

ACTA DE EVALUACIÓN DE LA TESIS DOCTORAL

Año académico 2018/19

DOCTORANDO: **PERPETUO CORRÊA, TOMÁS**
D.N.I./PASAPORTE: ****4691

PROGRAMA DE DOCTORADO: **D441-ELECTRÓNICA: SISTEMAS ELECTRÓNICAS AVANZADOS. SISTEMAS INTELIGENTES**

DPTO. COORDINADOR DEL PROGRAMA: **ELECTRÓNICA**

TITULACIÓN DE DOCTOR EN: **DOCTOR/A POR LA UNIVERSIDAD DE ALCALÁ**

En el día de hoy 09/04/19, reunido el tribunal de evaluación nombrado por la Comisión de Estudios Oficiales de Posgrado y Doctorado de la Universidad y constituido por los miembros que suscriben la presente Acta, el aspirante defendió su Tesis Doctoral, elaborada bajo la dirección de **EMILIO JOSÉ BUENO PEÑA // FRANCISCO JAVIER RODRIGUEZ SÁNCHEZ**.

Sobre el siguiente tema: *DIGITAL COMMUNICATIONS FOR THE CONTROL OF MODULAR MULTILEVEL CONVERTERS*

Finalizada la defensa y discusión de la tesis, el tribunal acordó otorgar la CALIFICACIÓN GLOBAL¹ de (no apto, aprobado, notable y sobresaliente): SOBRESALIENTE

Alcalá de Henares, 9 de Abril de 2019

EL PRESIDENTE



Fdo.: ROBERTO ALVES

EL SECRETARIO



Fdo.: RAUL MATEOS GIL

EL VOCAL



Fdo.: LUS ALMEIDA

Con fecha 24 de abril de 2019 la Comisión Delegada de la Comisión de Estudios Oficiales de Posgrado, a la vista de los votos emitidos de manera anónima por el tribunal que ha juzgado la tesis, resuelve:

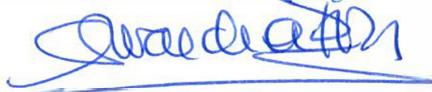
- Conceder la Mención de "Cum Laude"
 No conceder la Mención de "Cum Laude"

FIRMA DEL ALUMNO,



Fdo.: PERPETUO CORRÊA, TOMÁS

La Secretaria de la Comisión Delegada



¹ La calificación podrá ser "no apto" "aprobado" "notable" y "sobresaliente". El tribunal podrá otorgar la mención de "cum laude" si la calificación global es de sobresaliente y se emite en tal sentido el voto secreto positivo por unanimidad.

INCIDENCIAS / OBSERVACIONES:



Universidad
de Alcalá

COMISIÓN DE ESTUDIOS OFICIALES
DE POSGRADO Y DOCTORADO

En aplicación del art. 14.7 del RD. 99/2011 y el art. 14 del Reglamento de Elaboración, Autorización y Defensa de la Tesis Doctoral, la Comisión Delegada de la Comisión de Estudios Oficiales de Posgrado y Doctorado, en sesión pública de fecha 24 de abril, procedió al escrutinio de los votos emitidos por los miembros del tribunal de la tesis defendida por *PERPETUO CORRÊA, TOMÁS*, el día 9 de abril de 2019, titulada *DIGITAL COMMUNICATIONS FOR THE CONTROL OF MODULAR MULTILEVEL CONVERTERS*, para determinar si a la misma se le concede la mención "cum laude", no habiendo obtenido la unanimidad de los miembros del Tribunal.

Por lo tanto, la Comisión de Estudios Oficiales de Posgrado **resuelve no otorgar la Mención de "cum laude"** a dicha Tesis.

Alcalá de Henares, 24 de abril de 2019

EL VICERRECTOR DE INVESTIGACIÓN Y TRANSFERENCIA



F. Javier de la Mata

F. Javier de la Mata de la Mata

Copia por e-mail a:

Doctorando: PERPETUO CORRÊA, TOMÁS

Secretario del Tribunal: RAUL MATEOS GIL

Directores de Tesis: EMILIO JOSÉ BUENO PEÑA//FRANCISCO JAVIER RODRIGUEZ SÁNCHEZ



Universidad
de Alcalá

ESCUELA DE DOCTORADO
Servicio de Estudios Oficiales de
Posgrado

DILIGENCIA DE DEPÓSITO DE TESIS.

Comprobado que el expediente académico de D./D^a _____
reúne los requisitos exigidos para la presentación de la Tesis, de acuerdo a la normativa vigente, y habiendo
presentado la misma en formato: soporte electrónico impreso en papel, para el depósito de la
misma, en el Servicio de Estudios Oficiales de Posgrado, con el nº de páginas: _____ se procede, con
fecha de hoy a registrar el depósito de la tesis.

Alcalá de Henares a _____ de _____ de 20 _____



Fdo. El Funcionario



Universidad
de Alcalá

PhD. Program in Electronics: Advanced Electronic
Systems. Intelligent Systems

Digital Communications for the Control of Modular Multilevel Converters

PhD. Thesis Presented by
Tomás Perpetuo Corrêa

2019



Universidad
de Alcalá

PhD. Program in Electronics: Advanced Electronic
Systems. Intelligent Systems

**Digital Communications for the
Control of Modular Multilevel
Converters**

PhD. Thesis Presented by

Tomás Perpetuo Corrêa

Advisors

**Dr. Emilio José Bueno and Dr. Francisco
Javier Rodríguez Sanchez**

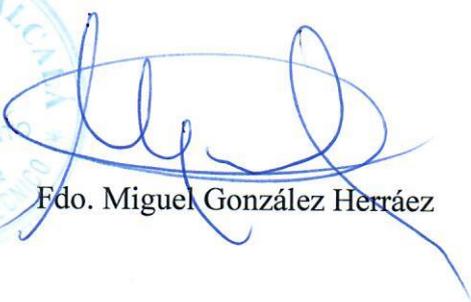
Alcalá de Henares, January 10th, 2019

Dr. Miguel González Herráez, Coordinador de la Comisión Académica del Programa de Doctorado del Departamento de Electrónica de la Universidad de Alcalá,

INFORMA: Que la Tesis Doctoral titulada “**Digital Communications for the Control of Modular Multilevel Converters**” presentada por D. Tomás Perpetuo Corrêa, y realizada bajo la dirección de los doctores D. Francisco Javier Rodríguez Sánchez y D. Emilio José Bueno Peña, cumple con todos los requisitos científicos y metodológicos, para ser defendida ante un Tribunal, según lo indicado por la Comisión Académica del Programa de Doctorado.

Alcalá de Henares, 09 de Enero de 2019




Edo. Miguel González Herráez

Dr. Francisco Javier Rodríguez Sánchez, Catedrático de la Universidad de Alcalá, y
Dr. Emilio José Bueno Peña, Profesor Titular de la Universidad de Alcalá,

INFORMAN: Que la Tesis Doctoral titulada “**Digital Communications for the Control of Modular Multilevel Converters**” presentada por D. Tomás Perpetuo Corrêa, y realizada bajo la dirección de los doctores D. Francisco Javier Rodríguez Sánchez y D. Emilio José Bueno Peña, dentro del área de las comunicaciones digitales de altas prestaciones empleadas dentro los lazos de control de alta velocidad de convertidores electrónicos de potencia multinivel, reúne los méritos de calidad y originalidad para optar al Grado de Doctor.

Alcalá de Henares, 09 de Enero de 2019



Fdo. Francisco Javier Rodríguez Sánchez



Fdo. Emilio José Bueno Peña

À Bruna, companheira de todas as horas,
com todo meu amor, admiração e carinho,

e à Ana, que como um passarinho,
alegra e dá razão às nossas vidas.

Abstract

Invented in 2001, the Modular Multilevel Converter (MMC) marked a leap in the technology for converting and controlling the electrical energy in high voltage levels. The MMC has gained much attention in both academy and industry ever since, and it has been widely studied.

Nevertheless, the number of works that explore the use of digital communications in this application domain is limited. The digital communication employed to implement an internal control network simplifies the converter assemblage and maintenance; it also brings several benefits, such as the ability to adopt new control strategies or to parameterize the cells during operation.

In this work, we investigate several aspects of high-speed digital communications for MMCs. It starts with a review of the operating principles and particular control characteristics of the Modular Multilevel Converter and the state-of-the-art of communication solutions for power electronic converters. We also discuss how the MMC and the internal network interact and influence both the design and operation of each other.

Next, a codesign strategy for the control and communication makes possible to operate Ethernet-based ring networks with quasi-optimum Minimum Cycle Time, allowing the control algorithms to execute at the necessary high rates, what is especially difficult in converter with hundreds of cells.

Following, we investigate the internal delay of Ethernet nodes and propose hardware accelerators for implementation in Field Programmable Array technology that can minimize the latency during the reception of packets.

Finally, a model-based predictor compensates for the loop delay and overcome some of the limitations caused by the introduction of the network. We explain the predictor in mathematical terms, assess the influence of parameter variations, and present simulation and experimental results to demonstrate its effectiveness.

After all, the thesis intends to make relevant contributions to the use of high-speed digital communications in Modular Multilevel Converters with a high number of cells, allowing at the same time all the benefits of such implementation without compromising the high performance of the system control.

Keywords: Modular Multilevel Converter, digital communications, network controlled systems..

Resumen

Inventado en 2001, el 'Modular Multilevel Converter' (MMC) marcó un salto en la tecnología importante en la gestión de redes energéticas de alta tensión. Desde entonces, los convertidores MMCs han ganado mucha atención en la academia y en la industria y han sido ampliamente estudiados.

Sin embargo, el número de trabajos que exploran el uso de las comunicaciones digitales dentro de los lazos de control internos de los MMCs, con periodos de muestreo del orden de los cientos de microsegundos, es muy reducido. La comunicación digital empleada para implementar una red de control interno simplifica el montaje y el mantenimiento del convertidor. También permite adaptar nuevas estrategias de control o parametrizar las celdas que conforman el MMC durante la operación. Todo ello justifica el estudio que se va a realizar en esta Tesis, relacionado con el empleo de comunicaciones de alta velocidad dentro de los lazos internos de control de los MMCs.

En este trabajo, se investigan varios aspectos de las comunicaciones digitales de alta velocidad para MMC. Comienza con una revisión de los principios operativos y las características de control particulares de los MMCs, y del estado de la técnica de las soluciones de comunicación para convertidores electrónicos de potencia de características similares. También se discute la interacción entre el modelo del MMC y los retardos introducidos por la red de comunicaciones interna de alta velocidad, lo que influye tanto en el diseño de los lazos de control como en la elección de dicha red de comunicaciones.

Para reducir el efecto en la operación del convertidor, de forma que este siga verificando los códigos de red desde el punto de vista de tiempos de actuación ante perturbaciones transitorias de la red, en esta Tesis se propone una estrategia de co-diseño para el control y las comunicaciones que permite operar a redes de comunicaciones conectadas en anillo y basadas en Ethernet con Tiempos de Ciclo Mínimos (Minimum Cycle Time), de forma que el convertidor verifique los tiempos de respuesta establecidos, lo cual es especialmente crítico en convertidores con cientos de celdas. La Tesis también estudia el retardo interno introducido por los nodos Ethernet y se proponen aceleradores de hardware implementados sobre Field Programmable Array (FPGA) lo que minimiza la latencia durante la recepción de paquetes.

Para compensar los retardos introducidos por las comunicaciones, y asegurar un comportamiento óptimo del lazo, se propone el empleo de un predictor basado en un modelo

del control de corriente. Se verifica su operación ante variaciones de los parámetros del convertidor, presentando resultados de simulación y experimentales.

En conclusión, la Tesis pretende ser una contribución relevante en la introducción de comunicaciones de alta velocidad en los lazos de control con tiempos de muestreo reducidos de convertidores multinivel de elevado número de celdas, verificando las normativas en cuanto a tiempos de respuesta, pero a la vez optimizando el montaje y mantenimiento de este tipo de convertidores.

Palabras clave: Modular Multilevel Converter, comunicaciones digitales, sistemas controlados por red..

Contents

Abstract	VI
Resumen	VIII
Contents	X
List of Figures	XIV
List of Tables	XIX
List of Acronyms	XXI
1. Introduction	1
1.1. HVDC and FACTS	3
1.2. Multilevel Converters	7
1.2.1. Diode-Clamped Converter	7
1.2.2. Flying Capacitor Converter	9
1.2.3. Cascaded H-Bridge Converter	9
1.2.4. Modular Multilevel Converter	10
1.3. Objectives	11
1.4. Review of Contributions	12
1.5. Thesis Structure	13
1.6. Thesis Context	13
1.7. Published Works	14
2. Overview of Modular Multilevel Converters	15
2.1. Modular Multilevel Converter Model	17
2.2. Control of Terminal Magnitudes	19

2.3. Specific Control Objectives and Methods	21
2.3.1. Modulation	21
2.3.1.1. Short Period Modulation	21
2.3.1.2. Instantaneous Voltage Modulation	25
2.3.1.3. Fundamental Frequency Modulation	26
2.3.1.4. Carrier-based Vs NLC modulation	28
2.3.2. Capacitor Voltage Balancing	28
2.3.2.1. Natural Balancing	29
2.3.2.2. Sorting Algorithm	30
2.3.2.3. Variations of the Sorting Algorithm	31
2.3.2.4. Closed-Loop Balancing	32
2.3.2.5. Open-loop or Partial Open-loop	34
2.3.2.6. Others	35
2.3.3. Circulating Current Control	36
2.3.4. Control Structure	37
2.4. Network Controlled MMCs	37
2.5. Conclusions	39
3. Overview of Communication Networks for MMCs	40
3.1. Requirements	42
3.1.1. Bandwidth and Latency	42
3.1.2. Payload	42
3.1.3. Reliability	42
3.1.4. Transmission Media	43
3.1.5. Network Topology	44
3.1.6. Synchronization Accuracy	46
3.1.6.1. Physical Layer Synchronization	47
3.1.6.2. AC Terminal Voltage Distortion	50
3.2. Existing Solutions	51
3.2.1. Power Electronics System Network	52
3.2.2. EtherCAT	54
3.3. Future Perspectives	56
3.4. Network Induced Latency	59
3.5. Conclusions	60

4. Network and Control Co-design	62
4.1. TTRing	64
4.1.1. Fast Forwarding	67
4.1.2. Minimum Cycle Time	67
4.2. DiSortNet	69
4.2.1. Dual Insertion Sorting	70
4.2.2. Distributed Minimum/Maximum Identification	75
4.2.3. Compact Modulation	76
4.2.4. Control Architecture	78
4.2.5. Minimum Cycle Time	78
4.3. Performance Comparison	80
4.4. Fault Signal	81
4.4.1. Error signal	82
4.4.2. Bit Encoding	82
4.4.3. Magic Number	82
4.4.4. Gigabit Ethernet	83
4.5. Simulation of MMC With Communication Network	83
4.5.1. Simulation Results	85
4.6. Conclusions	86
5. Minimal Reception Delay for Ethernet Interfaces	87
5.1. Media Access Control	88
5.2. Implementation Details and Possibilities	89
5.2.1. Lightweight Internet Protocol	90
5.3. Measurements	91
5.3.1. Discussion	93
5.4. Hardware Accelerators	97
5.5. Conclusions	100
6. Model-Based Compensation for Network Delays	102
6.1. Influence of Sampling Rate and Delay on Control Performance	103
6.2. Proposed Estimation Algorithm	103
6.2.1. Estimation of Circulating Current	106
6.2.2. Modulation and Capacitor Balancing	107

6.3. Closed-Loop Stability	107
6.4. Parameters Sensitivity	109
6.5. Simulation and Experimental Results	111
6.6. Conclusions	115
7. Conclusions and Future Work	116
7.1. Future work	118
Bibliography	120
A. Node Carrier Board	132
A.1. Introduction	132
A.2. Characteristics	133
A.3. Schematics	134

List of Figures

- 1.1. Change in CO2 emissions (Giga Tones), 1990 to 2011. [1] 2
- 1.2. Predictions on Electrical Energy Consumption. [2] 2
- 1.3. Comparison of right-of-way of AC and DC transmission lines. [3] 3
- 1.4. AC and DC transmission system costs. 4
- 1.5. FACTS ideal compensators. 6
- 1.6. FACTS shunt compensators phasor diagram. 6
- 1.7. FACTS series compensators phasor diagram. 6
- 1.8. Multilevel Converters topologies. 8
- 1.9. Modular Multilevel Converter circuit configurations (as classified in [4]). . . 11
- 1.10. Star control architecture. The empty cycle represents the central controller
and the filled ones the cells. 11

- 2.1. Converter hall of a 1000 MW HVDC transmission link between France and
Spain. [5] 16
- 2.2. Cell topologies. 16
- 2.3. Modular Multilevel Converter circuit diagram. 18
- 2.4. Arms and the terminal voltage of a five levels MMC. 19
- 2.5. Abstract control. 20
- 2.6. Power/Voltage and current control loops considered. 20
- 2.7. Space-vector diagram of a 5-level converter. [6] 22
- 2.8. Carrier-based PWM. 22
- 2.9. Multilevel carrier PWM: (a) Phase Disposition, (b) Phase Opposition Dis-
position, (c) Alternative Phase Opposition Disposition, (d) Saw-tooth and
(e) Phase-shifted carriers. 24
- 2.10. Output voltage of NLC with critical and infinite sampling in an MMC with
20 cells per arm. 26
- 2.11. Upper arm command of a MMC with NLC modulation. 26

