888	—	-19.663191	—
$w_{2,3}^{x,\text{cross}}$	0.002055	—	—	—
$w_{1,1,1/1,1}^{x,\text{cross}}$	-0.005107	-0.007110	0.356393	-0.266645
$w_{1,1,2}^{x,\text{cross}}$	0.008513	-0.000375	—	—
$w_{1,1,3}^{x,\text{cross}}$	0.006492	0.002031	—	—
$w_{1,2,1/1,2}^{x,\text{cross}}$	0.013578	0.001272	-0.155646	0.018622
$w_{1,2,2}^{x,\text{cross}}$	0.000498	-0.048376	—	—
$w_{1,2,3}^{x,\text{cross}}$	-0.005737	-0.004789	—	—
$w_{1,3,1}^{x,\text{cross}}$	0.006845	0.006610	—	—
$w_{1,3,2}^{x,\text{cross}}$	0.002189	0.002987	—	—
$w_{1,3,3}^{x,\text{cross}}$	0.024363	-0.019914	—	—
$b_{1,1}^{x,\text{cross}}$	0.006003	0.028532	—	—
$b_{1,2}^{x,\text{cross}}$	0.000243	0.002877	—	—
$b_{1,3}^{x,\text{cross}}$	0.003881	0.006408	—	—

[17] Shih-Wei Su, Christoph Michael Hackl, and Ralph Kennel. Analytical prototype functions for flux linkage approximation in synchronous machines. *IEEE Open Journal of the Industrial Electronics Society*, pages 1–1, 2022.

[18] Z. Qu, T. Tuovinen, and M. Hinkkanen. Inclusion of magnetic saturation in dynamic models of synchronous reluctance motors. *International Conference on Electrical Machines*, pages 994–1000, 2012.

[19] M. Seilmeier and B. Piepenbräuer. Identification of steady-state inductances of PMSM using polynomial representations of the flux surfaces. *39th Annual Conference of the IEEE Industrial Electronics Society (IECON)*, pages 2899–2904, 2013.

[20] S. Kuehl, P. Landsmann, and R. Kennel. Bivariate polynomial approximation of cross-saturated flux curves in synchronous machine models. *IEEE International Energy Conference and Exhibition (ENERGYCON)*, pages 219–224, Sept. 2012.

[21] E. Armando, R. Bojoi, P. Guglielmi, G. Pellegrino, and M. Pastorelli. Experimental Identification of the Magnetic Model of Synchronous Machines. *IEEE Transactions on Industry Applications*, 49(5):2116–2125, Sept. 2013.

[22] A. Varatharajan, G. Pellegrino, and E. Armando. Kinetic-rotor self-commissioning of synchronous machines for magnetic model identification with online adaptation. *IEEE Transactions on Industry Applications*, pages 1–1, 2022.

[23] M. Hinkkanen, P. Pescetto, E. Molsa, S. Saarakkala, G. Pellegrino, and R. Bojoi. Sensorless self-commissioning of synchronous reluctance

motors at standstill without rotor locking. *IEEE Transactions on Industry Applications*, 2016.

[24] Paolo Pescetto and Gianmario Pellegrino. Sensorless standstill commissioning of synchronous reluctance machines with automatic tuning. In *2017 IEEE International Electric Machines and Drives Conference (IEMDC)*, pages 1–8, 2017.

[25] Y. Murai, T. Watanabe, and H. Iwasaki. Waveform Distortion and Correction Circuit for PWM Inverters with Switching Lag-Times. *IEEE Transactions on Industry Applications*, pages 881–886, September 1987.

[26] Y. Murai, A. Riyanto, H. Nakamura, and K. Matsui. PWM strategy for high frequency carrier inverters eliminating current clamps during switching dead-time. *Conference Record of the 1992 IEEE Industry Applications Society Annual Meeting*, pages 317–322, October 1992.

[27] J. Holtz and J. Quan. Sensorless vector control of induction motors at very low speed using a nonlinear inverter model and parameter identification. *Conference Record of the 2001 IEEE Industry Applications Conference. 36th IAS Annual Meeting*, pages 2614–2621, September 2001.

[28] N. Pothi, S. Premrudeepreechacharn, and C. Rakpenthai. Voltage Losses Compensation Using Artificial Neural Network for Estimation Nonlinear Characteristic of Switches. *International Conference on Power Electronics and Drive Systems*, pages 1015–1019, November 2007.

[29] A. R. Munoz and T. Lipo. On-line dead-time compensation technique for open-loop PWM-VSI drives. *IEEE Transactions on Power Electronics*, 14(4):683–689, July 1998.

[30] Y. Park and S. Sul. A novel method utilizing trapezoidal voltage to compensate for inverter nonlinearity. *IEEE Transactions on Power Electronics*, 27(12):4837–4846, 2012.

[31] Max A. Buettner, Niklas Monzen, and Christoph M. Hackl. Artificial neural network based optimal feedforward torque control of interior permanent magnet synchronous machines: A feasibility study and comparison with the state-of-the-art. *Energies*, 15(5):1838, 2022.

[32] S. Wiedemann and R. Kennel. Modelling of Inverter Nonlinear Effects. *International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management, PCIM Europe*, 2018.

[33] Christoph M. Hackl. *Non-identifier based adaptive control in mechatronics: Theory and Application*. Number 466 in Lecture Notes in Control and Information Sciences. Springer International Publishing, Berlin, 2017.

[34] Shih-Wei Su, Christoph M. Hackl, and Ralph Kennel. Analytical prototype functions for flux linkage approximation in synchronous machines. *IEEE Open Journal of the Industrial Electronics Society*, 3:265–282, 2022.

[35] Xavier Glorot, Antoine Bordes, and Y. Bengio. Deep sparse rectifier neural networks. *International Conference on Artificial Intelligence and Statistics (AISTAT)*, 14, 2011.

[36] Hisham Eldeeb, Christoph M. Hackl, Lorenz Horlbeck, and Julian Kullick. A unified theory for optimal feedforward torque control of anisotropic synchronous machines. *International Journal of Control*, 2017.

[37] Dierk Schröder and Martin Buss. *Intelligente Verfahren: Identifikation und Regelung nichtlinearer Systeme*. Springer-Verlag, Berlin, 2017.

[38] S.A. Odhano, P. Giangrande, R. Bojoi, and C. Gerada. Self-commissioning of interior permanent magnet synchronous motor drives with high-frequency current injection. In *2013 IEEE Energy Conversion Congress and Exposition*, pages 3852–3859, 2013.

[39] Lennart Ljung. *System Identification*. Prentice Hall PTR, 2 edition, 1999.

[40] E. Howard, M. Kamper, and S. Gerber. Asymmetric flux barrier and skew design optimization of reluctance synchronous machines. *IEEE Transactions on Industry Applications*, 51(5):3751–3760, 2015.

[41] PMSM Electrical Parameters Measurement. Application note, number: an4680, rev. 0, 02/2013, Freescale Semiconductor, Feb. 2013.

IEEE TRANSACTIONS ON ENERGY CONVERSION - PREPRINT, DOI: 10.1109/TEC.2023.3263353



SIMON WIEDEMANN was born in Kassel, Germany, in 1987. He received the B.Eng. degree in electrical engineering and information technology from the Hamburg University of Applied Sciences, Hamburg, Germany, in 2012, the M.Sc. degree in Electrical Power from the University of Newcastle-Upon-Tyne, Newcastle-Upon-Tyne, U.K., in 2014 and his Dr.-Ing. (Ph.D.) degree in the Institute for Electrical Drive Systems and Power Electronics, Technical University of Munich, Munich, Germany, as a Marie Curie Research Scholar in 2021. Since

2014 he is with the Research and Development Department, MACCON GmbH, Munich, Germany. His research interests include self-commissioning, characterisation and control of electrical drives.



CHRISTOPH M. HACKL (M'12-SM'16) was born in 1977 in Mannheim, Germany. After studying Electrical Engineering (with focus on mechatronics and systems and control) at Technical University of Munich (TUM), Germany and University of Wisconsin-Madison, USA, he received the B.Sc., Dipl.-Ing., and Dr.-Ing. (Ph.D.) degrees in Electrical Engineering in 2003, 2004 and 2012, respectively, from TUM. Since 2004, he has been teaching electrical drives, power electronics, and mechatronic & renewable energy systems. Since 2014, he has been

the head of the research group "Control of Renewable Energy Systems (CRES)" at TUM. In 2018, he became a Professor for Electrical Machines and Drives and the head of the "Laboratory for Mechatronic and Renewable Energy Systems (LMRES)" at the Hochschule München (HM) University of Applied Sciences, Germany. In 2019, he completed his habilitation on "Mechatronic and Renewable Energy Systems" and co-founded the research Institute for Sustainable Energy Systems (ISES) at HM, which he co-heads since then. His research interests include nonlinear, adaptive and optimal control and design of electrical drives, and mechatronic and renewable energy systems.

Anhand der gewonnenen nichtlinearen Flusskarten der Maschinen, ist es möglich, die Maschinenregler adaptiv anzupassen, um eine gleichbleibende Dynamik über alle Betriebspunkte zu erhalten. Zudem kann durch die Fluss- bzw. Induktivitätskarten, die Energieeffizienz gesteigert werden. Dies ist für beide Maschinen anhand der MTPA (maximum torque per ampere) Kurve dargestellt. Es ist zu sehen, dass durch die ANNs eine Steigerung des Drehmoments bei gleichem Strom von 10,8% für die SynRM und 7,1% für die IPMSM erreicht werden kann.

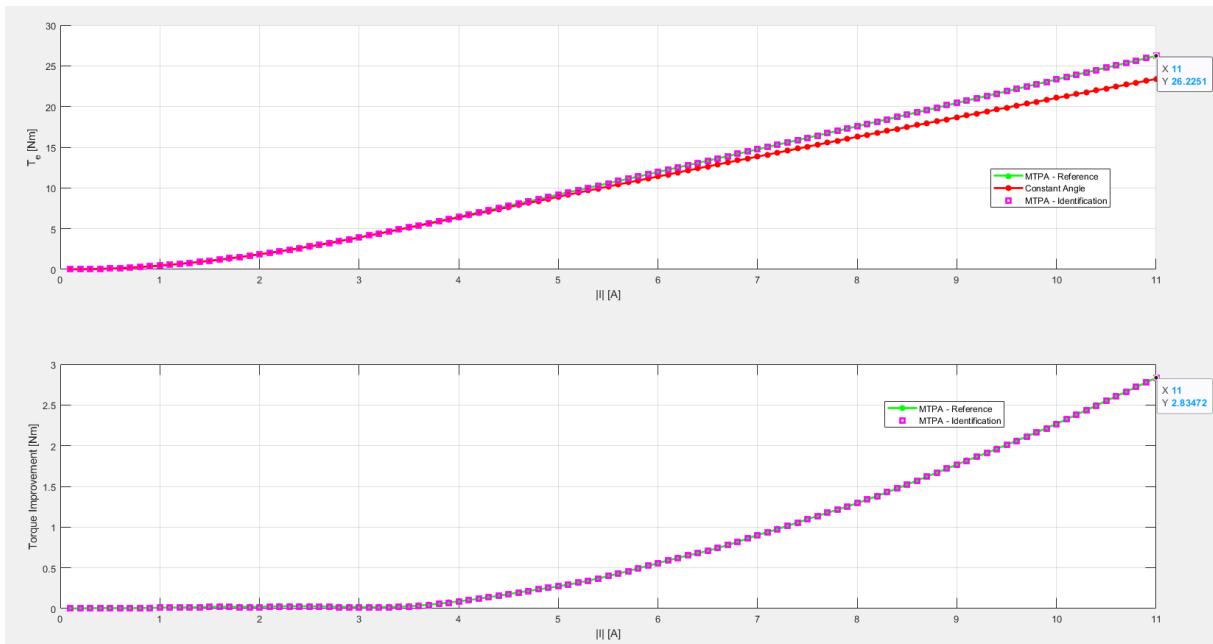


Figure 24: MTPA of the SynRM, maximum torque gain of 10.8% at peak current

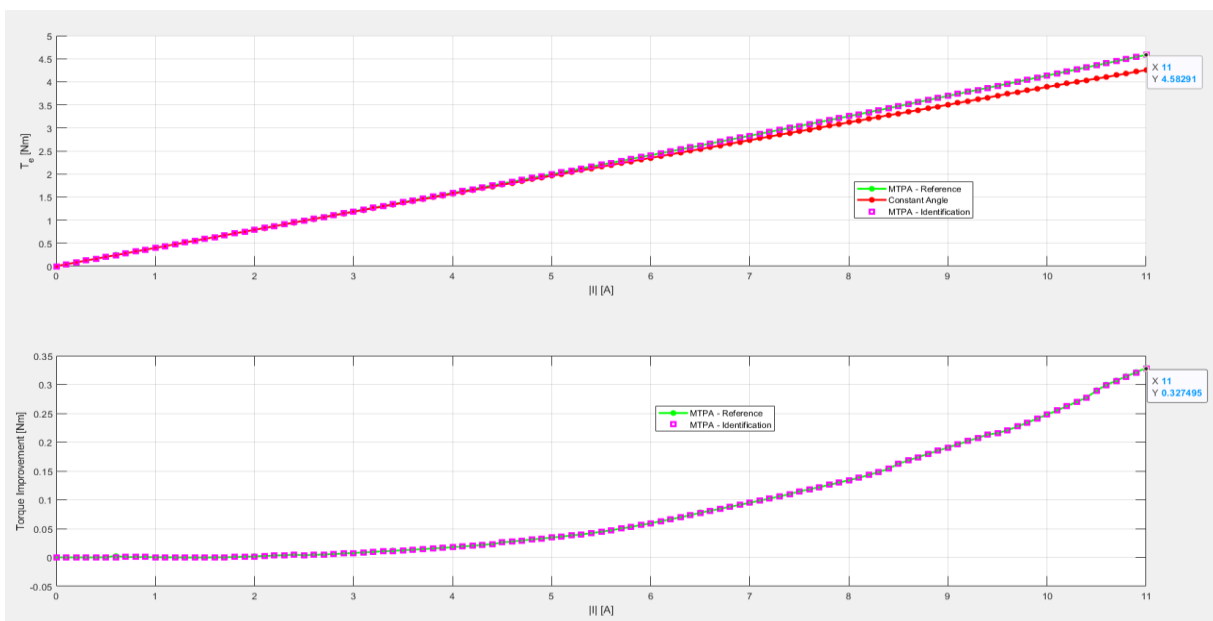


Figure 25 MTPA of the IPMSM, maximum torque gain of 7.1% at peak current

2.1.7 Kompensation von Subharmonischen für die geberlose Regelung

In der Regel wird zur effektiven Regelung elektrischer Synchronmaschinen die feldorientierte Regelung verwendet. Die dazu notwendigen Koordinatentransformationen von Strömen und Spannungen, von einem drei-phasigen Stator orientierten System in ein rotorfestes zwei Phasen System, benötigt die Informationen über die Rotorlage, welche durch einen mechanischen an der Rotorwelle montieren Positionsgeber gemessen wird. Dieser Geber kann durch intelligente Beobachter ersetzt werden, welche im Betrieb die Position schätzt. Dies hat verschiedene Vorteile, wie Kostenersparnis, Robustheit, Redundanz, etc. Durch das nichtlineare Verhalten der elektrischen Maschine und des Umrichters, kommt es zu Ungenauigkeiten bei der Positionsschätzung der Beobachter. Ziel der nachfolgenden Arbeit ist es, diese Schätzfehler durch künstliche neuronale Netzwerke – artificial neural networks (ANN) zu minimieren.

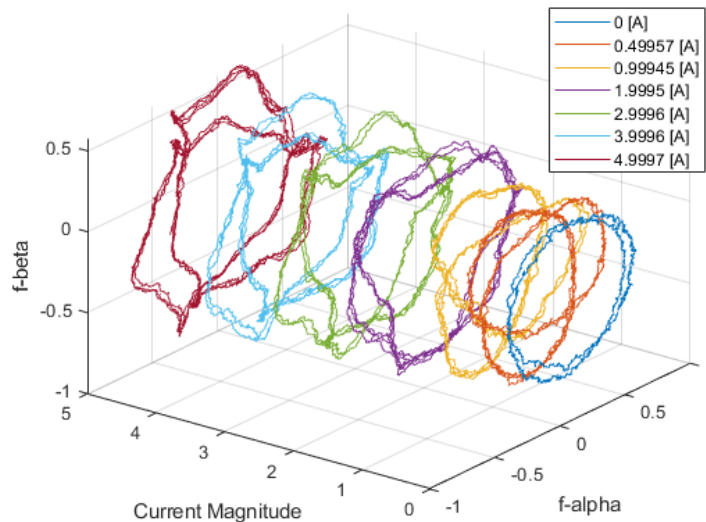
Das in dieser Arbeit verwendete Beobachterverfahren basiert auf der „Arbitrary Injection“ [4] Methode welche die Anisotropie der Maschine zur Lagebestimmung ausnutzt. Dabei werden zwei wesentliche Signale, f_{α} und f_{β} , durch Demodulation der Phasen-Spannung und -Ströme erzeugt und zur Positionsberechnung herangezogen (i.d.R. durch eine nachgeschaltete PLL (Phase-Locked-Loop)). Die direkte Umsetzung einer Positionsschätzung ist allerdings fehlerbehaftet, da f_{α} und f_{β} durch mehrere subharmonischen Komponenten überlagert sind. Dies ist auf Nichtlinearitäten in Phasen-Spannungen und -Strömen zurückzuführen, welche u.a. durch parasitäre Effekte im Umrichter und durch Sättigungseffekte in der Maschine entstehen. Um die Positionsschätzung zu verbessern, müssen diese Subharmonischen vor dem Betrieb der Maschine gemessen werden, um diese anschließend während des Betriebs zu kompensieren. Für die Kompensation können verschiedene Verfahren herangezogen werden, siehe z.B. [5]. Dabei können die harmonischen z.B. durch eine Fourier-Reihe dargestellt werden, um eine Kompensation während des Betriebs zu realisieren [5]

$$f_{\Sigma\alpha} = \sum_{k=-n}^n a_{2k} \cos(k\varphi_e) + b_{2k} \sin(2k\varphi_e)$$