2.12. Generalized multilevel waveform.	27
2.13. Harmonic content of the arm voltage in a PWM cycle. The first harmonic magnitude is small compared with others ($n=10$).	32
2.14. Control strategy with averaging and balancing terms. [7].	33
2.15. Partly distributed control of MMCs using phase-shifted carrier PWM. [8].	35
2.16. Block diagram of the control.	38
2.17. Double-star MMC circuit diagram indicating the measured magnitudes.	38
3.1. Use of Fiber Optics (orange lines).	43
3.2. Network topologies. The hollow cycle represents the master node and the filled ones the slaves.	44
3.3. Minimum Cycle Time for the ring (EtherCAT) and tree (Fast Ethernet) network topologies, when broadcasting information from the master controller to cells, only.	45
3.4. Cells and communication link.	46
3.5. Variable delay due to Clock Domain Crossing.	48
3.6. Physical Layers synchronization. The continuous lines represent one direction of the ring network and the dashed lines the other one.	49
3.7. Histogram of Υ for the first harmonic and different σ 's.	51
3.8. PESnet data frames.	52
3.9. PWM generation using a time-division of the period.	53
3.10. PESNet Minimum Cycle Time for various numbers of empty words (P_{null}).	53
3.11. EtherCAT Minimum Cycle Time with a different number of bytes per node.	55
3.12. Ethernet frames traveling in a network. The green and red lines indicate the start and end of a frame, respectively.	59
3.13. The network introduces additional delay to the actuation. As the sampling period reduces from (a) to (b), the loop delay in number of samples increases. \mathbf{x}_k is the state vector in instant k , and $\mathbf{u}_{k k-n}$ is the plant input vector in instant k calculated in $k - n$	60
4.1. The protocols proposed are for networks with (a) ring or (b) hybrid ring/star topology.	63
4.2. MMC controlled through a ring network. The small boxes represent the cells of the three phases.	63
4.3. Representation of TTRing phases.	65

4.4. Two periods of the TTRing network with 43 nodes and modeled using OMNeT++.	66
4.5. Block diagram of the control when using TTRing.	68
4.6. Details of the slave implementation. It uses the receiver clock <i>Rxc</i> as the transmitter clock.	69
4.7. Dual Insertion Sorting. It reorders the list by moving the element with maximum value to the top of the list and with the minimum value to the bottom. The remaining elements are shifted right or left, to accommodate the new max/min, respectively.	71
4.8. Dual Insertion Sorting in a converter with five cells per arm.	72
4.9. Dual Insertion Sorting in a converter with 20 cells per arm.	73
4.10. Simulation of the capacitor voltage, phase A, upper arm, 20 cells per arm. Capacitor rated voltage of 1500 V.	74
4.11. Simulation of the capacitor voltage, phase A, upper arm, 200 cells per arm. Capacitor rated voltage of 2000 V.	74
4.12. DisortNet protocol frames.	76
4.13. Capacitor voltage representation with a reduced number of bits without loss of accuracy.	76
4.14. Experimental results of pulse commands over the network using the Compact Modulation Strategy.	77
4.15. Block diagram of the control when using DiSortNet.	79
4.16. Minimal critical frequency and the inverse of the DiSortNet MCT. The necessity to run the Min/Max identification faster than the Minimum Critical Frequency limits the protocol coverage to networks with less than 110 nodes approximately.	80
4.17. Minimum Cycle Time depending on the network size, when the payload is 48 bytes (TTRing), 4 bytes/node (EtherCAT), or following (4.6) (DiSortNet).	81
4.18. Use of a triggered sub-system to include the network behavior.	85
4.19. Central controller sends a reference to the nodes (dashed line), but due to failures in the network, deviations occur (solid lines). We modeled the network with two loss probabilities: (a) 0.0001 and (b) 0.005.	85
5.1. AXI Read transaction from the Ethernet Lite MAC to the processor main memory. When the memory range is configured as Device Memory, the AXI Master reads data four times faster (the <i>RVALID</i> signal indicates a read transaction).	90

5.2. Outgoing packet sending. The dashed lines with arrow indicate the measurement points.	92
5.3. Incoming packet processing. The dashed lines with arrow indicate the measurement points.	92
5.4. Incoming packet: delay to enter ISR after receiving packet	94
5.5. Incoming packet: delay to transfer data to PBUF after receiving packet . .	94
5.6. Incoming packet: delay to enter UDP callback after receiving packet	95
5.7. Outgoing packet: delay to start sending packet after UDP command	95
5.8. Outgoing packet: delay to send packet after starting copying from PBUFs .	96
5.9. Outgoing packet: delay to send packet after triggering the MAC hardware	96
5.10. Fast-track for incoming UDP port 1026 packets.	98
5.11. Ethernet Direct Copy hardware accelerator block diagram.	98
5.12. Capture showing EDC internal signals while receiving a packet with 64 Bytes. The x -axis is in samples, and the sampling period equal to 10 ns. .	99
5.13. Delay to enter UDP callback after receiving packet using Ethernet Direct Copy accelerator. The bar fillings are transparent to show that the measurements are overlapping.	100
6.1. Open-loop Bode diagram of the plant with controller.	104
6.2. Configuration of the proposed control with long network delays.	105
6.3. Block diagram of a closed-loop system with two loop delays, a PI controller, and the model-based predictor.	106
6.4. Mismatch between u_{circ} and its reference. As a consequence, an estimation of i_{circ} based on the references delivers poor results.	107
6.5. Maximum deviation to the average capacitor when the sorting uses measurements with and without delays. Capacitor rated voltage of 170V and sampling period of 100 μ s.	108
6.6. Stability analysis and influence of model error in the system closed-loop poles and zeros.	110
6.7. Modular Multilevel Converter prototype with five cells/arm.	112
6.8. Block diagram of the control when using DiSortNet.	113
6.9. Experimental and simulation results for a system with delay of two samples.114	
6.10. Experimental results for a system with a delay of three samples and the model-based prediction.	114

6.11. Comparison of the arm capacitor voltages when the modulation uses a sorted list based on measurements without delay (dashed blue line) and with a delay of two samples (continuous orange line). 114

A.1. Node Carrier board designed for the emulation of the communication network of an MMC. 133

A.2. Cover sheet 134

A.3. Sheet 2 135

A.4. Sheet 3 136

A.5. Sheet 4 137

A.6. Sheet 5 138

A.7. Sheet 6 139

A.8. Sheet 7 140

A.9. Sheet 8 141

A.10.Sheet 9 142

A.11.Sheet 10 143

A.12.Sheet 11 144

A.13.Sheet 12 145

List of Tables

- 1.1. Five-levels NPC converter and its switching states. 8
- 1.2. Five-levels Flying Capacitor converter and its switching states. 9
- 1.3. Cascaded H-bridge cell switching states. 10

- 4.1. Latency of PHYs operating with 100BASE-T RGMII 67
- 4.2. 4b/5b Bit Encoding. 82

- 5.1. Incoming and outgoing packet mean delay and deviation, in μ s. 97

- 6.1. Simulation and prototype parameters. 111

List of Acronyms

APOD	Alternate Phase Disposition.
CDC	Clock Domain Crossing.
CHB	Cascaded H-Bridge.
DMA	Direct Memory Access.
EDC	Ethernet Direct Copy.
ESC	EtherCAT Slave Controller.
EtherCAT	Ethernet for Control Automation Technology.
FACTS	Flexible Alternate Current Transmission Systems.
FCS	Frame Check Sequence.
FIFO	First In First Out.
FPGA	Field Programmable Gate Array.
GEM	Gigabit Ethernet MAC.
GHG	greenhouse gases.
GMI	Gigabit MII.
GOF	Glass Optical Fiber.
HVDC	High-Voltage Direct Current.
IP	Internet Protocol.
ISR	Interrupt Service Routine.
lwip	Light-Weight Internet Protocol.

MAC	Media Access Control.
MCT	Minimum Cycle Time.
MII	Media Independent Interface.
MMC	Modular Multilevel Converter.
NLC	Nearest Level Control.
NPC	Neutral-Point-Clamped.
PD	Phase Disposition.
PDI	Process Data Interface.
PESnet	Power Electronics System Network.
PHY	Physical Layer.
PI	Proportional-Integral.
POD	Phase Opposition Disposition.
POF	Plastic Optical Fiber.
PR	Proportional-Resonant.
PSC	Phase-Shifted Carrier.
PWM	Pulse Width Modulation.
RAM	Random Access Memory.
RGMII	Reduced Gigabit MII.
SHE	Selective Harmonic Elimination.
SoC	System-on-Chip.
STATCOM	Static Synchronous Compensator.
TCP	Transmission Control Protocol.
THD	Total Harmonic Distortion.
UDP	User Data Protocol.
VSC	Voltage-Source Converter.

Chapter 1

Introduction

The way energy is consumed and produced has faced considerable changes in the last years and will continue to change in the years to come. The driver of such change is the verification of global warmth and an almost consensual understanding that its caused by the emission of greenhouse gases (GHG). In 1997, an international agreement was reached in the city of Kyoto, Japan, targeting to reduce in 5.2% the GHG emission by 2012 as compared to 1990 levels [9]. This treaty, known as the Kyoto Protocol, became into force only in 2005, after the ratification process in all the countries. By that point, global emissions had risen substantially [10], as several nations, China in particular, that had no reduction target at all, produced more GHG than the reductions accomplished by other countries in the period (Fig. 1.1).

After a failed attempt to extend the Kyoto Protocol to 2020, the states have closed an agreement in Paris to continue combating climate changes. As of July 2018, 195 parties have signed it, and 180 have become part of it. The Paris Agreement pledges to keep “global temperature rise this century well below 2 degrees Celsius above pre-industrial levels” [11]. Humankind, to be successful, has to limit the amount of GHG in the atmosphere, but two-thirds of the budget has already been emitted [12]. In other words, the emission of GHG has to be cut dramatically in the following years.

According to Williams et al. [13], if the target adopted in some countries of reducing in 80% the GHG 1990 emission levels are to be met by 2050, three energy system transformations are necessary: improvement of energy efficiency; decarbonization of the electricity supply; and electrification of most existing direct fuel uses. Other authors [14, 15] seem to agree that increased use of electricity is imperative for fighting climate changes.

Frisch et al. [2] present results of studies that made projections for the future of electricity consumption (the Total Electricity Demand) to the year 2050 in the United States. These studies use different models and scenarios, but all “share certain attributes: they assume rapid electrification of end uses, rely on a balanced portfolio of low- and zero-carbon electric generation technologies, achieve high-decarbonization goals.” [2] They produced a wide range of projected growth (Fig. 1.2), but all agree that electricity consumption

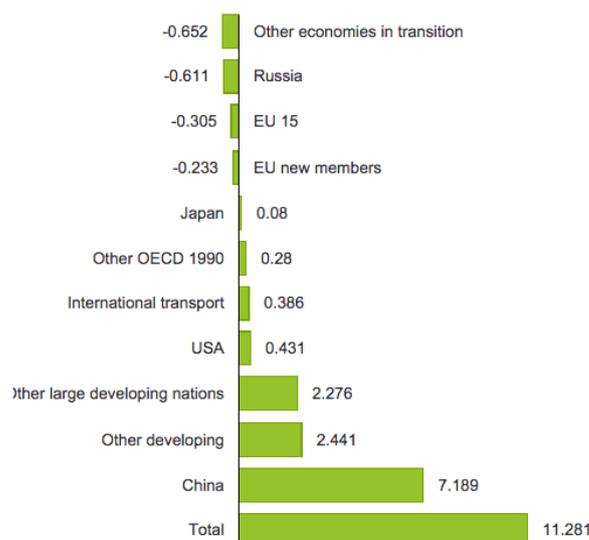


Figure 1.1: Change in CO2 emissions (Giga Tones), 1990 to 2011. [1]

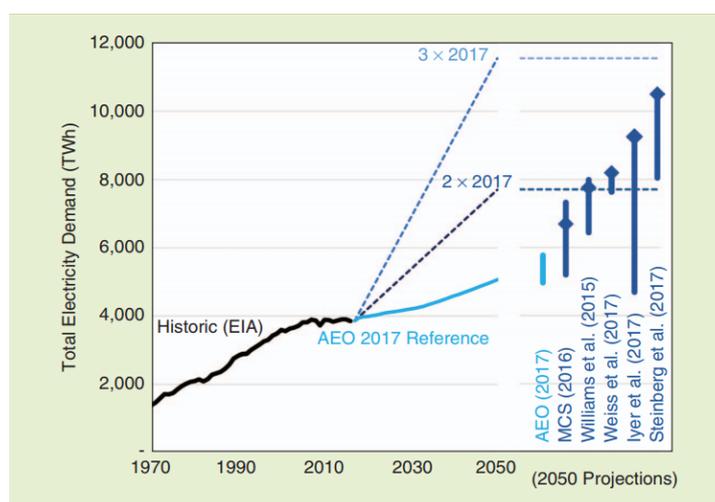


Figure 1.2: Predictions on Electrical Energy Consumption. [2]

will have a considerable increase, from 1.5 times to 3 times the Total Electricity Demand of 2017.

The challenge is to increase the electricity production and decarbonize the present generation plants in a short period of 30 years. Historically, Kramer and Haigh [16] argue that the development and “materialization” of existing energy technologies take about this long. If compared to consumer electronics, this development cycle seems like an eternity, but few differences explain it: the energy systems have considerable dimension; the reliability requirements are tougher; and the “consumers,” i.e., utility companies, are conservative when adopting “new” technologies. As a conclusion, we could say that it is unlikely that entirely new technologies will play a vital role in the transition to a low-carbon world.

Under those circumstances, the generation, transmission, and distributions of electric

energy must evolve to cope efficiently with this increasing demand. High-Voltage Direct Current (HVDC) and Flexible Alternate Current Transmission Systems (FACTS) are two relevant alternatives for higher transmission capacity of actual and future lines, thus being an example of how utility applications of medium and high voltage conversion are important to reach a low-carbon energy matrix.

1.1. HVDC and FACTS

High Voltage Direct Current technology uses Direct Current to transmit energy. It has been recognized as a competitive solution to deliver a large amount of energy over long distances, to cross long distances under the sea, and to make connections between asynchronous systems [17]. As the Direct Current has no oscillating component, it can transmit continuous power with only two conductors, with the advantage of “smaller right-of-way, simpler and cheaper towers, and reduced conductor and insulator costs.” [3] As an example, Fig. 1.3 shows a comparison of AC and DC transmission towers for a system carrying 2000 MW. Further benefits of HVDC transmission are the reduction of dielectric losses, the absence of skin-effect [3] and voltage changes caused by the line reactances.

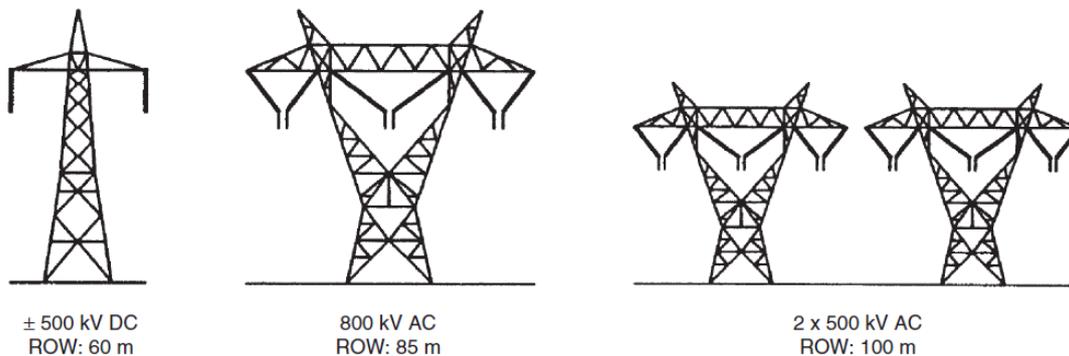


Figure 1.3: Comparison of right-of-way of AC and DC transmission lines. [3]

HVDC systems need converters on both ends to interface with the AC side and passive elements to filter the current harmonic components. In short lines, the converter and filter costs exceed the savings mentioned earlier, hence the HVDC system is more expensive than an AC line. As the distance increases, the savings grow, till a so-called break-even distance, when HVDC and AC total costs match. This distance is typically between 500 km and 800 km for overhead transmission lines and between 40 km and 80 km for underground or undersea cables [18]. Fig. 1.4 illustrates the evolution of the AC and DC transmission costs with the line length.

Two types of converter dominate the modern HVDC systems. The first type is the Line-Commutated Current Source Converters, whose basic building block is a three-phase, six-pulses, Thyristors bridge. In this topology, the commutation of the current between

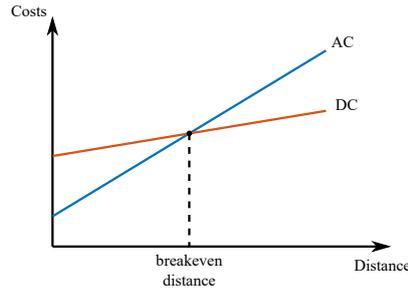


Figure 1.4: AC and DC transmission system costs.

phases needs a strong voltage source, and the AC current must lag the voltage; therefore the converter consumes reactive power (typically about 50-60% of active power [3]). Local installed capacitor banks provide the necessary reactive power, but the AC system must accommodate any lack or surplus [17].

The second type is the Voltage-Source Converter (VSC) based in self-commutated switches. The main advantage of VSC-HVDC systems is the full controllability of the converter, as it operates in all four-quadrant and can impose independent levels of active and reactive power. Other advantages are higher tolerance to disturbances in the AC network; possibility to connect the VSC-HVDC system to a “weak” AC network [19, 20]; faster dynamic response; and reduced filtering effort due to the higher commutation frequency of the switches.

The first VSC-HVDC systems employed two-level topology and had to connect several power switches in series to reach the total blocking voltage needed [19]. This connection brings several difficulties concerning dynamic voltage share between the individual components that are caused by timing differences of the driver circuits as well as physical differences in the structure of the devices [21].

Today, the Modular Multilevel Converter (MMC) is the dominant technology, with major companies like Siemens, ABB, and Alstom (GE Energy) having a commercial solution. The MMC avoids the necessity of switches serialization by using several identical cells (or sub-modules) in series.