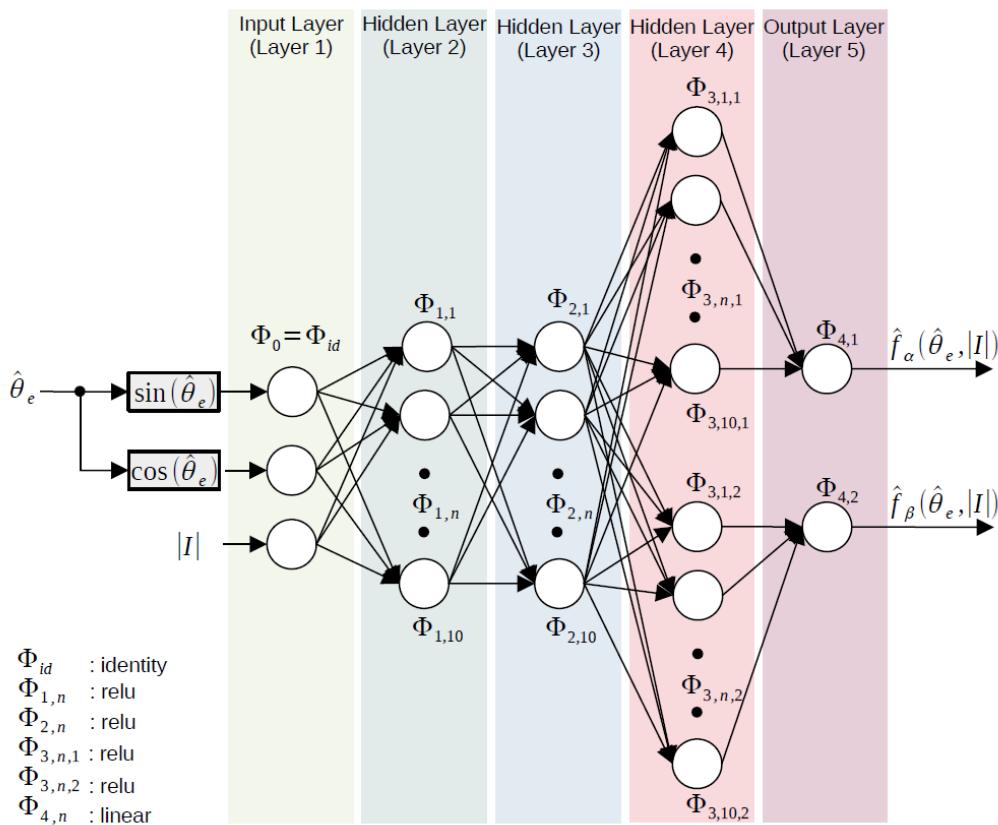
$$f_{\Sigma\beta} = \sum_{k=-n}^n -b_{2k} \cos(2k\varphi_e) + a_{2k} \sin(2k\varphi_e)$$

Die Koeffizienten der Fourier-Reihe sind wiederum eine Funktion des Stroms in der Maschine. Dies und die Abhängigkeit von zahlreichen trigonometrischen Funktionen macht eine Fourier-Modellierung wesentlich komplexer und für eine Implementierung in einem Mikrocontroller oder FPGA von Nachteil. In dieser Arbeit wurde daher ein neuronales Netzwerk mit Hilfe von Tensorflow entwickelt, welches zur Kompensation herangezogen wird.

Dieses Modell/ANN übernimmt die Abbildung der Funktionen f_{α} und f_{β} in Abhängigkeit des Stroms und der geschätzten Position (welche sich am Ausgang einer PLL ergibt), wie in der nachfolgenden Abbildung gezeigt ist.

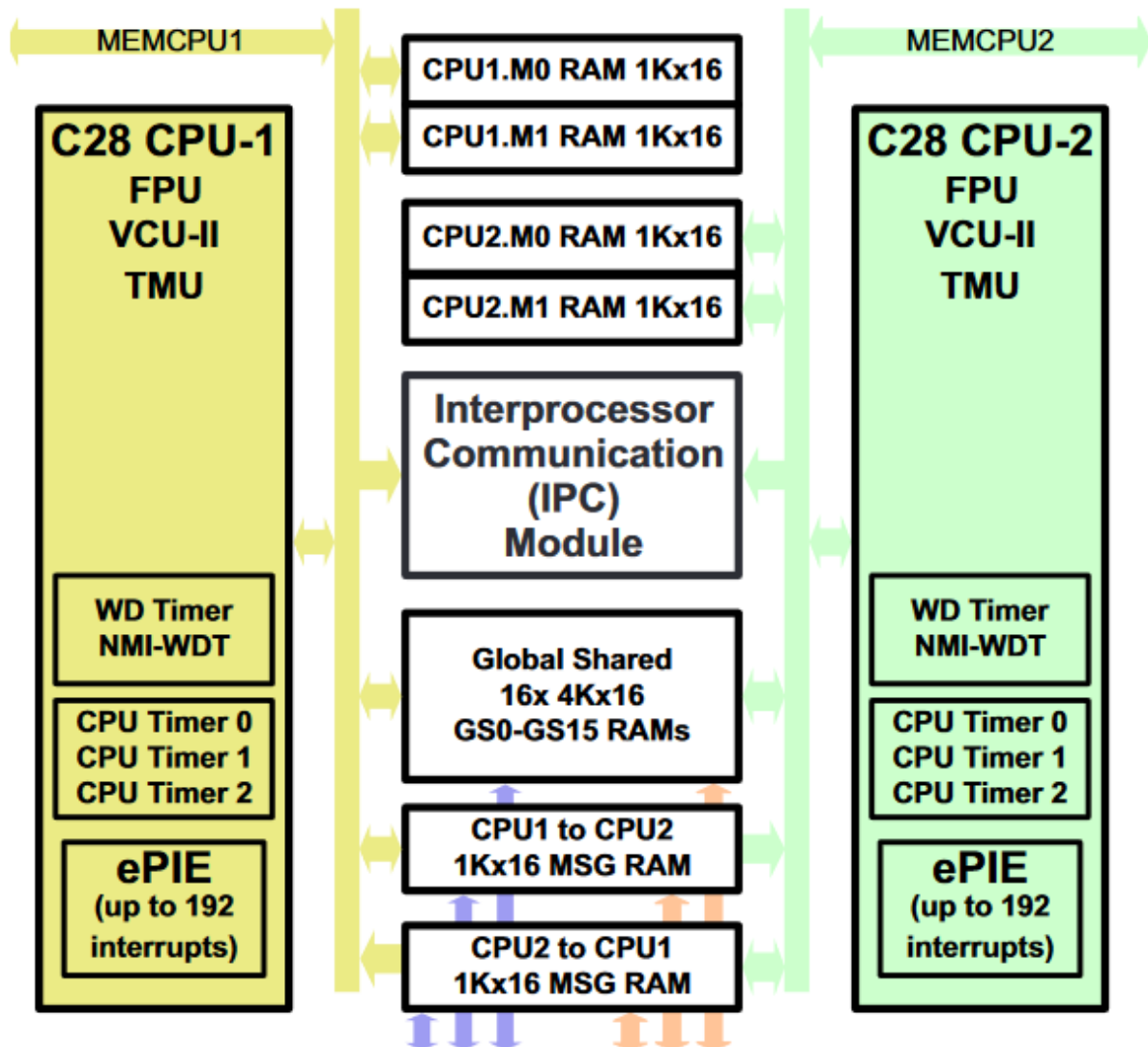


Das entwickelte ANN hat fünf Layer und verwendet „Relu“- und „linear“-Aktivierungsfunktionen und ist folgend dargestellt

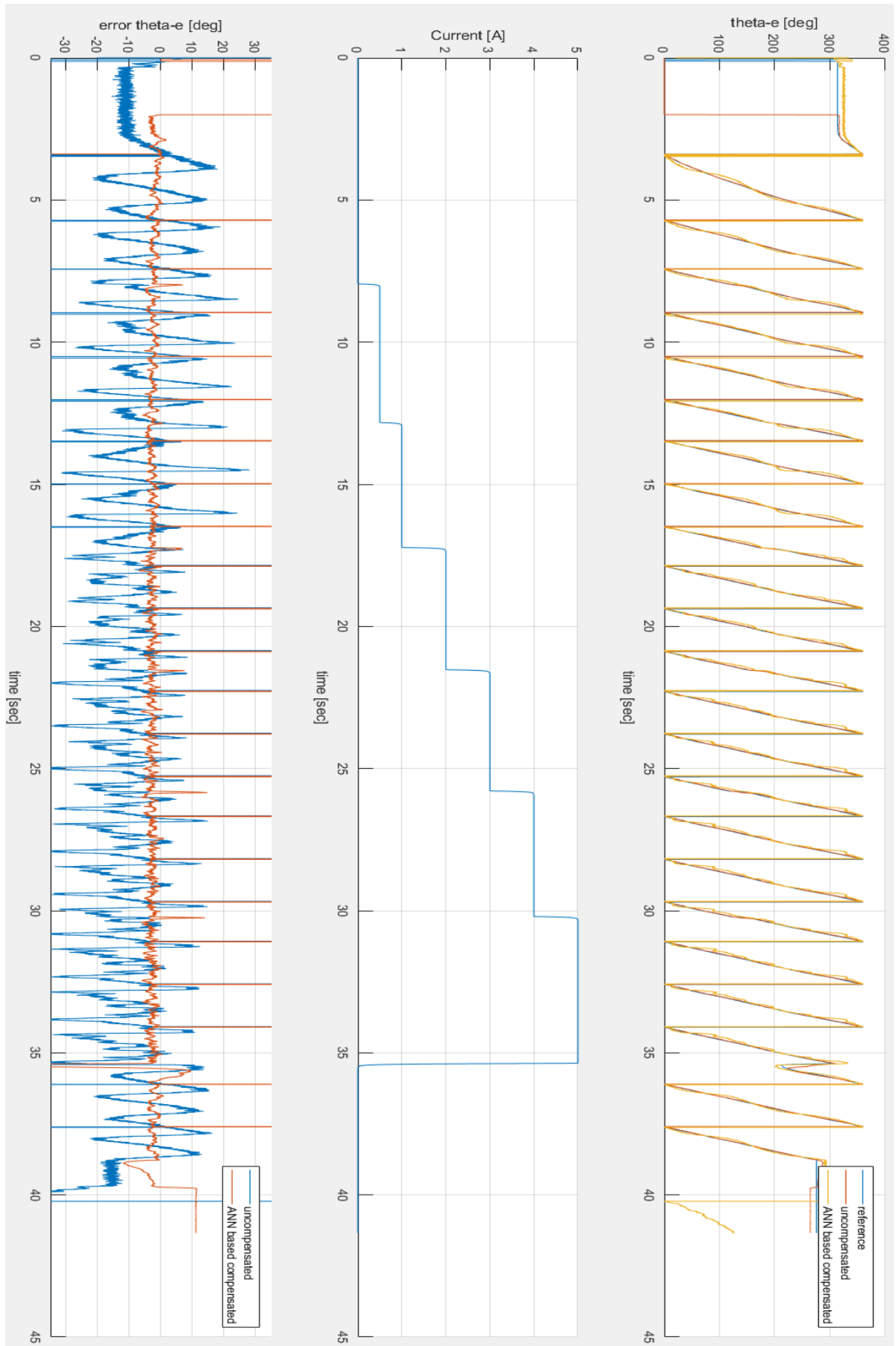


Nachdem die ANN basierte Kompensation mit Hilfe von Matlab/Simulink erfolgreich getestet worden ist, wurde die Kompensation während des Betriebs an Prüfstand Nr. 2 ebenfalls erfolgreich in Betrieb genommen.

Auf dem „dual core microcontroller“, wurde die Motorregelung auf CPU-1 durchgeführt und das ANN auf CPU-2. Die Kommunikation zwischen beiden CPUs wurde durch IPC-Module realisiert. Die Eingangswerte des ANN von CPU-1 zu CPU-2 und die Ausgänge von des ANN von CPU-2 zu CPU-1.



Die Messergebnisse sind folgend dargestellt und zeigen das eine Kompensation erfolgreich durchgeführt werden kann.



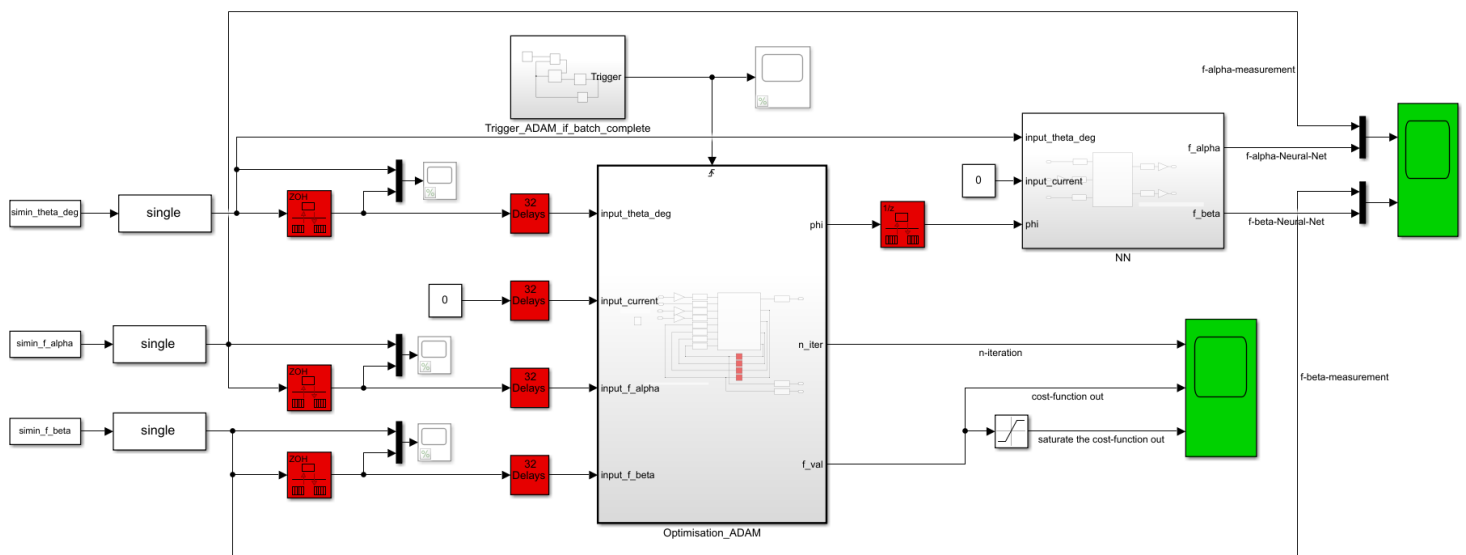
2.1.8 Onlineoptimierung der Positionsfehler-ANNs

In Kapitel 2.1.7 wurde ein Verfahren zur Kompensation der Subharmonischen einer sensorlosen Regelung vorgestellt. Dieses basiert darauf, dass die Subharmonischen als Funktion der tatsächlichen Position bekannt sind. Die Messung der Subharmonischen erfolgte daher durch einen Referenzgeber zur Positionserfassung. Weiterhin wird für die Messung eine Lastmaschine verwendet, um unterschiedliche Stromamplituden zu vermessen. Im Idealfall, soll auf eine Lastmaschine und einen zusätzlichen Positionsgeber verzichtet werden. Im genannten Kapitel wurde das ANN durch eine offline Optimierung Mittels der Software Tensorflow durchgeführt. Auch dies sollte idealerweise auf dem Embedded System geschehen.

Für die Optimierung des ANN in Tensorflow wurde als Optimierungsalgorithmus der „stochastic gradient decent“ Algorithmus „ADAM“ verwendet. Ziel ist es, diesen Algorithmus auf einem „embedded system“ anzuwenden. Daher wurde in einem ersten Schritt, dieser Algorithmus in Matlab implementiert und das ANN mit denselben Messwerten wie in Tensorflow trainiert. Nach erfolgreicher Implementierung wurde der ADAM-Algorithmus in Simulink implementiert. Die Ergebnisse sind in folgendem Kapitel dargestellt.

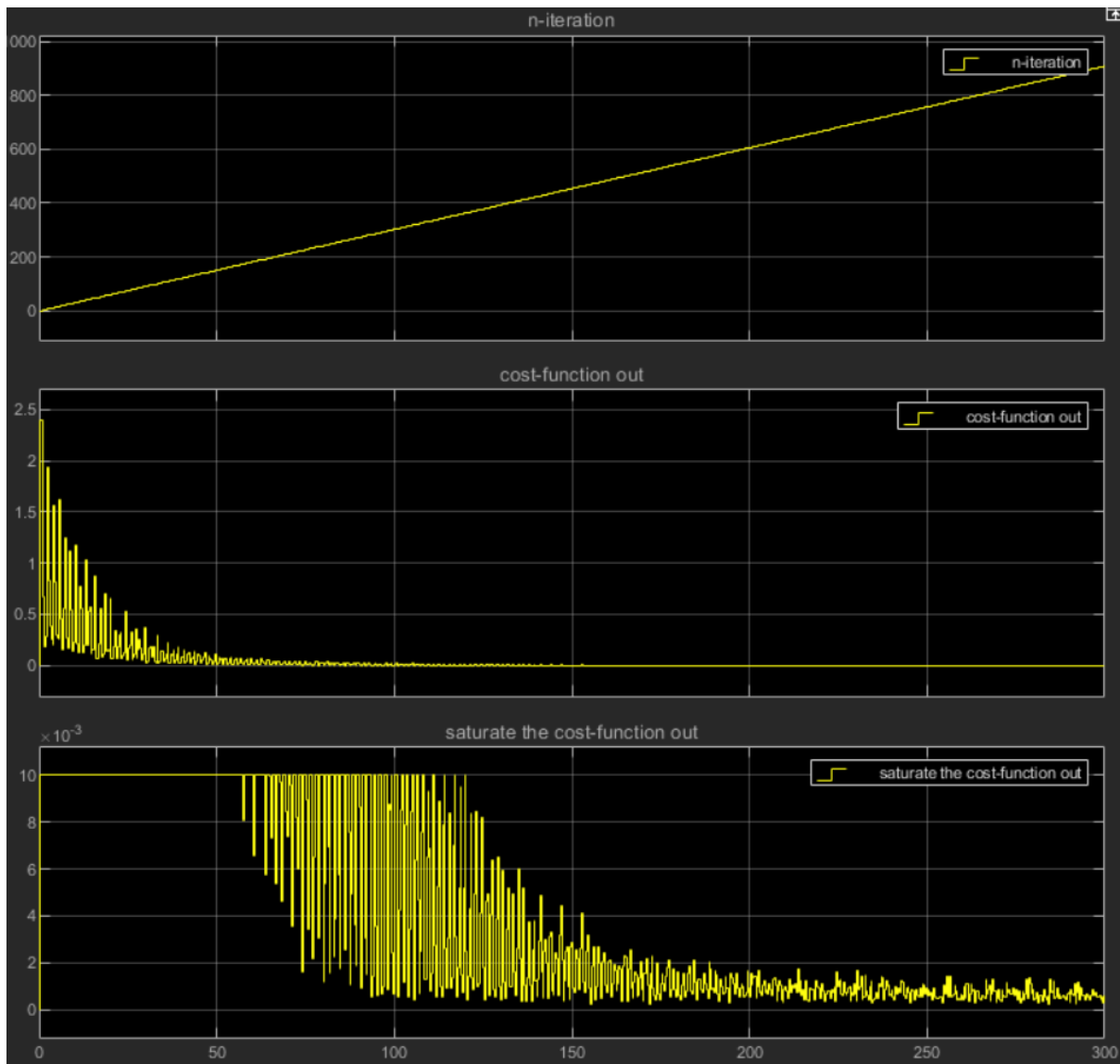
2.1.9 Ergebnisse der Optimierung in Simulink