The FACTS are AC lines that use series or shunt compensators (Fig. 1.5) to achieve a high level of control and flexibility [20]. To understand the concept behind the FACTS, consider the two generators connected with a transmission line, modeled as an inductive reactance X_L , as depicted in Fig. 1.5a. Consider also that the two generators have the same voltage magnitude ($|U_s| = |U_r|$) and are phase-shifted by δ degrees. Without the third voltage source at the middle of the line, the power transmitted from the source generator to the receptor is expressed by (1.1).

$$P_s = \frac{U_s^2}{2 \cdot X_L} \sin(\delta). \quad (1.1)$$

If the third voltage source is variable and set to have the same magnitude as the other two and its phase is exactly between them, i.e., $\delta/2$ lagging U_s and $\delta/2$ leading U_r , the circuit phase diagram will be that of Fig. 1.6a. In this case, the voltage source acts as a reactive shunt compensator, because the resulting current flowing through is orthogonal to V_m . The power transmitted between S and R is then equal to (1.2).

$$P_s = \frac{U_s^2}{X_L} \sin(\delta/2). \quad (1.2)$$

As P_s in (1.2) is larger than in (1.1) for any δ [20], the shunt compensator increases the transmission line capacity. If the compensator has source or storage capability, it can draw active power, the U_s angle can be other than $\delta/2$, and the currents i_s and i_r may have different magnitudes (Fig. 1.6b) .

Consider a second situation, when the controlled voltage source is inserted in series with the transmission line (Fig. 1.5b). In this case, the current flowing in the line is lagging the line voltage drop, (1.3) and (1.4).

$$I_s = U_{X_L}/2jX_L, \quad (1.3)$$

$$U_{X_L} = (U_s - U_r - U_c). \quad (1.4)$$

When the controlled source has a voltage 90° out-of-phase with the current, it only produces or consumes reactive power. If the displacement factor is capacitive, the voltage lags the current, U_{X_L} increases, and the current flowing between the two generators also increases (Fig. 1.7a). On the other hand, if it is inductive, the voltage leads the current, U_{X_L} reduces, and the transmission line current also reduces (Fig. 1.7b). Therefore, the series compensator can influence the energy exchange between the two generators without requiring active power.

The FACTS compensators use power electronics technology and can be classified as variable impedance or voltage-source converter types [22]. Examples of the first type are static VAR compensator, Static Synchronous Series Compensator, Thyristor Controlled Series Capacitor or compensator, and Thyristor Controlled Phase Shifting Transformer. Examples of the second type are Static Synchronous Compensator (STATCOM), Static Synchronous Series Compensator, Interline Power Flow Controller, and the Unified Power Flow Controller.

As in the HVDC, the use of VSC has several advantages over the variable impedance

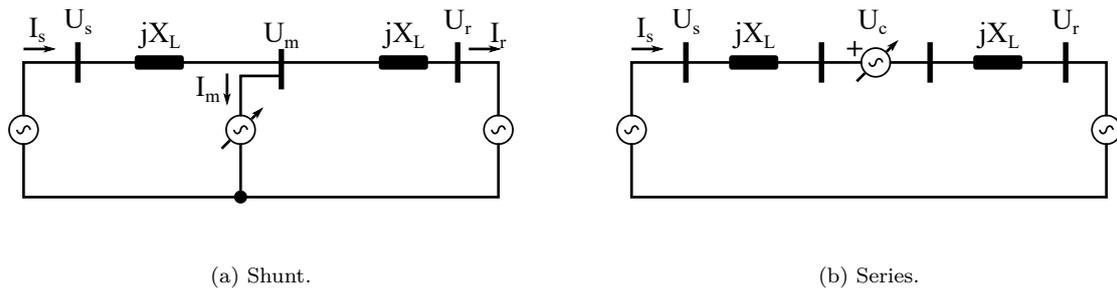


Figure 1.5: FACTS ideal compensators.

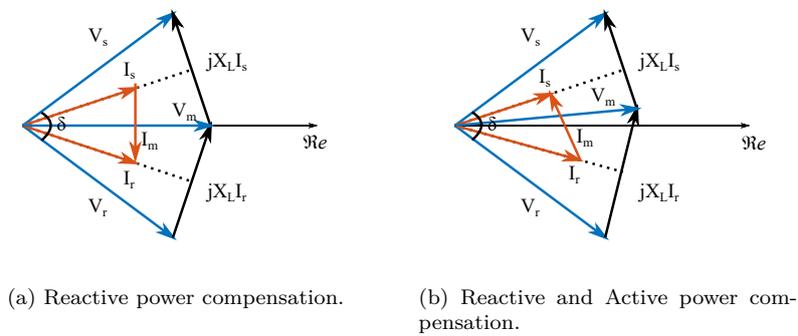


Figure 1.6: FACTS shunt compensators phasor diagram.

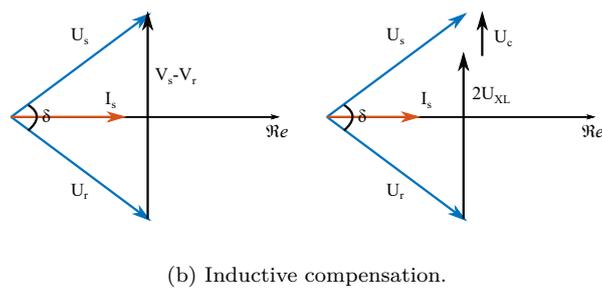
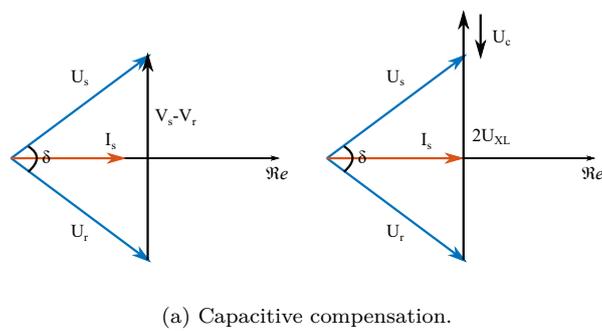


Figure 1.7: FACTS series compensators phasor diagram.

type [22]: they are more compact, operate even under low bus voltage, have overload capability, and can supply active power if associated with an energy storage system. The Multilevel Converters, introduced in the next section, were responsible for the advances in VSC solutions for both HVDC and FACTS applications.

1.2. Multilevel Converters

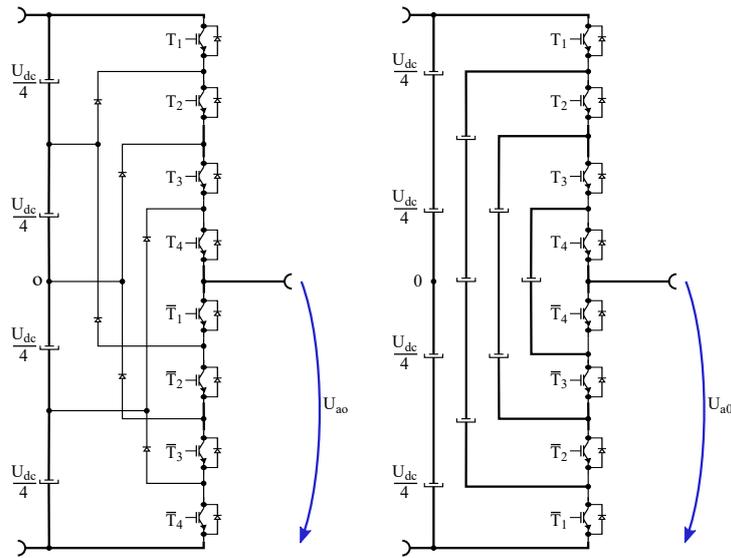
While the deployment of low voltage converters is widespread, the same does not hold for higher voltage levels. The extensive adoption of power electronics in high power, high and extra high voltage applications is recent [23] and was made possible by the use of multilevel converters. The most popular topologies of multilevel converters are: the diode-clamped converter (Fig. 1.8a), proposed in 1981 [24], the flying capacitor converter (Fig. 1.8b), proposed in 1992 [25], the cascaded H-bridge converter (Fig. 1.8c), proposed in 1988 [26], and the Modular Multilevel Converter (Fig. 1.8d), patented in 2001 [27]).

In general, multilevel converters have increased power processing capability due to the higher output voltage for a given power switch blocking rate, and better output signal quality, because the higher number of levels reduces the voltage harmonic content when compared to conventional two-level alternatives [28].

1.2.1. Diode-Clamped Converter

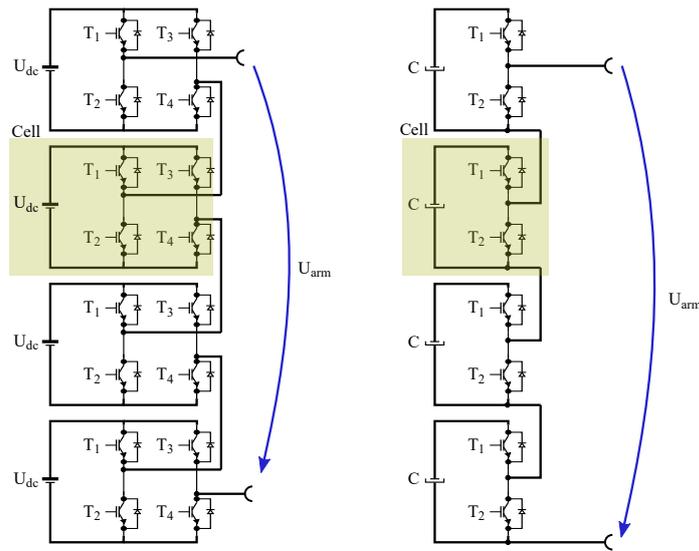
Nabae et al. [24] proposed the three-level diode-clamped converter in 1981. In the 1990s, several researchers reported converters with four, five and six levels [29]. This topology, also known as Neutral-Point-Clamped (NPC), is the most widely used multilevel converter for high power applications [30], such as medium voltage drives, static VAR compensation and interconnects [29]. Fig. 1.8a shows a leg of the diode-clamped converter with five levels. Table 1.1 resumes the five possible switching states, and the applied phase voltage referred to the negative DC rail. Note that always five adjacent switches are ON in a given phase.

The diode-clamped converter has an uneven power loss distribution, and the diodes have to withstand higher voltages than the switches. If the diode employed is rated to the same voltage of the power switches, the number of diodes per phase is $(m - 1) \cdot (m - 2)$ for an m -level converter [31]. The quadratic number of diodes per level and their connections between power switches make the construction of the converter more complex as the number of levels increase. As a consequence, diode-clamped converters with more than five levels are rare [30].



(a) Diode-clamped phase.

(b) Flying Capacitor phase.



(c) Cascaded H-Bridge arm.

(d) Modular Multilevel Converter arm.

Figure 1.8: Multilevel Converters topologies.

Table 1.1: Five-levels NPC converter and its switching states.

Voltage U_{a0}	Switching State							
	T_1	T_2	T_3	T_4	\bar{T}_1	\bar{T}_2	\bar{T}_3	\bar{T}_4
$U_4 = 1/2 U_{dc}$	1	1	1	1	0	0	0	0
$U_3 = 1/4 U_{dc}$	0	1	1	1	1	0	0	0
$U_2 = 0V$	0	0	1	1	1	1	0	0
$U_1 = -1/4 U_{dc}$	0	0	0	1	1	1	1	0
$U_0 = -1/2 U_{dc}$	0	0	0	0	1	1	1	1

1.2.2. Flying Capacitor Converter

The Flying Capacitor topology, proposed by Meynard and Foch in 1992 [25], has a configuration similar to the diode-clamped converter, but it uses capacitors in a ladder structure to limit the voltage blocked by each power switch. Fig. 1.8b shows the phase-leg of a five-levels flying capacitor converter. Unlike the diode-clamped type, this converter has more freedom in the choice of which power switches are on. For this reason, it has redundant switching states (see table 1.2), i.e., the configuration of power switches conducting and blocking that yields the same phase voltage, that is used to charge and discharge the “flying” capacitors to keep their operating voltage close to the reference.

However, an m -level converter requires $(m - 1)$ DC bus capacitors plus $(m - 1) \cdot (m - 2) / 2$ auxiliary capacitors per phase, considering that the capacitor rating is the same of the power switches [31]. As in the diode-clamped case, the construction of converters with several levels is complex, as well as the control to generate the desired output voltage and keep the auxiliary capacitors charged.

Table 1.2: Five-levels Flying Capacitor converter and its switching states.

Voltage U_{a0}	Switching State							
	T_1	T_2	T_3	T_4	$\overline{T_1}$	$\overline{T_2}$	$\overline{T_3}$	$\overline{T_4}$
$U_4 = 1/2U_{dc}$	1	1	1	1	0	0	0	0
$U_3 = 1/4U_{dc}$	1	1	1	0	0	0	0	1
$U_2 = 0V$	0	1	1	1	1	0	0	0
	1	0	1	1	0	1	0	0
	1	1	0	0	0	0	1	1
	0	0	1	1	1	1	0	0
	1	0	1	0	0	1	0	1
	1	0	0	1	0	1	1	0
$U_1 = -1/4U_{dc}$	0	1	0	1	1	0	1	0
	0	1	1	0	1	0	0	1
	1	0	0	0	0	1	1	1
$U_0 = -1/2U_{dc}$	0	0	0	1	1	1	1	0
	0	0	1	0	1	1	0	1
	0	0	0	0	1	1	1	1

1.2.3. Cascaded H-Bridge Converter

The Cascaded H-Bridge (CHB) converter consists of single-phase full-bridges connected in series, as Fig. 1.8c illustrates. Each full-bridge has its own DC source and can generate three voltage levels (see Table 1.3). The voltage in one full-bridge is independent of the others; thus a CHB converter with n full-bridges generates a phase voltage that has $2n + 1$ levels. In comparison with the diode-clamped and flying-capacitor converters, it needs fewer power devices for the same number of levels [32].

The CHB converter has a modular structure and avoids clamping diodes or auxiliary capacitors. Both characteristics make simpler the construction of converters with a higher number of levels. In contrast, if it transfers active power, each full-bridge needs an

Table 1.3: Cascaded H-bridge cell switching states.

Voltage U_{cell}	Switching State			
	T_1	T_2	T_3	T_4
$U_2 = U_{dc}$	1	0	0	1
$U_1 = 0V$	1	0	1	0
	0	1	0	1
$U_0 = -U_{dc}$	0	1	1	0

independent energy source that often is implemented using a multi-winding transformer. The transformer complexity increases with the number of full-bridges, what puts a ceiling in the number of levels of this type of converter. On the other hand, the CHB converter is well suited to integrate renewable energy sources or battery-based applications into AC systems [29].

1.2.4. Modular Multilevel Converter

Since its introduction in 2001 [27], the Modular Multilevel Converter has drawn much interest from both industry and academia. Its new topology allowed the use of controllable power switches in voltage levels not yet possible and represented a technology leap in High-Voltage Direct Current transmission. Since chapter 2 presents the state-of-the-art of Modular Multilevel Converters, we will shortly introduce them in this subsection.

The Modular Multilevel Converter consists of several *cells* (or sub-modules) that are stacked together to form the converter arms. They can be arranged in different ways, e.g. single-star (Fig. 1.9a), single-delta (Fig. 1.9b), and double-star (Fig. 1.9c) [4].

Fig. 1.8d illustrates an arm of the MMC, where the boxes represent the cells. The cells have several different configurations depending on the number of power switches, diodes, and capacitors, and how they are connected [33]. They can apply different voltages to its output terminals by connecting the internal capacitors to them, leading to a positive (or negative) voltage, or making a short-circuit, when the output voltage is zero.

The exact number of cells depends on the switches blocking voltage (between 1.7 kV for IGBTs and 6.5 kV for IGTCs or IGBTs [34]) and the grid voltage (from few kilovolts to hundreds of kilovolts), but it can exceed 1200 cells in a three-phase converter [35].

The MMC has several advantages: the modularity and scalability allow to reach any required voltage level [6, 36]; the manufacturing of the cells has economies of scale; easy redundancy improves reliability [6]; the high number of levels produces low distorted voltages; it has high efficiency [36]; use of standard components; and avoidance of power switches serialization.

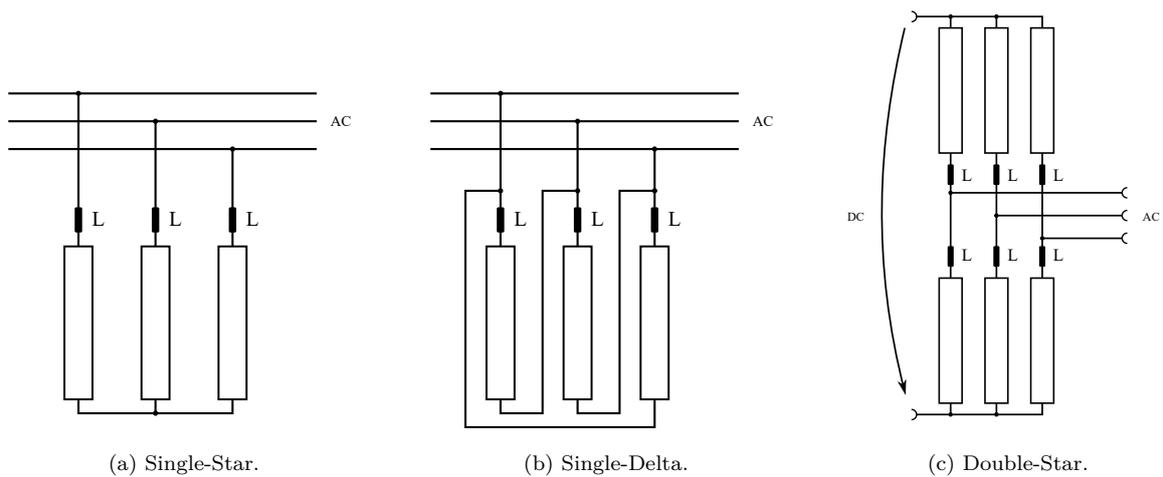


Figure 1.9: Modular Multilevel Converter circuit configurations (as classified in [4]).

1.3. Objectives

The [MMC](#) marked a leap in the technology for converting and controlling the electrical energy in high voltage levels. It employs standard medium voltage power switches in extra and high voltage levels without the intricate switching of serialized devices.