Folgende Abbildung zeigt das Top-level Model der Simulink Implementierung

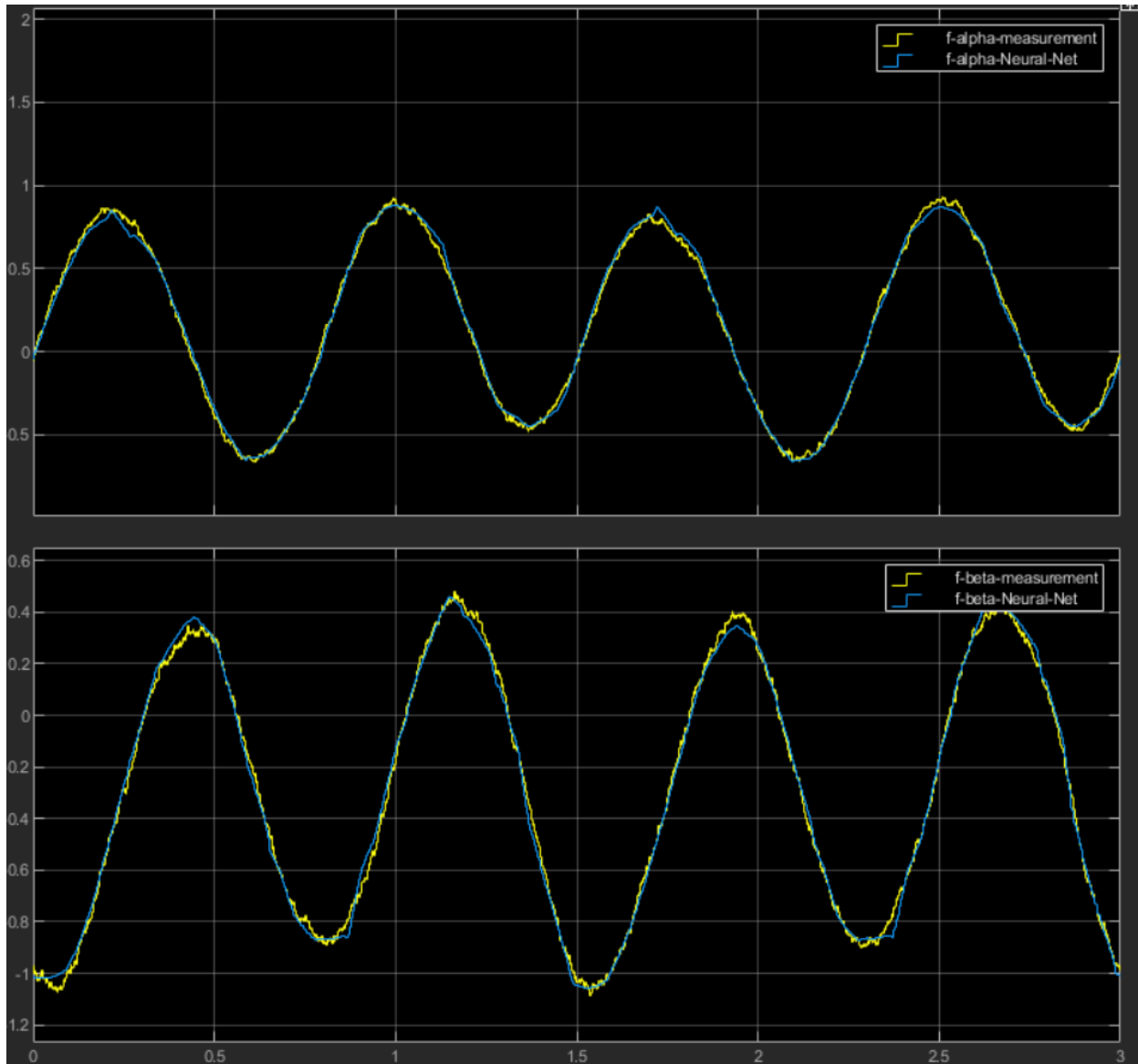


Das Subsystem „Optimisation_ADAM“ beinhaltet den ADAM Algorithmus sowie das ANN dessen Parameter während der Laufzeit trainiert werden. Das Subsystem „NN“ beinhaltet ein redundantes ANN, dessen Parameter durch das Subsystem „Optimisation_ADAM“ in jedem Rechenschritt bereitgestellt werden. Dadurch ist die online Optimierung in Echtzeit verfolgbar. Die Ergebnisse dieser Optimierung sind folgend über die Zeit in Sekunden dargestellt. In diesem Versuch ist eine konstante Last angelegt. Das obere Bild zeigt die

momentane Iteration des Optimierungs-Algorithmus ADAM. Das mittlere Bild ist der zur Iteration berechnete Wert der Kostenfunktion. Das untere Bild zeigt das mittlere Bild mit einer Begrenzung in der Y-Achse auf $10e-3$.



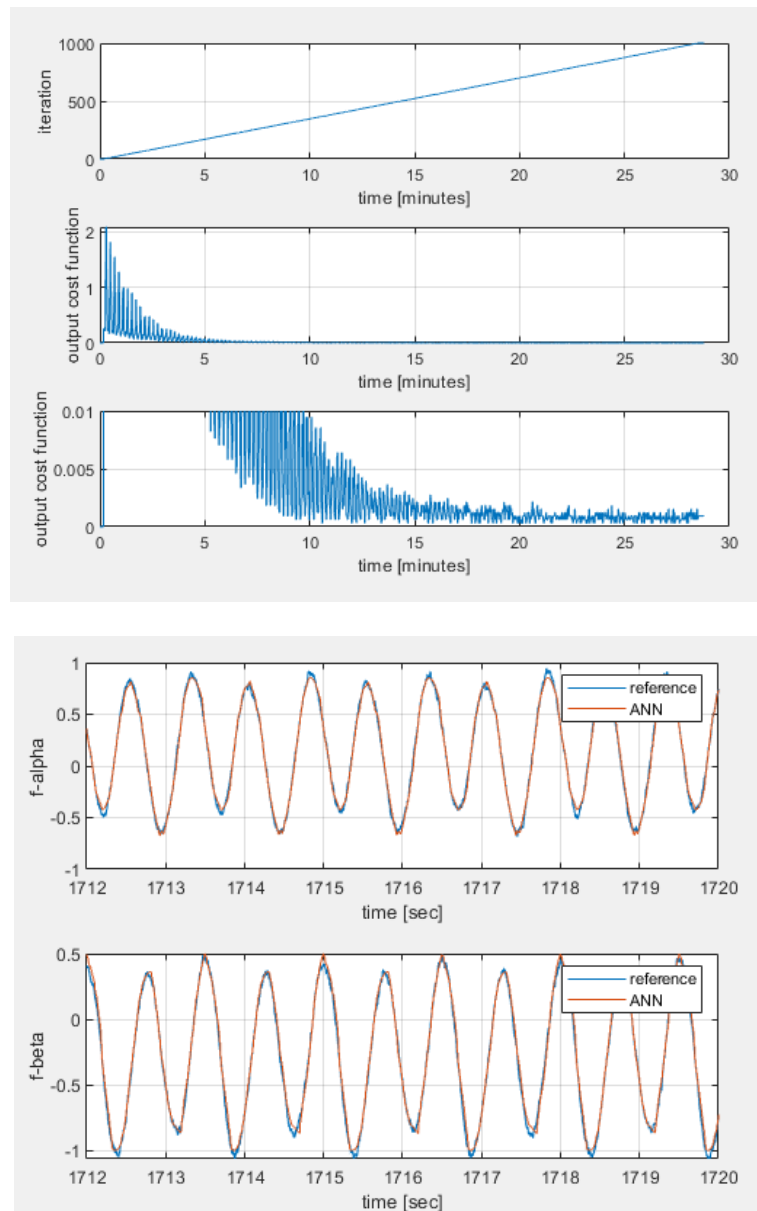
Die Grafiken zeigen eine stetige Verringerung des Fehlers zwischen ANN und Messung. Ab ca. 250 Iterationen ist keine wesentliche Verbesserung mehr zu erkennen. An dem graphisch dargestellten letzten Schritt der Iteration ist in der nachfolgenden Grafik der Ausgang des ANN-Modells zu der Messung über die Zeit von 3 Sekunden dargestellt. Es ist zu erkennen, dass das ANN eine gute Abbildung der Messung wiedergibt.