However, the wide range in the number of cells is a challenge for the control hardware of Modular Multilevel Converters. Moreover, as the voltage ratings increase, the control architecture termed *star* (Fig. 1.10), *i.e.*, single connections between the central controller and the cells, leads to an overwhelming amount of interfaces and cables that quickly become impracticable in industrial converters or at least unreliable and cumbersome to assemble and maintain.

The introduction of an internal control network based in digital communications technology can overcome those issues, but also brings challenges to the design, as we will discuss in this dissertation. Even though the [MMC](#) has been a favorite topic of research since its invention, only a few works deal with the use of digital communications in the control of Modular Multilevel Converters.

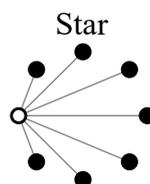


Figure 1.10: Star control architecture. The empty circle represents the central controller and the filled ones the cells.

In the technical literature, two approaches towards implementing an internal network for an [MMC](#) dominate: the user approach [37–45], where the designer looks into the

available industrial solutions and tries to select the one that best fits the purpose; or the one-design-to-fit-all approach [46–49], where the designer seeks a communication solution appropriate for any modular converter.

In this work, we show that both fail to solve the problem since they fall short of performance as the number of cells in an MMC increases. For this reason, we adopt a co-design approach, in which we explore the control and network details to find where one influences the other, seeking compromises that optimize the end solution.

This strategy casts light into several implementation details and helps the interested community to understand where are the limitations, what are the possibilities and the critical aspects to consider when using digital communication in power electronics converters. The price to pay is forgo the benefits of using off the shelf communication protocols and modules that have been used, refined, and validated over the years (the major advantage of the user approach).

To assess the performance of the techniques proposed in this text, we consider the application of MMCs as a STATCOM, both in a simulation model developed in MATLAB/Simulink and a reduced scale prototype.

1.4. Review of Contributions

The main contributions of this work are:

- A time-triggered communication protocol for partially decentralized control of MMCs;
- A distributed sorting strategy associated with summation frame communication;
- A protocol able to reduce the amount of data sent to the slaves and still being flexible to implement several modulation strategies;
- A simplified co-simulation that models the network, the control algorithms, and the power circuit using MATLAB/Simulink and OMNeT++;
- A model-based predictor to compensate for the increased loop delay caused by the network;
- The demonstration that the sorting algorithm has acceptable performance even using delayed capacitor voltage measurements;
- Two new hardware accelerators that reduce the latency inside a node between a packet arrival and the entrance of the software callback function;
- A new board that emulates an MMC control hardware and which is suited to hardware-in-the-loop simulation.

1.5. Thesis Structure

This thesis structure is as follows:

- Chapter 2 reviews the operating principles of the MMC, its control objectives, and the control schemes proposed in the literature.
- Chapter 3 discusses which requirements the MMC application imposes on the communication network. It also reviews the state-of-the-art protocols for power electronics converters and looks into new technologies for improving performance. Moreover, it discusses how the network delay influences the control loop delay and what is the relationship between Minimum Cycle Time, sampling period and the sampling-to-actuation delay.
- Chapter 4 brings two proposals of protocols tailored to two classes of control schemes. It also discusses the importance of a trustworthy simulation environment and proposes a co-simulation method based on OMNeT++ to model the network and MATLAB/Simulink to model the control and power circuits.
- Chapter 5 investigates the internal delays of a node and proposes hardware accelerators for minimizing them. It explains the accelerators concept and implementation, presents measurements using standard Media Access Control, and shows the solution effectiveness with experimental results.
- Chapter 6 proposes a model-based predictor to compensate for the loop delay and overcome its limitations. It shows the mathematical description of the model-based predictor, assesses the parameter sensitivity under several controller gains, and explains how to adapt modulation and capacitor balancing in a network controlled MMC with longer loop delays.
- Chapter 7 summarizes the conclusions of this work and discusses possibilities of future works.

1.6. Thesis Context

This Thesis has been carried out in the context of the following projects and funding programs:

- PRICAM project (S2013-ICE-2933), funded by the Regional Government of Madrid (Spain).
- CONPOSITE project (ENE2014-57760-C2-2-R), funded by the Spanish Ministry of Economy and Competitiveness.
- “Science without borders” program, funded by the Conselho Nacional de Desenvolvimento Científico e Tecnológico, Brazil, process number 233411/2014-3.

1.7. Published Works

During the development of this work, we published the following papers, presented in chronological order:

- T. P. Corrêa, L. Almeida, and E. B. Peña, “Minimal Reception Delay for Ethernet Interfaces,” submitted to the *IET Electronics Letters*, in 29/11/2018.
- T. P. Corrêa, F. Francisco J. Rodríguez, and E. J. Bueno, “Model-based latency compensation for network controlled modular multilevel converters,” *Electronics*, vol. 8, no. 22, pp. 1–16, 2018.
- T. P. Corrêa, L. Almeida, and E. B. Peña, “Hardware/Software Implementation Factors Influencing Ethernet Latency,” in *IEEE Int. Conf. Ind. Informatics (INDIN)*, 2018.
- T. P. Corrêa, L. Almeida, and F. J. Rodriguez, “Communication aspects in the distributed control architecture of a modular multilevel converter,” in *IEEE Int. Conf. Ind. Technology (ICIT)*, 2018.
- T. P. Corrêa., E. J. Bueno, and F. J. Rodriguez, “Communication network latency compensation in a modular multilevel converter,” in *IEEE Energy Conversion Congress and Exposition (ECCE)*. IEEE, 2017.
- T. P. Corrêa and L. Almeida, “Ultra short cycle protocol for partly decentralized control applications,” in *IEEE Int. Conf. Emerging Technologies & Factory Automation (ETFA)*, 2017.
- T. P. Corrêa, O. König, and R. Greul, “Multisampling in interleaved converters and modular multilevel converters,” in *Ann. Conf. IEEE Industrial Electronics Society (IECON)*, 2016.

Chapter 2

Overview of Modular Multilevel Converters

The Modular Multilevel Converter (MMC) is a key enabling technology for High-Voltage Direct Current (HVDC) and Flexible Alternate Current Transmission Systems (FACTS). It was invented in 2001 [27], but only in 2010 Siemens commissioned the first major project using an MMC, the Trans Bay Cable link. This is an 85 km cable running under the San Francisco Bay, in the United States, able to transmit up to 400 MW, at +/- 200 kV [50]. Since then, several other projects and a few other companies (namely ABB and GE-Alstom) have adopted the MMC technology.

The reduced number of manufacturers is easy to explain. Though the MMC HVDC station footprint is between a third or a quarter of competing technologies [50], the converter has considerable dimensions, as Fig. 2.1 demonstrates, and its development requires large amounts of capital. A video [51] with some details of the DolWin3 project shows how off-shore MMCs are an impressive work of engineering.

Modular Multilevel Converters are built by series connections of identical modules (or cells). The number of cells in a converter depends on the nominal terminal voltage and the reliability requirements, with industrial MMCs reported in the literature in the medium (1 kV–35 kV), high (35 kV–230 kV), and extra high (higher than 230 kV) voltage levels [23,52]. As the typical cell blocking voltage is between 1.7 kV (IGBTs) and 6.5 kV (IGCTs or IGBTs) [34], the possible number of cells per arm can go from just a few to hundreds.

Researchers proposed several topologies for the cells (see [33] for a review), but two are the most relevant to industrial converters: the half-bridge and the full-bridge (Fig. 2.2).

The half-bridge cell (Fig. 2.2a) has two power switches and can produce two voltage levels at its terminals: U_{cell} , when T_1 is on, and the capacitor is connected to the output; or zero volts, when T_2 is on, and the terminals are short-circuited. In both states, current flows only through one power switch at a time, what minimizes conduction losses. A converter built only using half-bridges has problems to cope with DC short-circuits,



Figure 2.1: Converter hall of a 1000 MW HVDC transmission link between France and Spain. [5]

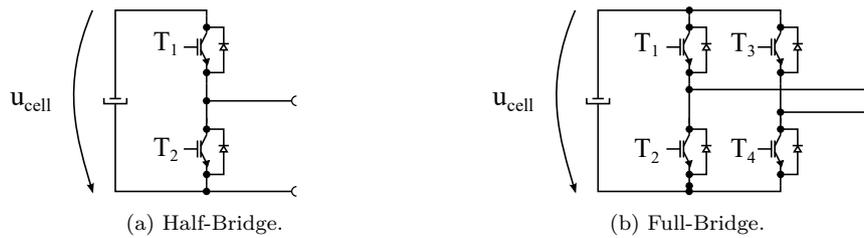


Figure 2.2: Cell topologies.

because it is unable to generate negative voltages to keep the antiparallel diodes blocking, so current flows through them when the DC voltage collapses [53].

The full-bridge cell (Fig. 2.2b) has four power switches and can produce three voltage levels at its terminals: U_{cell} , when T_1 and T_4 are on; $-U_{cell}$, when T_2 and T_3 are on; and zero volts, when either T_1 and T_3 or T_2 and T_4 are on. In all states, two power switches conduct simultaneously, thus leading to higher losses. Unlike the half-bridge, converters using this cell can control the AC and DC currents under DC faults.

In this work, we will consider converters assembled only with half-bridges, as this cell topology has minimum power losses and high efficiency is a central requirement of MMCs.

In this chapter, we explain how the MMC works, present its model, and review its specific control objectives and the methods proposed in the technical literature to date. The primary purpose of this review is to identify the requirements that the MMC operation and control impose to the internal communication network, so when reviewing networking technologies, we can match control and communication strategies (discussed in the next chapter) to improve the overall performance.

2.1. Modular Multilevel Converter Model

The MMC, in its three-phase double-star configuration¹, has six *arms*, each one built with the association of n cells in series and an arm inductor. Two arms are connected in a *leg* and the middle point forms an AC terminal. The arms on the top are connected to the DC positive terminal and named *upper* arms. The bottom arms are connected to the DC negative terminal and termed *lower* arms. The DC terminals may or may not be used depending on the application. The diagram of Fig. 2.3 shows the converter studied.

The symbols we adopt for the currents and voltages in this Modular Multilevel Converter are as follows: the arm inductance is called L and a resistor R models the losses; the phases are named a , b and c and the subscripts u and l indicate upper and lower arms magnitudes respectively ($u_{u\{a,b,c\}}$, $u_{l\{a,b,c\}}$, $i_{u\{a,b,c\}}$, and $i_{l\{a,b,c\}}$, where one instance is taken from $\{\}$). The AC phase voltages and currents receive the phase subscript ($u_{\{a,b,c\}}$ and $i_{\{a,b,c\}}$), while the line voltages names indicate between which phases (u_{ab} , u_{bc} and u_{ca}). The DC voltage and current receive the subscript dc (u_{dc} and i_{dc}).

Considering the circuit diagram of Figure 2.3, the sum of the voltage of the inserted cells, i.e., those that have the internal capacitor connected to the terminals, forms the upper and lower arm voltage, (2.1) and (2.2), respectively, where $u_{u\{a,b,c\}i}$ and $u_{l\{a,b,c\}i}$ represent the i -th upper or lower cell capacitor voltage, respectively, of phase a , b or c , and $s_{u\{a,b,c\}i}$ and $s_{l\{a,b,c\}i}$ are the switching functions of the i -th upper or lower cell, respectively, of phase a , b or c . For the half-bridge cell, the switching functions can assume two values: 1, when the top switch is on, and 0, when the bottom switch is on. For the full-bridge, additionally -1 is possible. The number of inserted cells in the upper and lower arms, known as insertion index (2.3).

$$u_{u\{a,b,c\}} = \sum_{i=1}^n s_{u\{a,b,c\}i} \cdot u_{u\{a,b,c\}i} \quad (2.1)$$

$$u_{l\{a,b,c\}} = \sum_{i=1}^n s_{l\{a,b,c\}i} \cdot u_{l\{a,b,c\}i} \quad (2.2)$$

$$n_{\{u,l\}\{a,b,c\}} = \sum_{i=1}^n s_{\{u,l\}\{a,b,c\}i}, \quad (2.3)$$

Using Kirchoff's law, we can express the AC phase voltage of phase a (the same is valid for the other phases) as (2.4) and (2.5).

¹See Fig. 1.9 for other possibilities.

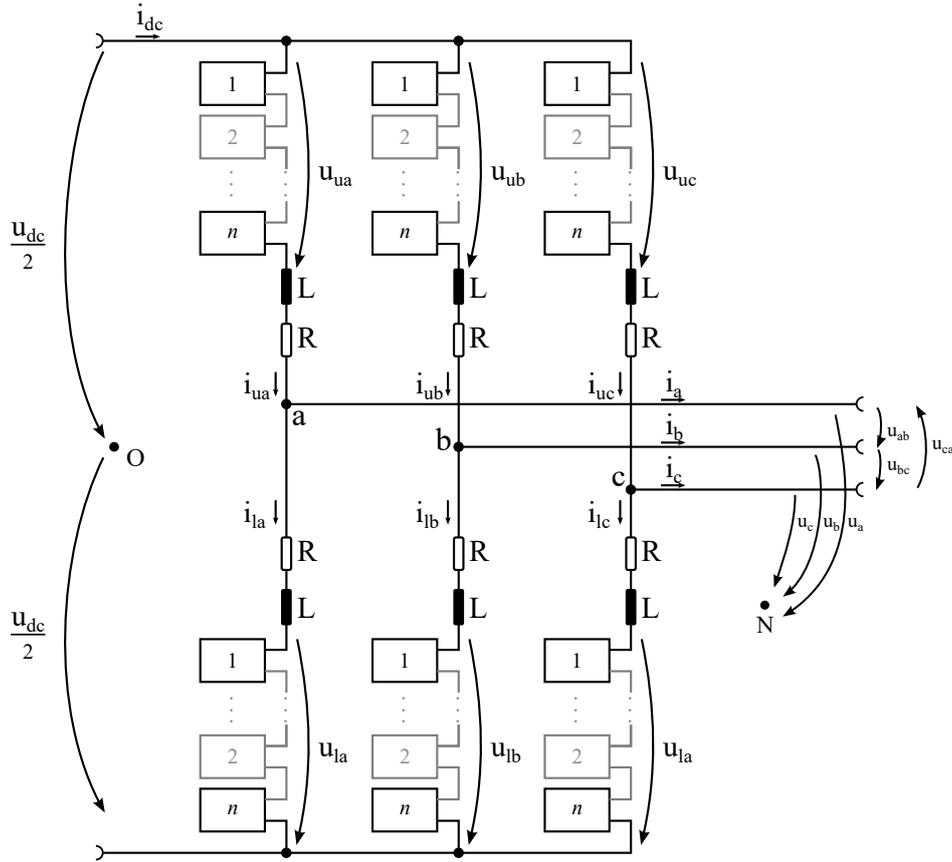


Figure 2.3: Modular Multilevel Converter circuit diagram.

$$u_a = \frac{u_{dc}}{2} - u_{ua} - L \frac{di_{ua}}{dt} - R \cdot i_{ua} - u_{NO} \quad (2.4)$$

$$u_a = -\frac{u_{dc}}{2} + u_{la} + L \frac{di_{la}}{dt} + R \cdot i_{la} - u_{NO} \quad (2.5)$$

In an MMC, the load current is only one component of the upper and lower arm currents. A second component, known as circulating current ($i_{circ,\{a,b,c\}}$), flows through both arms and is defined as (2.6). Therefore, arm, load and circulating current attend the relationships (2.7), (2.8), and (2.9).

$$i_{circ,a} \triangleq \frac{i_{ua} + i_{la}}{2} \quad (2.6)$$

$$i_{ua} = \frac{i_a}{2} + i_{circ,a} \quad (2.7)$$

$$i_{la} = -\frac{i_a}{2} + i_{circ,a} \quad (2.8)$$

$$i_a = i_{ua} - i_{la} \quad (2.9)$$

If we sum (2.5) with (2.4), the result is the expression (2.10). Replacing the arm currents by (2.7) and (2.8) in (2.10), we obtain the (2.11) that models the dynamics of the load current.

$$2u_a + 2u_{NO} + u_{ua} - u_{la} = -L \left(\frac{di_{ua}}{dt} - \frac{di_{la}}{dt} \right) - R \cdot (i_{ua} - i_{la}) \quad (2.10)$$

$$\frac{L}{2} \cdot \frac{di_a}{dt} + \frac{R}{2} \cdot i_a = \frac{u_{la} - u_{ua}}{2} - (u_a + u_{NO}) \quad (2.11)$$

On the other hand, if we subtract (2.5) from (2.4), it results in the expression (2.12). Then, substituting the upper (2.7) and lower (2.8) arm currents in (2.12), we obtain the dynamic equation of the circulating current (2.13).

$$u_{dc} - (u_{ua} + u_{la}) = L \left(\frac{di_{la}}{dt} + \frac{di_{ua}}{dt} \right) + R \cdot (i_{ua} + i_{la}). \quad (2.12)$$

$$L \frac{di_{\text{circ},a}}{dt} + R i_{\text{circ},a} = \frac{u_{dc}}{2} - \frac{u_{ua} + u_{la}}{2} \quad (2.13)$$

Observe from (2.13) and (2.11) that the circulating current is influenced by the mean value of the arms voltages and the DC voltage, while the difference between the arm voltages governs the load current dynamics. To illustrate the MMC voltage generation, Fig. 2.4 shows the upper and lower arm voltages and the generated terminal voltages of a five-level MMC using a carrier-based modulation.

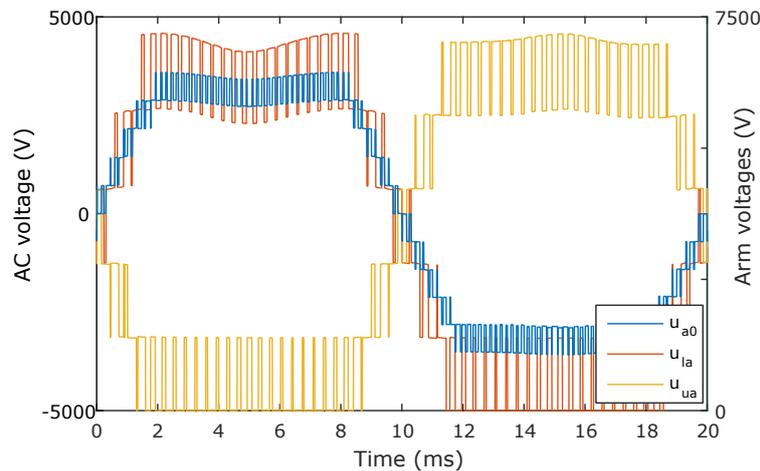


Figure 2.4: Arms and the terminal voltage of a five levels MMC.

2.2. Control of Terminal Magnitudes

The essence of a Modular Multilevel Converter is to control the power flow in its AC or DC terminals. The precisely controlled variables depend upon the application: in a

Static Synchronous Compensator, reactive power is consumed from or supplied to the grid; in an HVDC transmission line or other back-to-back applications, one converter is responsible for keeping the DC voltage level while the other will control the flow of active power; or in a Drive, it controls the AC currents.

No meaningful differences in the control of these variables exist between the MMCs and others Voltage Source Converters, like two- or three-level ones [54]. Therefore, most of the knowledge built around these Voltage Source Converters can be directly applied to the control of MMCs and will not be repeated here (refer to [55]). We assume that the control measures the AC currents and voltages periodically and, whatever the control law of choice, it has as output the phase voltages to be synthesized by a modulation strategy, as Fig. 2.5 illustrates.



Figure 2.5: Abstract control.

The development of strategies to control the output magnitudes is not an objective of this work. Nevertheless, as some strategy is necessary for the operation of the converter and its simulation, we opted to use Proportional-Integral (PI) and Proportional-Resonant (PR) controllers in a synchronous dq -frame. A Dual Second Order Generalized Integrator-Frequency Locked Loop (DSOGI-FLL) [56] provides the synchronization with the grid. An internal loop controls the currents and, depending on the application, external ones control the active power, reactive power or voltage, as Fig. 2.6 illustrates.

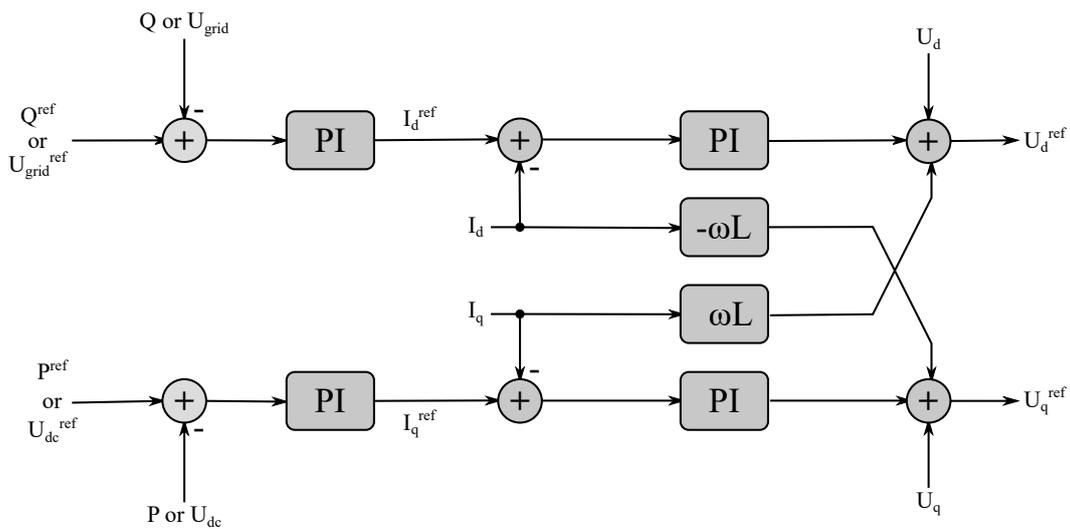


Figure 2.6: Power/Voltage and current control loops considered.

2.3. Specific Control Objectives and Methods

As a Voltage Source Converter, the MMC actuates in the power circuit by controlling the voltage in its terminals, hence researchers proposed several strategies for this purpose. We refer to them broadly as modulation methods, even though sometimes they do not involve a modulation rigorously. It is the first specific control objective discussed in this sub-section.

When a cell is inserted, the arm current flows through the capacitor, charging or discharging it. As not all cells are charged/discharged simultaneously, their voltages might drift apart unless the control prevents them. The balancing of the capacitor voltage is the second specific control objective discussed in this sub-section.

Balancing and Modulation have a close link, so authors tend to propose balancing strategies associated with a specific type or group of modulators. For the sake of simplicity, this text considers modulation and capacitor balancing independently. For a network controlled MMC, both have the most influence on the amount of data handled by the network and the timing requirements.

Besides these two, the Modular Multilevel Converter control may actuate to reduce the circulating current or to control it to shape the capacitor voltage ripple, what is the last specific control objective analyzed here.

2.3.1. Modulation

The reference voltage can be generated in different ways. There are essentially three classes of methods to generate the reference for an MMC, namely, over short periods (e.g. Pulse Width Modulation), instantaneously (e.g. Nearest-Level Control), or over a period of the fundamental (e.g. Selective Harmonic Elimination). Following we present these strategies in detail.

2.3.1.1. Short Period Modulation

Space Vector Modulation was the first modulation proposed for Modular Multilevel Converters in Leniscar and Marquardt inaugural paper [6]. The concept is the same for two-level converters. The states are located in a two dimensions plan, Fig. 2.7, and the three closest to the desired voltage vector are chosen.

The number of states has a relationship with the power of three of the number of levels (2.14), but the number of different states increases with the power of two (2.15) [6] due to redundancy, i.e., states that generate the same voltage output. Though nowadays the processors are powerful to handle the necessary calculations, this method is time-consuming to implement and to verify. Additionally, the calculation time is considerably

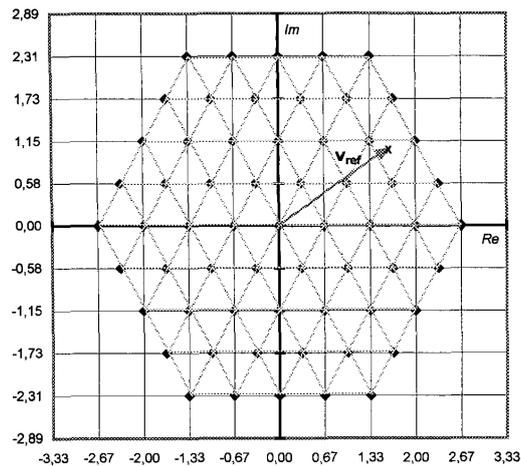


Figure 2.7: Space-vector diagram of a 5-level converter. [6]

higher than other possibilities, which will affect the minimum control cycle period. All combined, Space Vector Modulation is not a favorite modulation for MMCs.

$$\text{Number of states} = n^3 \quad (2.14)$$

$$\text{Number of different states} = 3 \cdot n \cdot (n - 1) + 1 \quad (2.15)$$

Carrier-Based Modulation compares a reference voltage with a carrier, typically a saw-tooth or triangle waveforms, and it uses the result to control the state of the power switches. It traces back to the early ages of Pulse Width Modulation (PWM) generation, when implementation was analog (Fig. 2.8). Many possibilities for the carrier waveform exists (Fig. 2.9): (a) Phase Disposition (PD), (b) Phase Opposition Disposition (POD), (c) Alternate Phase Disposition (APOD), (d) Saw-tooth and (e) Phase-Shifted Carrier (PSC).

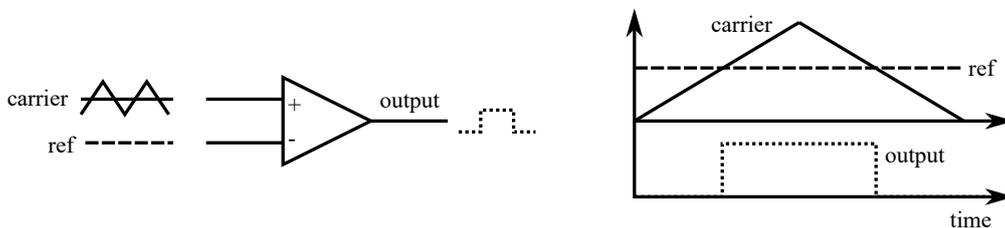


Figure 2.8: Carrier-based PWM.

The phase-disposition carriers (waveforms a, b and c), also known as level-shifted carriers, distribute the switching losses unevenly because the reference stays longer in the band level that corresponds to its peak value. Rotation methods assign the carriers to different cells to prevent thermal stress and also improve balancing [57].

Often when comparing the carrier based modulations, authors select a lower effective switching frequency for the phase-disposition carriers and make wrong conclusions: the phase-shifted carrier has better Total Harmonic Distortion (THD), but higher switching frequency and lower efficiency [58]. For a correct comparison, refer to [59].

As we will see next, some PWM strategies are equivalent when the controller executes a sorting balancing strategy, because it keeps reassigning the carriers to different cells. A pure carrier-based modulation, in which the comparison between the carrier and the set point directly defines the state of the power switches, is less common. The works that have this direct link based on the balancing strategy proposed in [7], in which every cell adds a local contribution to the reference for balancing the voltages within a certain time window, as we will see in subsection 2.3.2.4.

Decoupled Pulse Width Modulation is the association of carrier modulation methods with sorting balancing (discussed in subsection 2.3.2.2), such that the comparison of the reference with the carrier triggers a switching event that will take place in a cell chosen by the selection process. With such scheme, we lose the direct connection between cell and carrier, hence its name.

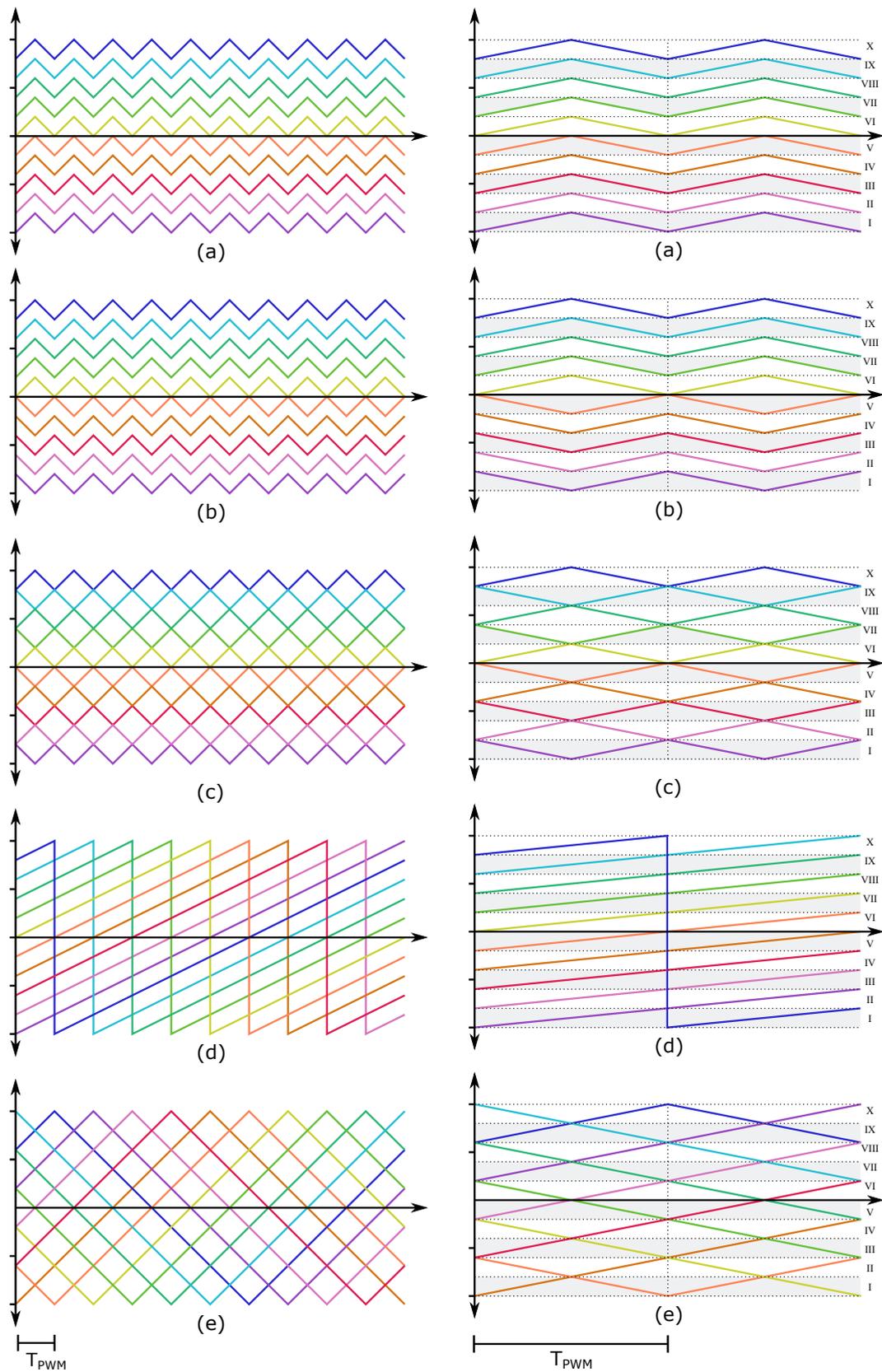
If we look into the carrier waveforms more closely, we identify the bands I to X, marked in Fig. 2.9b to have only three possibilities: the carrier counts up then down, down then up, or only up. Though the modulations differ a bit, they match in several regions. For instance, APOD carriers are equal to PSC; the PD regions V to X correspond to those of the POD, and also match the APOD bands VI, VIII, and X and PSC bands VII and IX. Therefore, it is possible to define a unified PWM strategy, as follows. Consider that the insertion index is a real number, calculated using (2.16), where $u_{\{u,l\}}^{ref}$ is the arm voltage reference.

$$n_{\{u,l\}} = n \cdot \frac{2 \cdot u_{\{u,l\}}^{ref}}{u_{dc}} \quad (2.16)$$

The modulator splits the insertion index into two terms, an integer that indicates the number of cells to insert during the next modulation cycle (2.17); and a fraction that is responsible for removing the error between the applied voltage and its reference. In (2.17) and (2.18), $\lfloor x \rfloor$ returns the greatest integer less than or equal to x and $frac(x)$ represents the decimal part of x .

$$n_{\{u,l\}}^{int} = \lfloor n_{\{u,l\}} \rfloor \quad (2.17)$$

$$D_{\{u,l\}} = frac(n_{\{u,l\}}) \quad (2.18)$$



(a) Several PWM periods.

(b) A PWM period.

Figure 2.9: Multilevel carrier PWM: (a) Phase Disposition, (b) Phase Opposition Disposition, (c) Alternative Phase Opposition Disposition, (d) Saw-tooth and (e) Phase-shifted carriers.

Depending on the carrier phase-shift to the PWM period, it emulates one of the several carrier modulation methods [60, 61].

An alternative proposal [62] is to select the minimum number of cells inserted with (2.17), but to switch one more cell in order to keep the flux error ψ_{diff} inside the tolerance band δ , such that it obeys to (2.19).

$$-\delta < \psi_{\{u,l\}diff} = \int (u_{\{u,l\}} - u_{\{u,l\}}^{ref}) dt < \delta. \quad (2.19)$$

2.3.1.2. Instantaneous Voltage Modulation

Nearest Level Control (NLC), also known as staircase modulation, is a simple to implement modulation strategy that aims at an instantaneous approximation of the reference by the nearest level available. To do so, it rounds the insertion index to the nearest integer [59], as in (2.20), where $\lceil x \rceil$ returns the lowest integer higher or equal to x .

$$n_{\{u,l\}}^{NLC} = \begin{cases} \lfloor n_{\{u,l\}} \rfloor & n_{\{u,l\}} < \lfloor n_{\{u,l\}} \rfloor + 0.5 \\ \lceil n_{\{u,l\}} \rceil & n_{\{u,l\}} \geq \lfloor n_{\{u,l\}} \rfloor + 0.5 \end{cases} \quad (2.20)$$

In Nearest Level Control, the apparent switching frequency seen by the load is not well established, as it depends on the sampling frequency and the voltage reference waveform², but we will consider the critical sampling frequency³ (2.21) as the uniform sampling⁴, where M is the modulation index and f_1 is the fundamental frequency (typically 50 Hz or 60 Hz). A higher sampling frequency would reduce the synthesized waveform phase error (Fig. 2.10), but the improvement concerning THD is modest [63].

$$f_{critical} = \pi f_1 M n \quad (2.21)$$

As we make the modulation and control frequency the same, it results either in limiting the modulator calculation frequency due to the computational capacity of the controller or reducing the sampling frequency to avoid increased switching losses⁵. The undesirable consequence of the latter is a reduced controller bandwidth and of the former it is a lower voltage waveform quality [63].

A better approach is to decouple modulation and control update rates. A possible implementation could use a synchronous (e.g., dq0) reference frame for the controller, but update the angle employed in the transformation of the control outputs back to ABC

²When the reference is sinusoidal, each cell will switch twice per fundamental cycle.

³With such sampling frequency, no more than one cell switches in a sampling period when the modulator has a sinusoidal voltage reference as input and the apparent switching frequency would be $n f_1$.

⁴A converter is uniformly sampled when sampling and switching have the same frequency.

⁵If the controller changes the reference more often, the power switch could change its state more often.

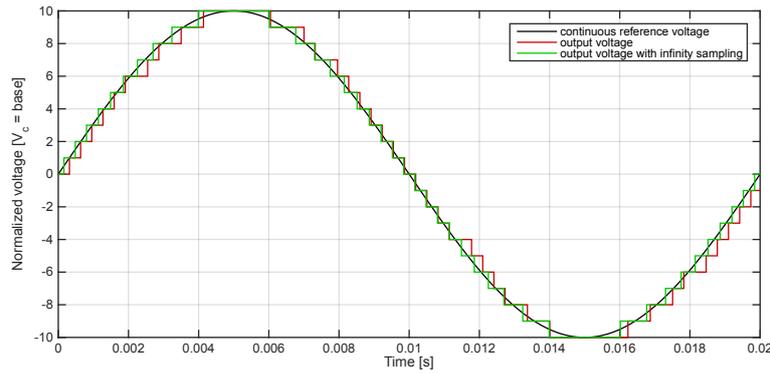


Figure 2.10: Output voltage of NLC with critical and infinite sampling in an MMC with 20 cells per arm.

by assuming a constant grid frequency between control cycles. Hence, the modulator could execute the necessary number of times to guarantee at least the critical rate (2.21) independent of the control sampling time. The implied computational load is generally low but, if needed, a Field Programmable Gate Array (FPGA) can be used to carry out computations in hardware.