2.1.9.1 Ergebnisse der Onlineoptimierung auf dem C2000 (Prüfstand 2)

In dieser Untersuchung soll das Laufzeitverhalten der Optimierung und des ANN auf einem Embedded System untersucht werden (Motorprüfstands 2). Die Motorregelung und Positionbeobachter (f_{α} , f_{β}) ist auf CPU1 implementiert, die Optimierung findet auf CPU 2 statt des. Die Ergebnisse sind auf folgenden Grafiken dargestellt. Für die Optimierung wurden mini-batches von 32 gewählt. f_{α} und f_{β} wurden simultan mit 100Hz gemessen. Die Maschine bewegte sich konstant mit 50rpm mechanisch. Eine Iteration dauerte ca. 1.8 Sekunden (wovon $32/100\text{Hz} = 0.32$ Sekunden für das sampling anfallen).

Das obere Bild zeigt die momentane Iteration des Optimierungs-Algorithmus ADAM auf dem CPU. Das mittlere Bild ist der zur Iteration berechnete Wert der Kostenfunktion. Das untere Bild zeigt das mittlere Bild mit einer Begrenzung in der Y-Achse auf $10e-3$.

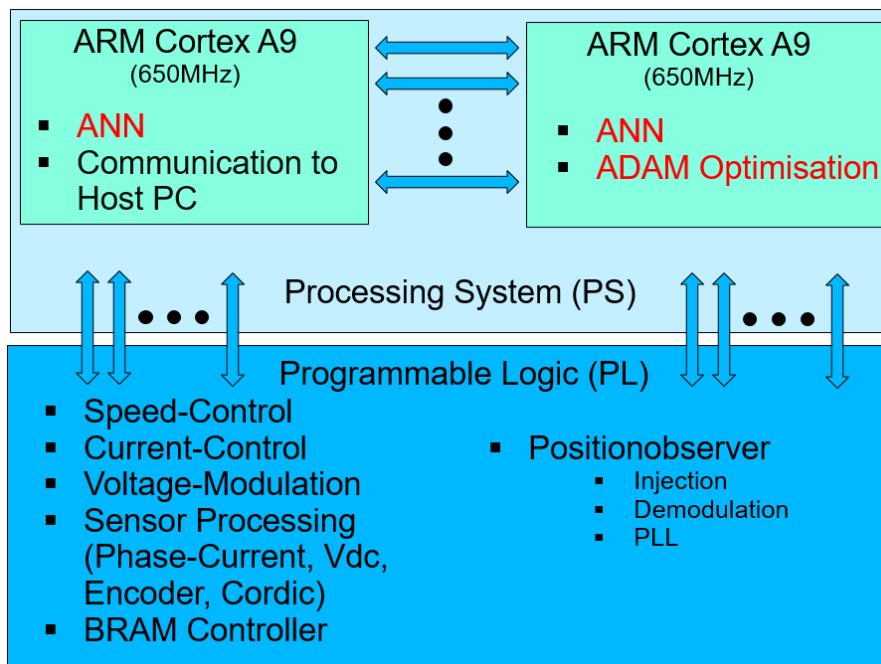


Es ist eine ähnliche Konvergenz wie unter Kapitel 6.1 festzustellen mit der Ausnahme, dass wesentlich mehr Zeit (28min vs. 5min) benötigt wird, um einen ähnlichen Wert der Kostenfunktion zu erreichen. Dies ist durch die limitierte Rechenleistung des Mikrocontrollers mit 200MHz zurückzuführen im Vergleich zu der Rechenleistung der Simulink Simulation auf dem PC. Der Vergleich der Messwerte zu den Ausgängen des ANN zeigen in der nachfolgenden Grafik eine zufriedenstellende Übereinstimmung. Eine weitere Untersuchung auf einer leistungsfähigeren Recheneinheit wie z.B. einem SoC (Motorprüfstand 3) wird im folgenden Kapitel gezeigt.

2.1.9.2 Ergebnisse der Onlineoptimierung auf dem Zynq7000 (Prüfstand 3)

Die Implementierung ist folgend verteilt:

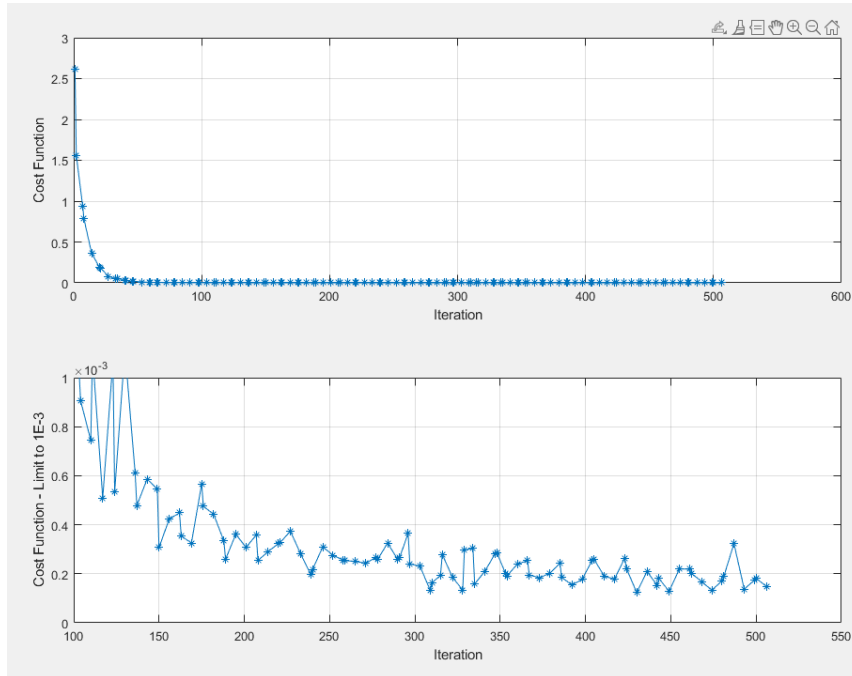
Die Strom- und Geschwindigkeitsregelung sowie Spannungsmodulation und Sensorauswertung wurde im FPGA umgesetzt. Die Kommunikation zu einem Host-PC ist durch ARM0 realisiert. Zudem wird auf ARM0 der Ausgang des ANN berechnet, basierend auf den geschätzten Modellparametern der ADAM-Optimierung. Die Optimierung der ANN-Parameter passiert auf ARM1. Beide Prozessoren teilen sich einen gemeinsamen Speicherplatz, auf welchem die Modellparameter ausgetauscht werden.



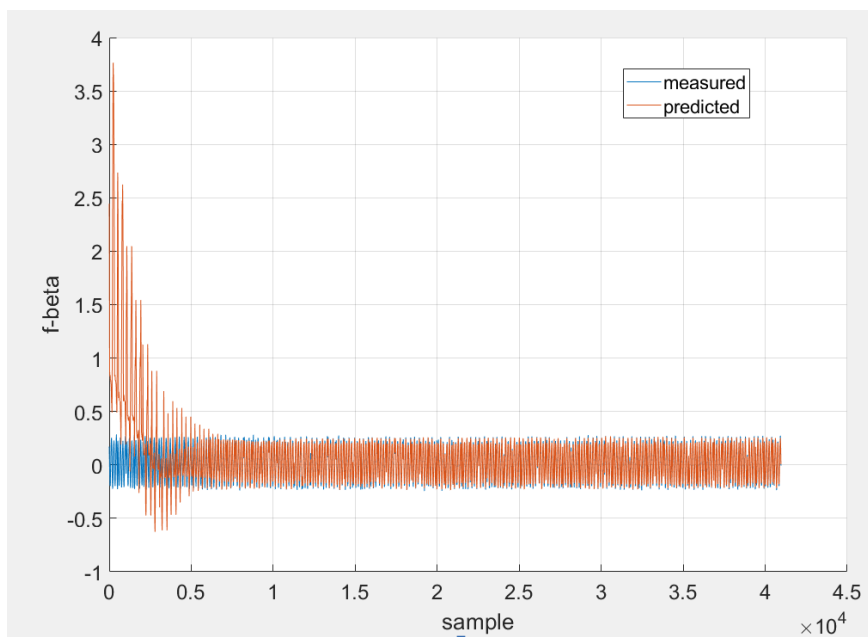
Nachfolgend sie die Implementierungsergebnisse zu sehen. Es wurde als Motor der NEMA 23 Motor [3] gewählt, um einen Vergleich zu Kapitel 4.2 zu schaffen. Für die Optimierung wurden mini-batches von 32 gewählt. F-alpha und f-beta wurden simultan mit 100Hz gemessen. Die Maschine bewegte sich konstant mit 50rpm mechanisch. Eine Iteration dauerte ca. 1 Sekunde (wovon $32/100\text{Hz} = 0.32$ Sekunden für das sampling anfallen).

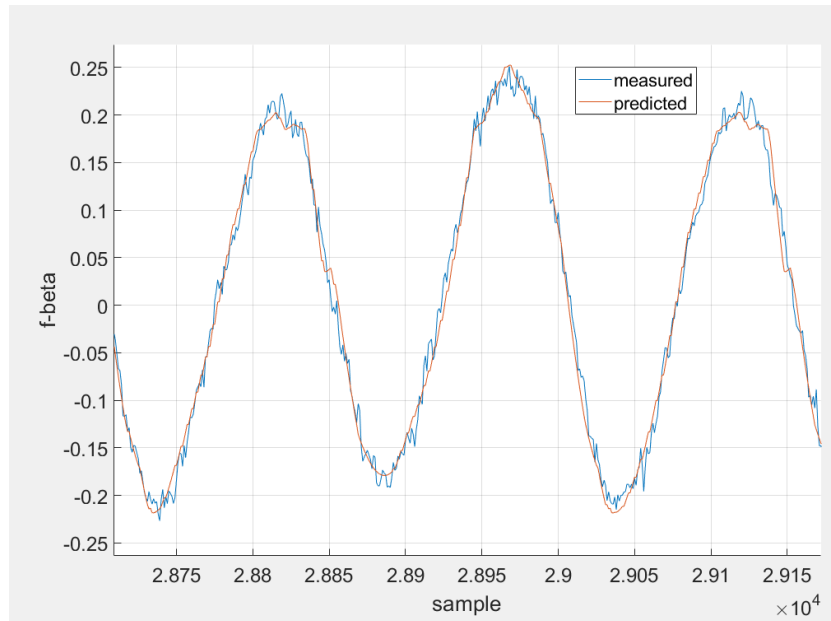
Ergebnisse der Optimierung von f-beta:

Die nachfolgende Abbildung zeigt online die Kostenfunktion der ADAM-Optimierung über die Iteration. Es ist eine zufriedenstellende Konvergenz zu sehen.



Die folgenden Abbildungen zeigen die gemessene Größe im Vergleich zum Ausgang der ANN während der Onlineoptimierung. Es ist ein gutes Einschwingverhalten und eine ausreichende Genauigkeit zu erkennen. (Vergleich zu der C2000 Optimierung).

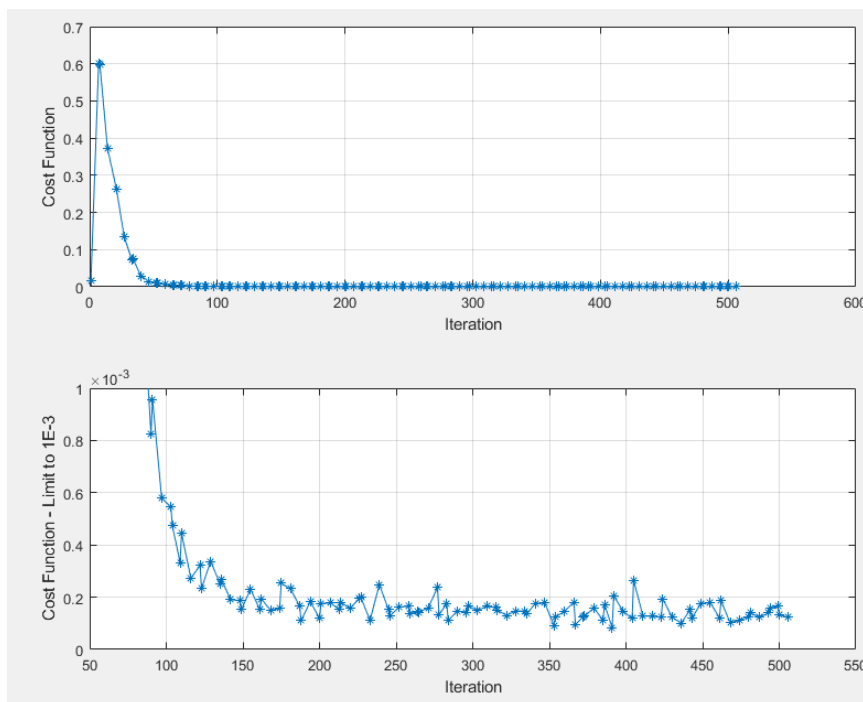




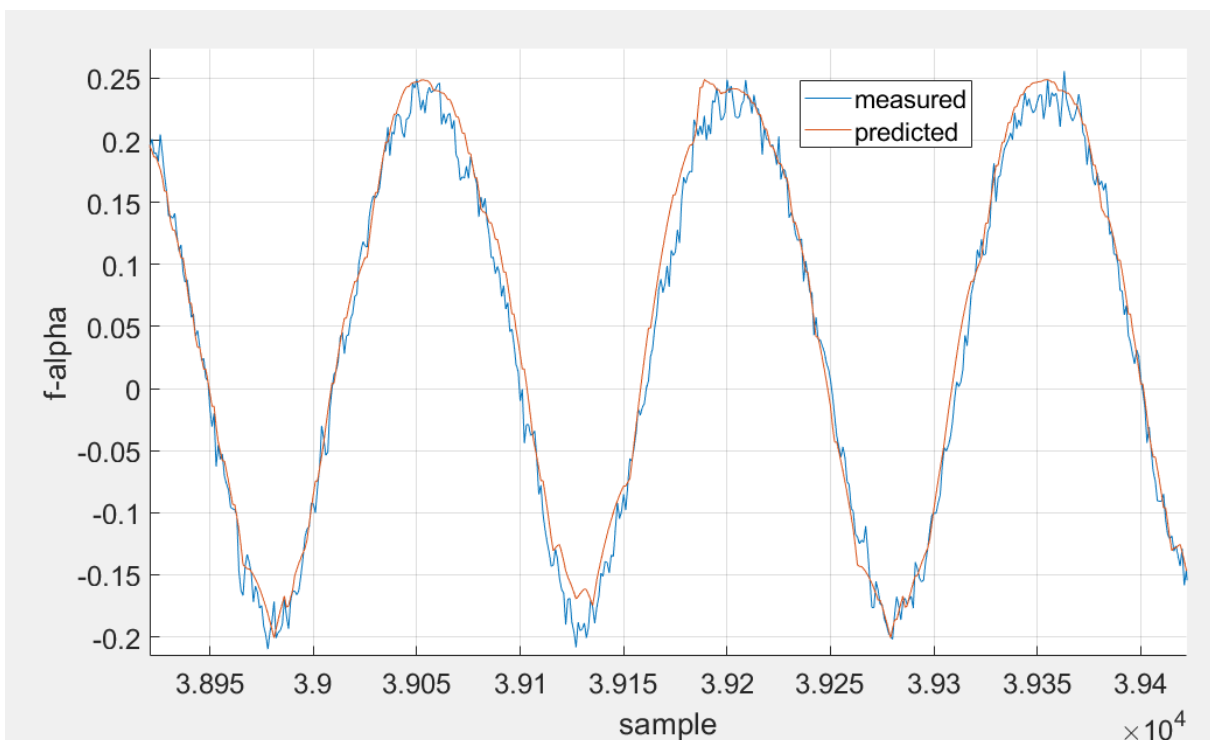
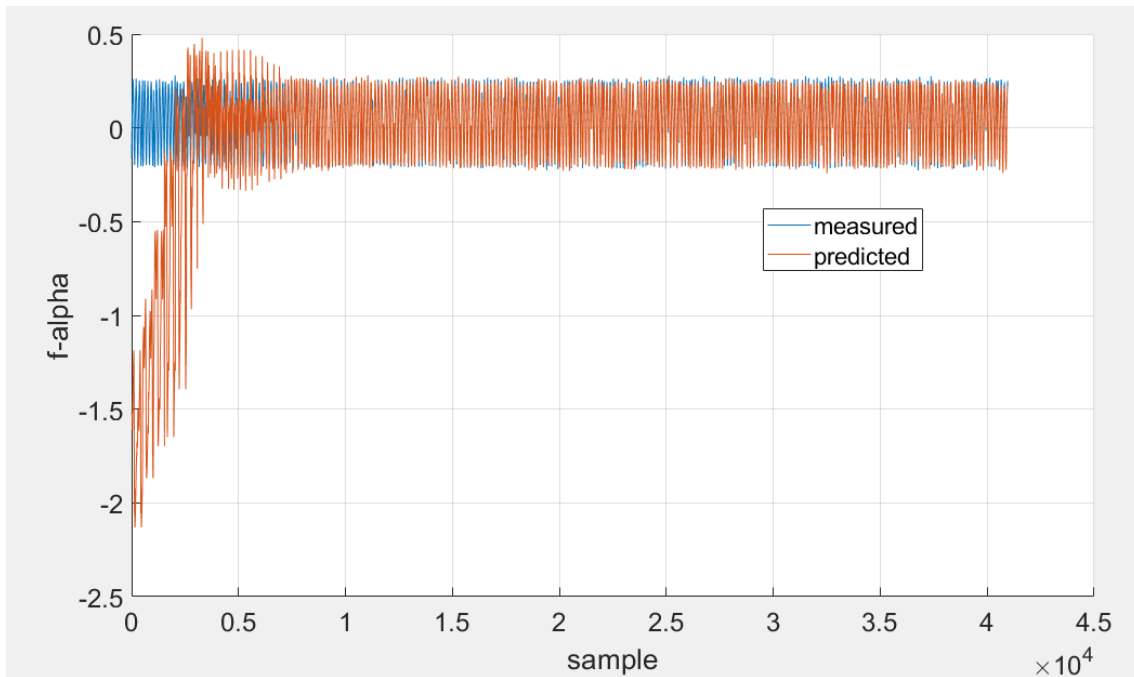
Die Kostenfunktion berechnet den Fehler für f -alpha und f -beta. Allerdings konnte nur die Kostenfunktion, Iteration sowie gemessenes f -alpha und geschätztes f -alpha aufgezeichnet werden. Daher wurde in einer weiteren Optimierung f -beta aufgezeichnet (siehe folgend). Es weicht daher die Darstellung der Kostenkurve von dem vorher gezeigten Ergebnis ab.