We simulated a Modular Multilevel Converter with 20 cells per arm in MATLAB/Simulink with a constant control sampling period of $500\ \mu\text{s}$ and two periods for the modulation: $500\ \mu\text{s}$ and $50\ \mu\text{s}$. The upper arms outputs of the modulator are presented in Fig. 2.11, where the benefit of the higher modulation frequency is visible. Note that the shorted modulation period causes only one cell to switch each time; as a result, the voltage waveform has a higher resolution than with the lower modulation update rate, although it is still the same converter.

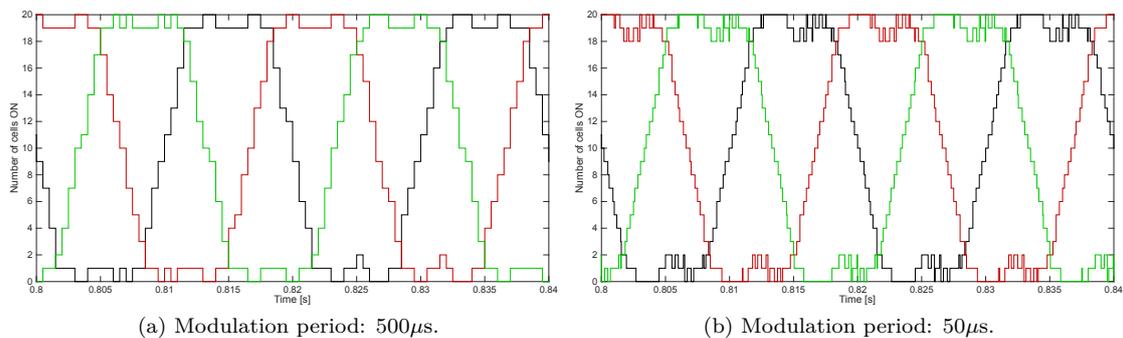


Figure 2.11: Upper arm command of a MMC with NLC modulation.

2.3.1.3. Fundamental Frequency Modulation

The fundamental frequency modulators differ from the previous two types in that they aim at generating a waveform at the output of the converter such that its fundamental frequency has the desired magnitude and phase. The Fourier series represents any periodic waveform with a sum of sines and cosines (2.22).

$$u = \sum_{h=1}^{\infty} a_h \cos(h\omega t) + b_h \sin(h\omega t) \quad (2.22)$$

As the waveform of Fig. 2.12 has odd and quarter-wave symmetry, its Fourier representation has only sine terms of odd order (2.23) with each term weight calculated by (2.24) when all voltage steps have the same height [64].

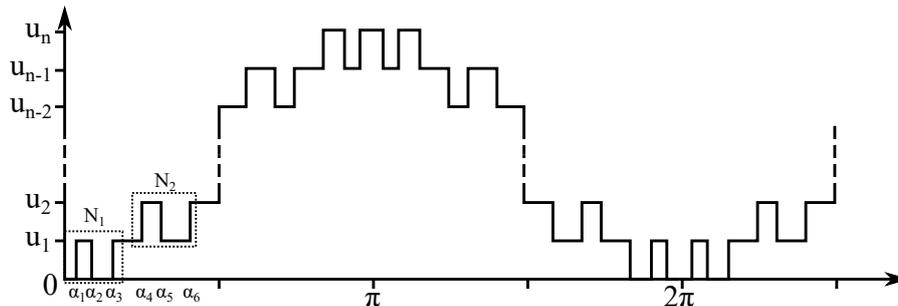


Figure 2.12: Generalized multilevel waveform.

$$u = \sum_{h=1,3,\dots}^{\infty} b_h \sin(h\omega t) \quad (2.23)$$

$$b_h = \frac{4 \cdot u_{dc}}{h\pi} \cdot \left[\sum_{i=1}^{P_1} (-1)^{i+1} \cos(h\alpha_i) \pm \sum_{i=P_1+1}^{P_2} (-1)^i \cos(h\alpha_i) \pm \dots \right. \\ \left. \pm \sum_{i=P_{n-1}+1}^{P_n} (-1)^i \cos(h\alpha_i) \right] \quad (2.24)$$

In (2.24), $h = 1, 3, 5, \dots, 2N - 1$ if N is odd or $h = 1, 3, 5, \dots, 3N - 2$ if N is even; $N = N_1 + N_2 + \dots + N_n$ is the total number of pulses per-quarter cycle; N_1, N_2, \dots, N_n are the number of pulses per-quarter cycle at cell 1, 2, ..., n ; $P_1 = N_1, P_2 = N_1 + N_2, \dots, P_n = N_1 + N_2 + \dots + N_n$; n is number of DC sources (number of cells per arm in an MMC); α_i is the i -th switching angle; and the polarity \pm is positive if P_{n-1} is odd and negative otherwise.

One modulation of this kind is the Selective Harmonic Elimination that chooses the switching angles α_i such that the fundamental frequency has the desired magnitude and it eliminates low order components, such as the third, fifth, and seventh harmonics. Typically, the calculation of the switching angles has high computational effort and involves solving transcendental equations [29].

Ilves et al. [65] employ this concept, but restrict the pulses of the individual cells to a square wave centered around an angle γ . The fundamental component of the sum of n square waves, each with a different angle $\gamma_i, i \in 1..n$, is given in (2.25), where A and B

are expressed by (2.26) and (2.27).

$$H_1 = \frac{4}{\pi} (A + B) \quad (2.25)$$

$$A = \left[\sum_{i=1}^N \cos(\gamma_i) \right] \cos(\omega t) \quad (2.26)$$

$$B = \left[\sum_{i=1}^N \sin(\gamma_i) \right] \sin(\omega t) \quad (2.27)$$

Again, by choosing the angles γ_i , the modulation can generate a waveform with the desired fundamental frequency amplitude and cancel some of the first order harmonics.

Fundamental frequency modulators yield low THD in steady state, but have problems coping with transients, because the controller can only change the modulation index slowly, resulting in potentially high overcurrents.

2.3.1.4. Carrier-based Vs NLC modulation

The two most common modulation strategies in the literature are the carrier-based and the NLC. Among the carrier-based modulation, the PSC stands out because it eliminates the harmonic content of the AC voltage at and around the carrier frequency and its multiples; the first harmonics present are the sidebands of n times the carrier frequency [59]. Additionally, the PSC cancels the high frequency content of the DC voltage.

For large number of cells, the PSC and NLC will exhibit similar THD performance, but, as the NLC spectrum is not-characteristic, i.e., it contains almost all harmonic orders, the filter requirement of the PSC is lower [59]. On the other hand, as the NLC switching frequency is typically equal to the fundamental one⁶, it has lower switching losses. The Selective Harmonic Elimination (SHE) modulation has a waveform similar to the NLC, but as it selects the switching angles to eliminate the lower harmonics, it stands between the two previous modulation strategies in terms of filtering requirements.

2.3.2. Capacitor Voltage Balancing

The MMC control must guarantee the stability of the capacitor voltages in an arm (i.e., fluctuation around a mean value) for proper operation of the converter. It is also necessary to make the voltage steps more uniform and reduce the harmonic content, though the effect in the output voltage and current is weak in converters with a high number of cells [66].

As the cells are equal, the controller has (directly or indirectly) the degree of freedom to chose which ones will be inserted or bypassed, so it can keep the voltage of the cells

⁶Though it highly depends on the balancing strategy and reference waveform.

balanced. When inserted, the current flows through the capacitor, charging or discharging it depending on the current polarity and how the capacitor is inserted (with Full-bridge cells); when bypassed, the capacitor voltage remains constant.

The periodic nature of the insertion index and the arm current induces a ripple in the mean capacitor voltage [59]. The ripple can be estimated using (2.28) (as in [66]), where ω_0 is the grid angular frequency, I_{ac} is the AC terminal RMS current, M is the modulation index, ϕ is the angle between the AC phase current and voltage, and C is the cell capacitance.

$$\Delta u = \frac{1}{C} \cdot \frac{1}{4\omega_0} I_{ac} \left[1 - \left(\frac{M}{2} \cdot \cos\phi \right)^2 \right]^{\frac{3}{2}}, \quad (2.28)$$

The balancing strategy aims to keep the capacitor voltages inside this Δu band centered at the rated value (typically u_{dc}/n). We can classify the various balancing strategies as natural, based on sorting (including several variations of sorting), using closed-loop control, open-loop or partially open-loop control, and still others that do not fit in the previous classes.

2.3.2.1. Natural Balancing

Ilves et al. [67] showed that the capacitor voltages could be stable without a balancing strategy. For it to occur, a necessary condition is that the switching frequency is not an integer multiple of the fundamental frequency, but the effective switching frequency (n times the carrier frequency for PSC modulation) might be. They also suggested that it is advantageous to choose an effective switching frequency that is an even harmonic of the fundamental, because it eliminates even harmonics on the grid side and odd harmonics on the DC side. We refer to this approach without an explicit balancing strategy as Natural Balancing.

The argument is that, if the capacitor voltage can naturally keep a certain level of balance, the proper choice of the carrier frequency can reduce the number of unnecessary switching (when an inserted cell is substituted by a bypassed one). In other words, Natural Balancing can reduce switching losses, but a formal proof is missing.

In Natural Balancing, capacitor voltage measurements are optional and the central controller requires no balancing action. However, it is hard to guarantee stability for all the grid conditions and faults that a converter might be exposed to during its lifetime. For the high availability applications that are typical of MMCs, it is difficult to sponsor it as a suitable solution.

2.3.2.2. Sorting Algorithm

Lenicar and Marquardt [6] adopted for the balancing of the capacitor voltages what later was named Sorting algorithm. This method sorts the cells according to the capacitor voltage magnitude and selects which ones will be inserted or removed depending on the arm current polarity. A positive current will charge the capacitors of the inserted cells, so those with lowest voltages must be on. A negative current will discharge the capacitors of the inserted cells, so those with the highest voltages must be selected.

Sorting and selecting is an effective solution and the most popular strategy in the technical literature [62, 66, 68–79], but it penalizes the converter efficiency with unnecessary switching. Cells might change their state even when the commanded number of inserted cells remains constant between two control cycles [36]. Additionally, it imposes two constraints to the system controller: all capacitor voltages are needed simultaneously at the central controller and the sorting process is computationally expensive (complexity is at least $O(n \cdot \log n)$ for the worst-case scenario [61]).

Regarding the sorting computational burden, though considerable, it can be tamed, especially if we consider that MMCs are large and expensive systems that can resort to high-performance hardware. For instance, Texas Instrument benchmarked sorting algorithms in a C64x+ processor core and measured a worst-case performance of 50 cycles per 32-bits word for the Merge Sort between 1 and 256 words (the best performing method). This core runs with a clock frequency between 500 Mhz and 1.2 GHz and needs, respectively 20 μ s or 8.3 μ s to sort a list with 200 members. Sklyarov and Skliarova [80] proposed an efficient and resource lean recursive implementation for FPGAs that need $\lceil \log_2 n \rceil (\lceil \log_2 n \rceil + 1) / 2$ steps for sorting, so 36 steps for 256 members or 360 ns with 100 MHz logic gates. Ricco et al. [73] implemented a Bitonic Sorting Network on an FPGA that take approximately between 38 μ s and 75 μ s to sort a list of 256 members.

Unlike some authors claim [61, 73], the sorting can update the list less often than the control algorithm, as proposed by Qin and Saeedifard [71]. Between updates, the modulation selects the cells to insert or bypass according to the last list available. In this work, the simulations show that the sorting can balance the capacitor voltages even if the sorting update rate is twice the fundamental frequency. With such low update rates, it is possible that the system becomes unstable, but the conditions when it occurs are unclear.

Later in this dissertation, in Chapter 6, we present experiments employing delayed capacitor measurements in the sorting algorithm. When the delay is of just a few samples, it has little effect on the quality of the balancing. Accordingly, we can infer that if the sorting takes more than one control period, but the update rate is still the same, the algorithm can still keep the voltages well balanced.

2.3.2.3. Variations of the Sorting Algorithm

Several authors proposed alternative methods or small modifications to overcome the drawbacks of the sorting method. Following, we review some of these works.

Dommaschk et al. [70] adopted a minimum voltage threshold (or tolerance band, as described in [62]) that allows an exchange between cells with the purpose of reducing the switching frequency due to unnecessary changes. Y. Li et al. [66] calculate the upper and lower boundaries of the switching frequency and voltage threshold product analytically. Besides being useful for choosing the voltage threshold and forecasting the power losses, their analysis shows good converter performance with switching frequencies of only a few hundreds of Hertz. Since the capacitor charging and discharging depends on the operating point, so will the average switching frequency [62].

Another proposal for reducing the switching frequency is to make changes when the number of inserted cells changes or when any capacitor voltage goes beyond a certain limit [62, 73, 81]. A comparison between a tolerance band around the mean value and an absolute tolerance band shows the latter to result in lower average switching frequency [62].

Deng and Chen [82] proposed a method that explores the current ripple at the PWM frequency to balance the capacitors. Their reasoning is that the cell which pulses are closer to the positive peak of the ripple will have their capacitor charged more (or discharged less) and the cells which carrier are farther from the positive peak will have their capacitor discharged more (or charged less). Therefore, sorting the cells in descending capacitor voltage order and assigning the carrier from the closest to the current peak to the farthest each PWM period would balance the voltages without an influence of the arm current polarity (its main advantage).

This method has a conceptual error, because phase-shifted carriers are $2\pi/n$ phase-shifted exactly to cancel out the harmonic content multiple of the PWM frequency (the first high-frequency component is n times the PWM frequency). In other words, the method resorts to a current ripple that is not there. Furthermore, they assume the dominance of the component at the PWM frequency, but the Fourier expansion proves otherwise (Fig. 2.13). The proposed workaround was to modify the phase-shift between the carriers, with obvious handicaps on the voltage and current harmonic distortions.

Siemaszko's method [83] converts the voltage measurements into frequency square signals and transmits them to the arm controller, thus assuming a star topology. There, it counts the transitions of the square signals. As higher voltage translates into higher frequency, the signal from the cell with the highest voltage will be the first to reach a predefined counter value and the one with the lowest will be the last. Then, the arm controller selects a cell to insert or bypass depending on the modulator reference, the known cell states (inserted or bypassed) and the current polarity.

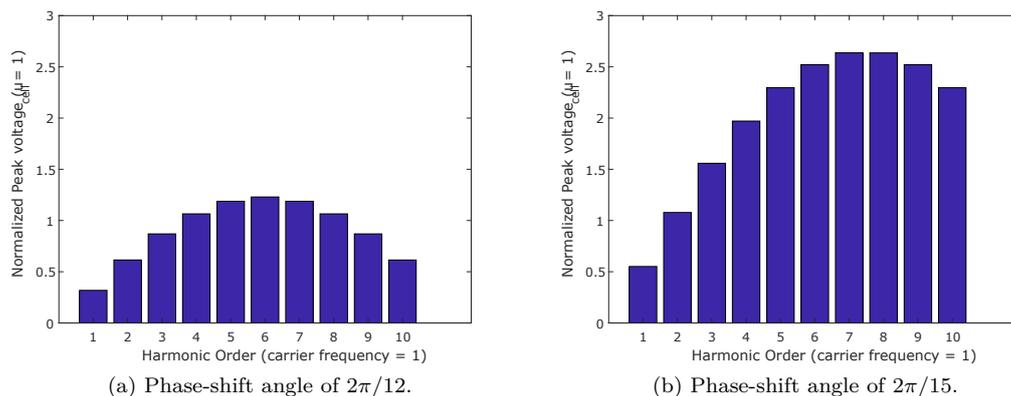


Figure 2.13: Harmonic content of the arm voltage in a PWM cycle. The first harmonic magnitude is small compared with others ($n=10$).

The time required for obtaining the cell with the least voltage, equivalent to the time to finish the sorting, depends on the minimum signal frequency and the predefined counter value. The latter must be high enough to reduce the inherent inaccuracies due to the asynchronous relationship between the several clocks. Low-cost fiber optics limits the maximum frequency to a few tenths of MHz and the minimum frequency ends in the range of hundreds of kHz. As a consequence, this method takes tenths of microseconds ($32 \mu\text{s}$ in the author's experiments) to complete, the same order of magnitude of the traditional sorting. Nevertheless, it is innovative in the sense that it distributes part of the sorting to the cells. Later in Chapter 4, we will build upon a similar concept to carry out a distributed sorting with low traffic in a network controlled MMC topology.

An algorithm that finds the k -th minimum or maximum elements can reduce the complexity to $O(n)$ [61]. While bubble sorting takes $20 \mu\text{s}$ to order 1024 values, the identification of the minimum and maximum values completes in 0.5 ns using 21 times fewer resources implemented in an FPGA [61]. Due to its speed, the controller can run the Min/Max identification several times if more than one cell needs switching, hence this method is immune to dynamic problems mentioned in [73], such as deviations from the reference or current spikes. Saad et al. [81] also used the minimum/maximum values for the balancing and likewise had a fast update of the measurements and identification of the minimum/maximum cells. This fast update rate is a problem for network controlled MMCs, because it demands the measurements to be available at the central controller at a high rate.

2.3.2.4. Closed-Loop Balancing

The control strategy proposed in [7] is based in Phase-Shifted Carrier PWM. The voltage command is the sum of the terminal voltage reference, the feedforward of the DC voltage, and two other terms coming from the averaging and the balancing control loops (Fig. 2.14a).