Ergebnisse der Optimierung von f -alpha:

Die nachfolgende Abbildung zeigt online die Kostenfunktion der ADAM-Optimierung über die Iteration. Es ist eine zufriedenstellende Konvergenz zu sehen.



Die folgenden Abbildungen zeigen die gemessene Größe im Vergleich zum Ausgang der ANN während der Onlineoptimierung. Es ist ein gutes Einschwingverhalten und eine ausreichende Genauigkeit zu erkennen. (Vergleich zu der C2000 Optimierung).



Für $f\text{-alpha}$ und $f\text{-beta}$ wurde ein Einschwingen nach ca. 150samples festgestellt. Das entspricht einer Zeit für die Optimierung von ca. 2.5min.

2.1.10 Zusammenfassung

Der Fokus der im Projekt entwickelten Methoden lag auf Anwendungen im Bereich dreiphasiger Synchronmaschinen mit Zwei-Level-Spannungsumrichtern. Neben der Durchführung von Voruntersuchungen und Literaturstudien, wurden verschiedene Verfahren entwickelt, welche u.a. künstliche neuronale Netzwerke (Artificial-Neural-Networks - ANNs) einsetzen, die zur Modellierung und Identifikation von nichtlinearen Effekten der Maschine und des Umrichters dienen. Diese ANNs wurden durch nichtlineare Optimierungsverfahren und durch KI-basierte Algorithmen optimiert. Die entwickelten Methoden unterstützen eine automatisierte und effiziente Inbetriebnahme von elektrischen Maschinen, was u.a. zu höherer Regel-Genauigkeit und Energieeffizienz während des Betriebs des Motors führen kann.

Es wurden zwei Hauptverfahren entwickelt und in Simulation sowie an unterschiedlichen Prüfständen validiert. Das erste Verfahren dient zur Selbstidentifikation von Synchronmaschinen und des Umrichters. Das Verfahren ermöglicht durch den Einsatz von verschiedenen entwickelten neuronalen Netzwerken und einer Particle-Swarm-Optimierung das gleichzeitige modellieren und identifizieren von nichtlinearen Sättigungen der elektrischen Maschine durch Fluss-Kurven, Fluss-Karten, nichtlinearen Induktivitäten und des Widerstandes. Zeitgleich werden die nichtlinearen Spannungsfehler des Umrichters durch ein weiteres neuronales Netzwerk modelliert und identifiziert.

Ein weiteres zentrales Ergebnis des Projekts ist die Kompensation von Subharmonischen für die geberlose Regelung von Synchronmaschinen. Hierfür wurden künstliche neuronale Netzwerke entwickelt, um die Schätzfehler der Rotorposition zu minimieren, die durch nichtlineare Effekte der Maschine entstehen. Diese Genauigkeit der ANNs wurden zunächst in Matlab/Simulink und anschließend an zwei Prüfständen erfolgreich getestet.

Insgesamt zeigen die entwickelten Identifikations-, Modellierungs-, Optimierungs- und Kompensations-Verfahren, gestützt durch KI-Methoden, dass die Inbetriebnahme von Synchronmaschinen wesentlich erleichtert werden kann. Das Projekt leistet somit einen Beitrag zur Weiterentwicklung moderner elektrischer Antriebstechnologien und ihrer Anwendungen in der Industrie.

2.2 Wichtigste Positionen des zahlenmäßigen Nachweises

Die wichtigste Position sind die Personalkosten. Daher ist ein überwiegender Teil der entstandenen Kosten auf Arbeiten in der Forschung und Entwicklung verschiedener Algorithmen und KI-Techniken zurückzuführen.

2.3 Notwendigkeit und Angemessenheit der geleisteten Arbeit

MACCON hat durch die Durchführung des KIRA-Projekts seine Fähigkeiten im Bereich der KI-unterstützten Ansteuerungsalgorithmik für MACCON Steuerungseinheiten erweitert. Der steigende Bedarf von leistungsoptimierten Motoren und Steuerungseinheiten bietet gerade für KMUs wie MACCON Herausforderungen und Chancen zugleich. Die manuelle Berücksichtigung von Maschinen- und Umrücker-Nichtlinearitäten durch Messungen und manuellen Einstellungen ist durch den zeitlichen Aufwand für einen Kunden kaum tragbar. Diese Informationen sind allerdings für einen energieoptimalen und optimierten Regler-Betrieb notwendig. Um trotzdem den Bedarf nach optimierten Antrieben zukünftig zu befriedigen, wird angestrebt, KIRA-Techniken in MACCON Produkte langfristig zu integrieren und damit dem Kunden der Zukunft Kosten und Zeit zu sparen. Für KMUs bilden die Komplexität und der Umfang eines solchen sehr zeitintensiven Vorhabens der automatischen und KI-gestützten Optimierung durch die verschiedenen Modellierungstechniken sowie Optimierungs-Algorithmen ohne Förderungen eine kaum überwindbare Barriere.

2.4 Eingehende Darstellung des voraussichtlichen Nutzens, insbesondere der Verwendbarkeit des Ergebnisses im Sinne des fortgeschriebenen Verwertungsplans

Im Rahmen des KIRA-Projekts wurden neue Verfahren entwickelt, die künftig in die Motorcontroller von MACCON integriert werden. Dazu zählen insbesondere Techniken zur Selbstinbetriebnahme elektrischer Maschinen sowie Methoden der geberlosen Regelung. Durch den Einsatz von Algorithmen zur Modelloptimierung können Motoren automatisch identifiziert und die Regelung auf dieser Basis selbstständig optimiert werden.

2.5 Eingehende Darstellung des während der Durchführung des Vorhabens dem Zuwendungsempfänger bekannt gewordenen Fortschritts auf diesem Gebiet bei anderen Stellen

Uns sind keine weiteren Fortschritte auf den von uns bearbeiteten Anwendungen bekannt.

2.6 Erfolgte oder geplante Veröffentlichungen der Ergebnisse

Im Rahmen des KIRA-Projekts wurde folgendes in dem IEEE Journal „*Transactions on Energy Conversion*“ publiziert:

S. Wiedemann and C. M. Hackl, "Simultaneous Identification of Inverter and Machine Nonlinearities for Self-Commissioning of Electrical Synchronous Machine Drives," in *IEEE Transactions on Energy Conversion*, vol. 38, no. 3, pp. 1767-1780, Sept. 2023, doi: 10.1109/TEC.2023.3263353

3. Literaturverzeichnis

[1] Bundesministerium für Verkehr, Bau und Stadtentwicklung, „Verkehr in Zahlen,“ 2010.

[2] S. Wiedemann. Self-Identification and Automatic Tuning of Synchronous Machine Drives. PhD thesis, Technical University of Munich, 2020.

[3] Application Report SPRACB8A–March 2018–Revised March 2018 “Dual Motor Control Using FCL and Performance Analysis Using SFRA on TMS320F28379D LaunchPad”

[4] D. Paulus, P. Landsmann and R. Kennel, "General arbitrary injection approach for synchronous machines," *2013 IEEE International Symposium on Sensorless Control for Electrical Drives and Predictive Control of Electrical Drives and Power Electronics (SLED/PRECEDE)*, Munich, Germany, 2013, pp. 1-6, doi: 10.1109/SLED-PRECEDE.2013.6684509.

[5] D. Paulus, P. Landsmann, S. Kuehl and R. Kennel, "Arbitrary injection for permanent magnet synchronous machines with multiple saliencies," *2013 IEEE Energy Conversion Congress and Exposition*, Denver, CO, USA, 2013, pp. 511-517, doi: 10.1109/ECCE.2013.6646744.

[6] Vorhabensbeschreibung: KIRA - KI-Methoden zur optimierten Regelung elektrischer Traktionsantriebe

[7] S. Wiedemann and C. M. Hackl, "Simultaneous Identification of Inverter and Machine Nonlinearities for Self-Commissioning of Electrical Synchronous Machine Drives," in *IEEE Transactions on Energy Conversion*, vol. 38, no. 3, pp. 1767-1780, Sept. 2023, doi: 10.1109/TEC.2023.3263353