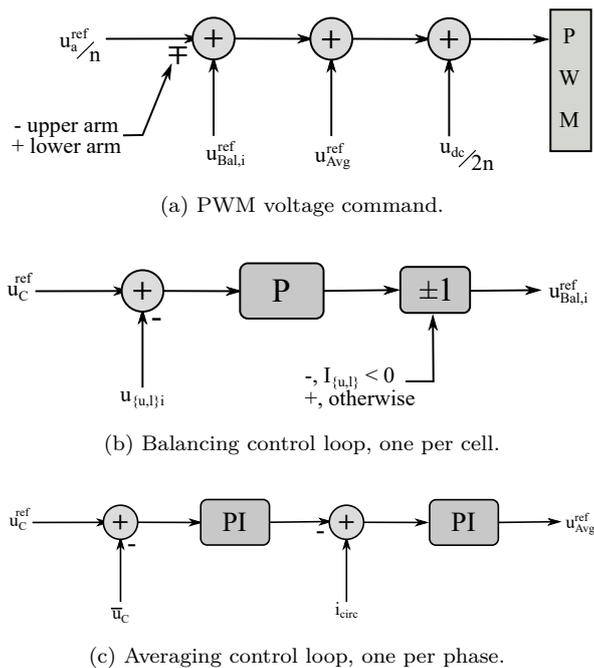


Figure 2.14: Control strategy with averaging and balancing terms. [7].

The balancing term is the output of individual control loops for each cell that compares the capacitor reference with the measurement. When the error is positive, the capacitor needs charging and the cell has to draw active power from the DC source, thus if the arm current is positive, the balancing term must also be positive; if the current is negative, it must be negative (Fig. 2.14b). Changing the balancing term in this way results in a discontinuous sawtooth waveform, thus causing low-order harmonics in the arm currents.

The averaging loop (Fig. 2.14c) forces the average capacitor voltage (2.29) to the reference by controlling the DC component of the circulating current in the inner loop. It has the same effect of regulating the DC link voltage.

$$\bar{u}_C = \frac{1}{n} \sum_{i=1}^n u_{ui} + \frac{1}{n} \sum_{i=1}^n u_{li} \quad (2.29)$$

Hagiwara, Maeda and Akagi [84] analyzed the stability of this strategy and found that it can be unstable under certain operating conditions. To prevent oscillations, they proposed the inclusion of an arm balancing control loop with the purpose of reducing the voltage difference between the capacitor voltages of the upper and lower arms.

A partly distributed implementation of this balancing method runs the averaging and balancing control loops in each cell [43–45, 85]. The central controller communicates with the cells the terminal voltage setpoint, the circulating current measurement, the load current, the capacitor voltage setpoint, and the average capacitor voltage. It calculates this last term with (2.29), hence the cells must send their measurements to the central controller. To avoid this traffic, Seleme et al. [86] proposed that each cell uses as reference

to the balancing control the mean voltage between the two neighboring cells, and only one master cell per arm would have its capacitor setpoint modified to regulate the DC link voltage.

Yang et al. [8] also removed the necessity of transmitting the capacitor voltages from the cells to the central controller, but went further and proposed having data flowing only from the central controller to the cells. The central controller is responsible for generating a common terminal voltage reference to control the load current, the reference for the circulating current loop, and the capacitor voltage setpoint, typically equal to u_{dc}/n . It also sends the load current angle, that corresponds to the fundamental frequency oscillation of the arm current, and the measured circulating current (Fig. 2.15a).

The authors transferred the circulating current controller to the cells and proposed a PR controller that removes the second harmonic component of the circulating current. The local loops have a common setpoint and feedback arriving from the central controller, but also a local term from the local average capacitor voltage control (Fig. 2.15b). As the error between the capacitor voltage setpoint and the local feedback may have different polarities among the cells, they might compete against each other. As a consequence, the average capacitor voltage gain has to be low to avoid oscillations, but the feedforward term (i_{diff_dc}) improves its regulation performance. Differently from the other closed-loop balancing strategies, Yang et al. use the output of the proportional balancing controller to set the peak value of sine in phase with the load current, thus avoiding the discontinuities of the balancing term as described before.

In Chapter 4, we will explore the performance limits of a ring network designing a protocol that explores the control strategies proposed by Seleme et al. [86] and Yang et al. [8].

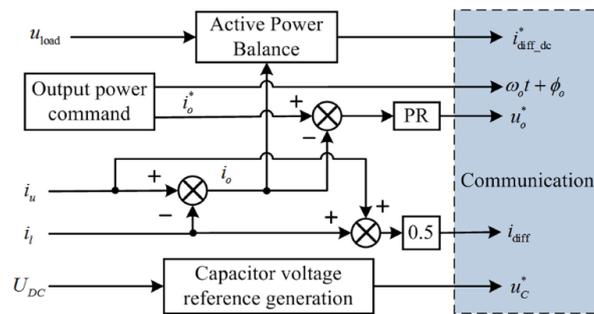
2.3.2.5. Open-loop or Partial Open-loop

The Loop Bias Mapping approach [57] rotates the carrier assignment of the Phase Disposition carriers between the cells of an arm periodically. This method increases the switching losses, as every time that the carriers are re-organized at least two cells switch. The Selective Loop Bias Mapping uses the same rotation principle, but reserves the upper and lower carriers to the cell with the maximum and minimum voltages, respectively.

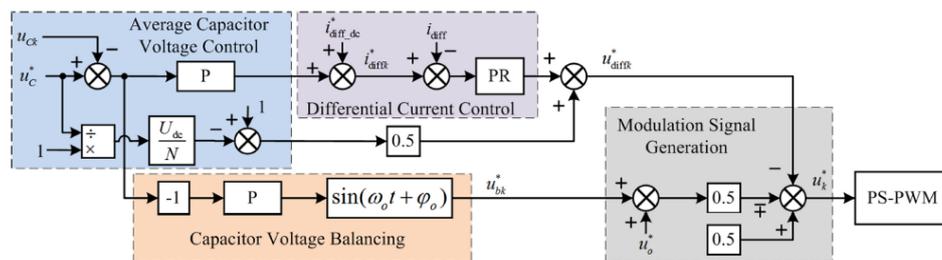
Similarly, Hassanpoor et al.'s *CTBsequence*⁷ method [62] monitors the capacitor voltages and replaces an inserted capacitor with the first bypassed one from an assignment vector when its voltage is outside the tolerance band $[u_{max}, u_{min}]$. Moreover, this strategy reverses the assignment vector every second cycle to ensure “balanced capacitor voltages”, though proof is missing.

In fundamental frequency switching (see 2.3.1.3), when each cell switches only twice

⁷Cell Tolerance Band + Sequence Reversing.



(a) Control loops at the central controller.



(b) Control loops and PWM generation at the cells.

Figure 2.15: Partly distributed control of MMCs using phase-shifted carrier PWM. [8].

per cycle, the control has to perform the balancing over more than one cycle. One proposal is to assign a sequence of angles, such as the net transfer of charge in one cycle counterbalance the previous one [65]. Interestingly, though not proved formally, simulation and experimental results show the capacitor voltage converging after perturbation in the voltages of some cells.

Fan et al. [87] proposed a carrier-based modulation that spreads the switching events evenly between all the cells (basically they transform PD into a PSC PWM). The authors argue that this strategy has an inherent balancing effect, though it is insufficient to guarantee balancing. Therefore, an additional balancing strategy identifies the cells with minimum and maximum voltage and modifies their pulses: if the current is positive, the controller delays the turn-on of the latter by Δt and the turn off of the first by the same amount. If the current is negative, the opposite happens.

Open-loop methods may work under steady-state conditions, but they result in higher ripple and fail to handle severe fault cases [62].

2.3.2.6. Others

Ilves et al. [88] proposed a predictive sorting strategy for converters with switching frequencies close to the fundamental, where the traditional sorting method has unsatisfactory results. Here, the algorithm divides each half-cycle of the arm current into a charging and a discharging period. It aims to balance the capacitors at the end of each period, minimizing the voltage ripple, while accepting spreads on the capacitor voltages

between the peaks.

At the beginning of a charging or discharging phase, the controller integrates the arm current to calculate the accumulated charge $Q_{\{u,l\}}^+$ and $Q_{\{u,l\}}^-$. Additionally, it associates each output level to a (virtual) cell and calculates the charge transferred or removed to/from it, $Q_{\{u,l\}i}^+$ and $Q_{\{u,l\}i}^-$, by integrating the current whenever the insertion index $n_{\{u,l\}} \geq i$. When the charging/discharging period ends, it stores the measured charges for the next cycle and uses them to calculate the target voltage $u_{\{u,l\}}^+ = u_{\{u,l\}0} + Q_{\{u,l\}}^+/n \cdot C$.

Each control cycle, the controller estimates the individual voltages at the end of the period using (2.30), where $u_{\{u,l\}0}$ is the initial voltage, $K_{\{u,l\}i}$ is equal to $s_{\{u,l\}i}$ if $n_{\{u,l\}} < i$, and equal to $s_{\{u,l\}i} - 1$ if $n_{\{u,l\}} \geq i$, and C is the cell capacitance. The integral term corrects the final voltage if the cell is inserted before the forecast ($n_{\{u,l\}} < i$) or removed while it should still be inserted ($n_{\{u,l\}} \geq i$).

$$\hat{u}_{\{u,l\}i}^+ = u_{\{u,l\}0} + \frac{Q_{\{u,l\}i}^+}{C} + \frac{1}{C} \int K_{\{u,l\}i} \cdot i_{\{u,l\}} dt \quad (2.30)$$

Finally, the algorithm inserts the $n_{\{u,l\}}$ cells whose estimated voltages are equal to the target voltage. If the number of cells that meet this condition is insufficient, the controller selects the remaining cells from those that will not overshoot the target voltage. Note that at the beginning of the cycles, ideally all the voltages are the same and the controller arbitrary chooses the cells to insert so that it may result in uneven thermal stresses [59].

This method works well in steady-state, but as the arm current or the switching functions vary from cycle to cycle, important over and undershoot occurs (see Fig. 11 in [88]).

Model Predictive Controllers can incorporate the balancing of the capacitors in their cost function together with the reduction of the average switching frequency and the circulating current, such as in [89,90]. In this case, all the capacitors voltages are necessary in the central controller.

2.3.3. Circulating Current Control

The difference in the voltage between the arms of the MMC causes the circulating current to flow, being predominantly a sum of a DC and second-order harmonic component [91]. Though it has no impact on the output magnitudes, its presence increases the peak and RMS values of the arm current, thus causing higher losses and voltage ripple.

The circulating current dynamics are governed by (2.13). Since it is influenced by the difference between the upper and lower arm voltages, in contrast to the mean value between them that defines the output voltage, the controller can regulate the phase and circulating current independently. Its control has two conflicting objectives: minimizing

the circulating current, what causes a higher voltage ripple in the cells; or shaping it to minimize voltage ripple, increasing the losses [92].

Various works proposed strategies to control the circulating current. As the second harmonic dominates the circulating current, using either a reference frame in this frequency [93], a proportional-resonant controller [94], or a repetitive controller [95]. Model predictive controllers reduce the circulating current by including a term related to it into its cost function [89, 90].

2.3.4. Control Structure

As mentioned earlier, the development of strategies to control the output magnitudes is not an objective of this work. Nevertheless, the test of the ideas proposed in this text requires the adoption of some control strategy for running both the simulation model and the converter prototype. The choice was for a standard control algorithm in the synchronous reference frame for the load currents and DC voltage, with a phase-locked loop tracking the grid voltage angle, and an α - β frame for the circulating current (Fig. 2.16). Fig. 2.17 shows a simplified circuit diagram of the double-star MMC to make clear where the control measures, with transducers connected to the central controller, the magnitudes represented by the red lines in Fig. 2.16.

2.4. Network Controlled MMCs

One important issue, that will be detailed in the following chapter, is the MMC control architecture topology. Small systems can afford direct hardware links between the central controller and the cells in a star fashion. However, this topology is impractical when the system grows in size. In that case, using a digital communication network brings several benefits:

- It supports other control architecture topologies than the star, dramatically reducing cabling;
- Simpler connections facilitate assembling and maintenance;
- The central controller hardware interface remains the same independently of the converter number of cells;
- The cells can share information with their peers, bringing the opportunity for new control strategies;
- The cells can send diagnostic data to the central controller;
- The central controller can parameterize the cells online;
- It enables to update the cells firmware while the converter is operating.

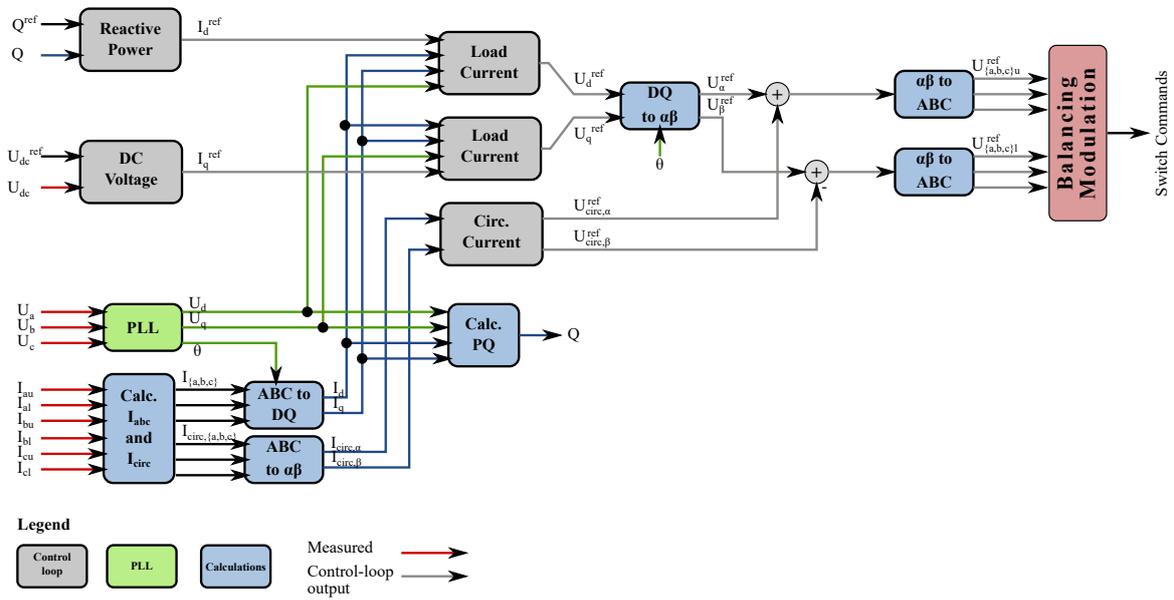


Figure 2.16: Block diagram of the control.

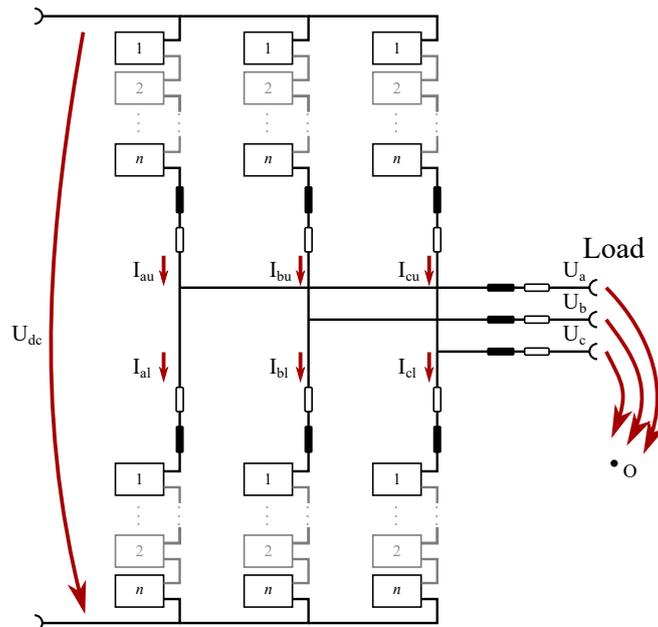


Figure 2.17: Double-star MMC circuit diagram indicating the measured magnitudes.

The introduction of the control network forces the architecture to have a central controller and embedded hardware in the cells (or group of cells), creating the opportunity to partly decentralize the control, e.g., generating the cell pulses locally or modifying the setpoint from the central controller to balance the capacitor voltages. In principle, we could envisage a completely decentralized control strategy, but such approach is not reported in the literature, yet. In this work, we explore possibilities for centralized and partly decentralized control strategies together with a digital communication network.

2.5. Conclusions

In this chapter, we reviewed the [MMC](#) operation and main control objectives and strategies. We have put a focus on the balancing and modulation strategies because they are the most relevant for the topic discussed in this dissertation.

Curiously, carrier-based (e.g., [PSC PWM](#)) modulation is the most explored strategy in the literature by far, but as the number of cells in a Modular Multilevel Converter increases, [PWM](#) ceases to be the preferable option, giving precedence to instantaneous (e.g., Nearest Level Control, [NLC](#)) or fundamental voltage synthesis (e.g., Selective Harmonic Elimination) [[35](#), p. 68].

Regarding balancing, the sorting strategy is the most popular and effective way of balancing the capacitor voltages. Its main drawback is to increase the switching frequency of the cells, but we discussed several improvements able to overcome it. However, the sorting in network controlled [MMC](#) causes increased network traffic, because the central controller needs to receive all the capacitor voltage measurements. The heavy traffic may lead to long update rates that may reverse several benefits brought by the introduction of networks. We listed such benefits at the end of this chapter as a link to distributed [MMC](#) architectures, which we will discuss in detail in the following chapter.

Chapter 3

Overview of Communication Networks for MMCs

Several authors studied the use of a digital communication network for modular converters with the purpose of improving the scalability, implementation, and maintenance of such converters.

The control structure of Modular Multilevel Converters consists of a central unit and embedded electronics in each cell or group of cells. The central controller has powerful computational resources and more access to data than the cells, e.g., measurements and setpoints from higher hierarchy levels; hence it calculates the load current and outer control loops. Then, the central controller sends references to the cells over the network, where the actuation will effectively happen.

Considering the review of the control of Modular Multilevel Converters (MMCs) from last chapter and that a few sensors can provide the necessary information to the control of the output magnitudes and the circulating current, we can point to the modulation and the balancing of the capacitors as the two functions that dominate the communication requirements between the central controller and the cells, because of a large amount of data traffic and the low latency required.

The modulation demands that the central controller sends commands to the power switches over the network, what generates a payload dependency with the number of cells. Three modes for controlling the actuation are possible: on/off commands, a reference per cell, or a single setpoint per arm [45].

In the first mode, the central controller determines the state of the power switches of the cells, i.e., it sends one bit for each power switch.

In the second mode, the cells divide the time axis with a particular resolution; hence the central controller references tell each cell in which specific moment it should insert or bypass its capacitor, either by comparing the reference with a triangular waveform [40] or sending two values, one corresponding to the turn on and another with the turn off

time [47].

In the third mode, the central controller sends arm voltage references and each cell compares this reference with a local phase shifted carrier to command the power switches [45].

Regarding the balancing of the capacitor voltages, the strategy has significant influence on the frame payload. The most popular balancing method, sorting, needs the central controller to gather all the capacitor voltage measurements before it can sort them and select which ones to insert. Another method that has the same demand is Model Predictive Control, because it includes the capacitor voltages in the cost function. In this respect, the proposals of using the minimum and maximum voltages as an alternative to the sorting bring little benefit, because their identification still needs the same flow of information.

Ilves et al.'s predictive method [88] releases the network from the heavy traffic, because it only needs the voltage measurements at the beginning of the charging/discharging cycle. Unfortunately, this method suffers from important over- and undershoots during transients and does not seem appropriate to real MMCs unless the controller monitors the voltages closely, such that it can provoke additional switching to avoid these deviations. In this case, though, the network traffic is again high.

More promising in terms of data flow are the closed-loop methods. The original control method [7] served as a base for proposals of partly distributed control architectures [8, 43–45, 85, 86], where the central controller sends setpoints and global measurements to the cells. These values, together with locally calculated terms, form the Pulse Width Modulation (PWM) reference that will be compared with the carrier and generate the pulses of the power switches. However, these methods apply only to Phase-Shifted Carrier PWM, that typically works well with relatively high switching frequencies and a low number of cells, but forces the designer to tune a high number of control loops.

In this chapter, we review the requirements imposed on the communication network due to converter characteristics. Then, we look into the main protocols proposed so far and discuss their characteristics. The solutions reviewed have as common properties the use of ring topology with a data rate of 100 Mbit/s. As it will become clear, they struggle to control MMCs that have hundreds of cells with the necessary high update rates. Therefore, we ignore other solutions employing communication links with lower data rates, such as Asynchronous Serial [96], Controller Area Network [97], or EIA-485 [46, 98].

We finish this chapter explaining how the communication network influences the sampling frequency and the loop delay, what will be a motivation for the model-based compensator proposed in Chapter 6.

3.1. Requirements

Often authors discuss the requirements of a communication network for the control of power electronics converters and draw similar solutions [37]: ring topology, Ethernet technology, fiber optics as transmission media, need of high accuracy synchronization between nodes, short communication cycle times, the presence of a central controller, and logical master/slave configuration. Though all these requirements make sense, often the authors put them as the only possibility with disputable arguments. In this section, we elaborate on the reasons for and consequences of these requirements.

3.1.1. Bandwidth and Latency

The bandwidth of a communication link is often taken as the first and most important parameter to consider. Though relevant, it alone says little about the communication performance and even the term is ambiguous: is it the maximum bandwidth, the sustained bandwidth, or the minimum guaranteed bandwidth? If the maximum bandwidth is meant, a better term choice would be link capacity.

Many other aspects, like binary encoding, minimum payload, addressing, error detection, error correction, packet loss, and protocol overhead, play an essential role in what is, in fact, the critical element: the end-to-end latency. The communication latency adds to the loop delay (Section 3.4) and reduces the control performance (Chapter 6).

3.1.2. Payload

The amount of data transported by the network can vary considerably according to the control method employed. Therefore, it is useless to try to put a typical figure to it. As just referred, the critical factor is the end-to-end latency that depends on the amount of data transmitted together within the link capacity. If the link capacity is high, the network can transmit more data without impacting the latency. On the other hand, when the capacity is low, it is crucial to consider control strategies that reduce the packet payload.

3.1.3. Reliability

Modular Multilevel Converters operate in mission-critical applications, such as high capacity High-Voltage Direct Current (HVDC) transmission lines, where a failure can lead to catastrophic consequences. Under this circumstance, the inclusion of the network must have a low impact on the overall system reliability; hence, tolerance to failure is a hard requirement of any solution.

3.1.4. Transmission Media

The use of fiber optics in Modular Multilevel Converters is recurrently deemed as *preferable* [37] or considered a *good choice* [44], but some form of optical isolation is indeed mandatory: no other solution allows transferring data with low latency while providing the required galvanic isolation for safe operation. We have, though, two possibilities for deploying the optical isolation: the first one is the communication link between the controller and the cell hardware (Fig. 3.1a); the second one is between the PWM signal generation and the gate-drivers inside the cells (Fig. 3.1b).

If the latter is chosen, the optical media for the network links is optional, but, in this case, the control strategy is a critical factor, because the control hardware is at earth potential. If the capacitor voltages have to be measured, it is not viable to place isolated transducers in all cells. Their estimation, such as in [99], eliminates the sensors in the cells and makes a case for the use of optical coupling for the PWM signals and electrical media for the communication link.

Toh and Norum [37] mention the *noisy* [Electromagnetic Interference (EMI)] *environment* as a reason to favor optical media. Huang et al. [44] point to reliability reasons stating that a *submodule* [cell] (*SM*) could “see” the whole DC link voltage if all the other *SMs* are bypassed, and high DC link voltage (...) can be passed to all the other *SM* and to the master controller (sic). Laakkonen [48] invokes both galvanic isolation of modules and EMI tolerance to prefer optical communication.

Let us discuss first the noisy environment argument. Seldom in an MMC more than a couple of cells switch simultaneously. These cells have a blocking voltage of a few kilovolts and produce similar dv/dt and di/dt as in other low/medium voltage converters. Also, each cell faces a high voltage level between its terminals and earth, so it is mandatory to provide the necessary creepage and clearance distances for both operational and safety reasons. These distances can reach the magnitude of meters, what reduces the parasitic coupling in the power circuitry and consequently the EMI.

Concerning the reliability argument, if a failure occurs and hundreds of kilovolts are applied to a cell designed to withstand only a few, the damage would be catastrophic

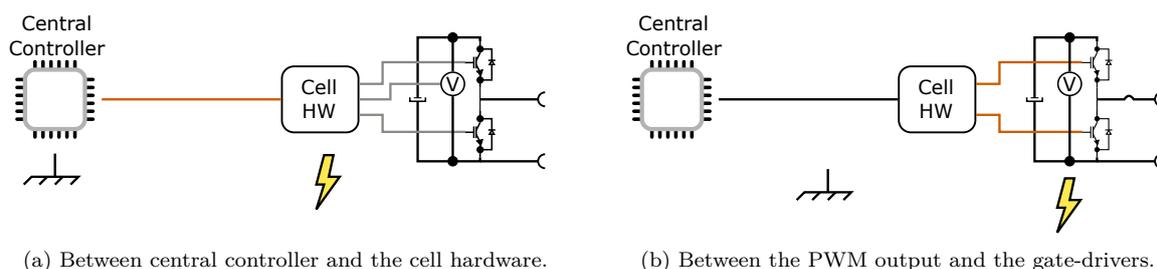


Figure 3.1: Use of Fiber Optics (orange lines).

whether or not the master controller would also suffer a hazard. Thus, isolating the master controller from the cells is irrelevant in this context.

3.1.5. Network Topology

The network topology has a strong impact on the number and length of the communication links, on the network delay, and the tolerance to failures. Among the topologies that can be considered (Fig. 3.2), those with more potential seem to be ring, tree or hybrid tree-ring.

While a tree may lead to more complex and longer cabling but with a shorter delay, the ring implies longer delays with clear benefits on the network cabling, reducing the number of links and their length, thus simplifying maintenance. Moreover, the ring topology is the simplest one that offers two disjoint paths between any two nodes, so it is tolerant to single node or link failures.

Despite the mentioned advantages of the ring topology, the control of an MMC demands very short cycles of less than 200 μs (Chapter 6). As the number of cells in the converter increases, ring networks struggle to achieve acceptable performance. As an example, consider a packet flowing in a ring transporting information from the central controller. The delay in updating all slaves in a ring network with the new data, also known as Minimum Cycle Time (MCT), is expressed by (3.1), where κ is the number of slaves, P is the payload in bytes, P_{overhead} is the number of bytes necessary for a proper packing of the frame, P_{max} is the maximum payload that fits in one frame, T_{byte} is the time to transfer one byte, T_{fw} is the forwarding delay, and $\lceil x \rceil$ returns the nearest integer not smaller than x .

$$MCT_{\text{Ring}} = P \cdot T_{\text{byte}} + \left\lceil \frac{P}{P_{\text{max}}} \right\rceil \cdot P_{\text{overhead}} \cdot T_{\text{byte}} + \kappa \cdot T_{\text{fw}}, \quad (3.1)$$

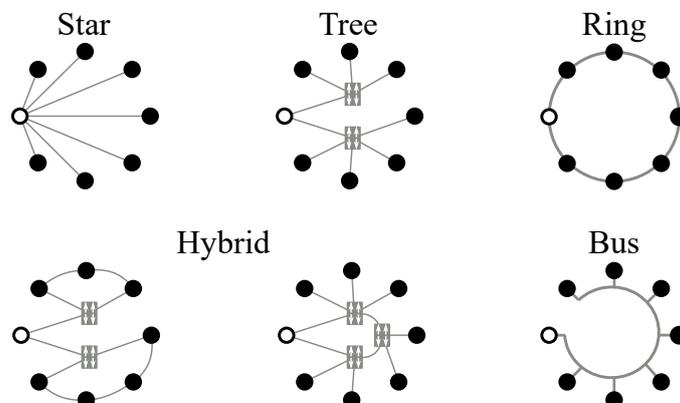


Figure 3.2: Network topologies. The hollow cycle represents the master node and the filled ones the slaves.

On the other hand, the delay in a tree topology (3.2) depends on the packet length and the number of hops (3.3), where $ports$ is how many downwards (i.e., leaves side) ports the switches have, and T_{hop} is the hop forwarding delay. Cut-through switches take forwarding decision immediately after the destination address field has arrived, so the forwarding delay is in the order of microseconds and can be less than 3 μs with Fast Ethernet and less than 1 μs with Gigabit Ethernet [100, 101].

$$MCT_{Tree} = P \cdot T_{byte} + \left\lceil \frac{P}{1500} \right\rceil \cdot 50 \cdot T_{byte} + hops \cdot T_{hop} \quad (3.2)$$

$$hops = \lceil \log_{ports} \kappa \rceil + 1 \quad (3.3)$$

In Figure 3.3, we show the MCT for the ring (for EtherCAT, $P_{max} = 1488$ Bytes, $P_{overhead} = 50$ Bytes, $T = 80$ ns, and $T_{fw} \approx 0.7 \mu s$) and tree (Fast Ethernet) topologies, in the latter case using switches with four and eight downward ports. The performance difference as the number of nodes increases is considerable and, more important, shows how ring topology performance is insufficient for a converter with a large number of cells. This is the reason why designers from ABB [42] and Alstom [70] opted for alternative network topologies (hybrid and tree, respectively).

Despite the difference in MCT, we must still consider other aspects. In what concerns tree topology, cut-through switches propagate errors, rising other questions regarding the coherence of the system, e.g., an error in one link will cause all downwards cells not to receive the master controls in that cycle. Its tolerance to failure is harder to achieve, as a larger number of components are necessary, and their failure affects several cells. The main point, though, is the traffic pattern. If we want to broadcast from the root (master), tree topology is unbeatable regarding performance. The communication in a tree is also very efficient when nodes need to send data to its switch neighbors, but communication

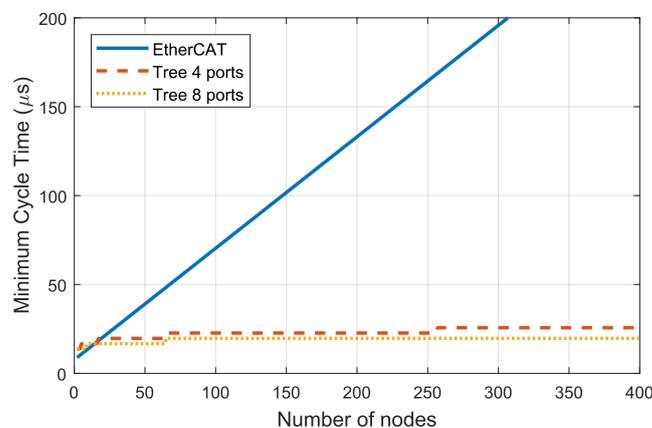


Figure 3.3: Minimum Cycle Time for the ring (EtherCAT) and tree (Fast Ethernet) network topologies, when broadcasting information from the master controller to cells, only.

across switches raises contention problems¹, impacting the end-to-end delay. Therefore, we believe that, as long as the network delay is tolerable by the application, the ring topology is more attractive.

3.1.6. Synchronization Accuracy

The error in the cells clock synchronization in a Modular Multilevel Converter depreciates the harmonic voltage cancellation gained when switching those cells in an interleaved fashion. To understand how it affects the harmonic content, consider the chain of cells represented in Fig. 3.4 as one arm of a Modular Multilevel Converter. When phase-shifted carrier PWM is used to control such converters, the carrier of each cell must be $360^\circ/\kappa$ out of phase with respect to its neighbors, where κ is the number of cells per arm.

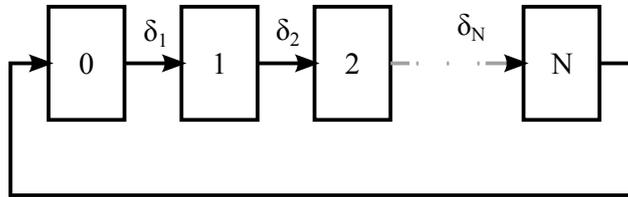


Figure 3.4: Cells and communication link.

The switching function of the k -th cell in the lower and upper arm are represented by s_{kl} and s_{ku} respectively. Assuming that double-edged PWM with natural sampling is used, the harmonic content can be expressed as in (3.4) and (3.5), where m is the modulation index and ω_1 is the fundamental angular frequency [67]. A_{abkl} and A_{abku} are the terms of the harmonic content of the carrier frequency, its harmonics, and the sidebands, as given by (3.6) and (3.7).

$$s_{kl} = \frac{1}{2} + \frac{m}{2} \cdot \cos(\omega_1 t) + \sum_{a=1}^{\infty} \sum_{b=-\infty}^{\infty} A_{abkl} \quad (3.4)$$

$$s_{ku} = \frac{1}{2} - \frac{m}{2} \cdot \cos(\omega_1 t) + \sum_{a=1}^{\infty} \sum_{b=-\infty}^{\infty} A_{abku} \quad (3.5)$$

$$A_{abkl} = \frac{2}{\pi a} J_b \left(\frac{\pi a}{2} m \right) \sin \left[(a+b) \frac{\pi}{2} \right] \cos [(a\omega_c + b\omega_1)t + a\theta_{kl}] \quad (3.6)$$

$$A_{abku} = \frac{2}{\pi a} J_b \left(\frac{\pi a}{2} m \right) \sin \left[(a+b) \frac{\pi}{2} \right] \cos [(a\omega_c + b\omega_1)t + b\pi + a\theta_{ku}] \quad (3.7)$$

The sum of all the switching functions for the lower arm is expressed by (3.8). In

¹Switched networks, such as switched Ethernet, have dedicated links between nodes and switches, so they avoid packet collisions. On the other hand, the switches may need to buffer incoming packets till the destination link is available, what we refer here as contention.

(3.6)-(3.9), J_b is the Bessel function, $\theta_{kl} = \frac{2\pi}{N}k + \alpha + \delta_{kl}$, $\theta_{ku} = \frac{2\pi}{N}k + \beta + \delta_{ku}$, κ is the number of cells in an arm, α and β are the phase displacement between the reference and the carrier, ω_c is the carrier angular frequency, and δ_{kl} and δ_{ku} are the synchronization errors of the k -th cell.

$$s_l^\Sigma = \kappa \cdot \left[\frac{1}{2} + \frac{m}{2} \cdot \cos(\omega_1 t) \right] + \sum_{a=1}^{\infty} \sum_{b=-\infty}^{\infty} \frac{2}{\pi a} J_b \left(\frac{\pi a}{2} m \right) \sin \left[(a+b) \frac{\pi}{2} \right] \cdot \Upsilon \quad (3.8)$$

$$\Upsilon = \sum_{k=1}^N \cos \left[(a\omega_c + b\omega_1) t + a \left(\frac{2\pi}{N} k + \alpha + \delta_k \right) \right] \quad (3.9)$$

We are interested in the effect of the synchronization errors δ_k into the voltage harmonic content, but first, it is necessary to understand how the physical implementation of the Physical Layer (PHY) influences the errors in each node.

3.1.6.1. Physical Layer Synchronization

Modern communications with high-speed² links, such as Ethernet, have synchronous nature. The transmitter embeds the clock signal together with the data, so the receiver can recover it and make it available for the destination node logic. In multi-port nodes, the PHY of the incoming port handles the data to one or more ports to transmit. As the received clock of the incoming port may be different than the one employed for transmission in another port, each node has potentially different clock domains that must be crossed appropriately to guarantee reliable operation of the network.

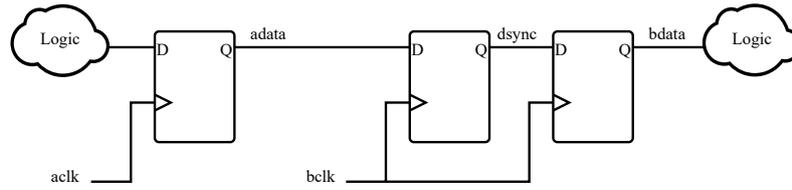
A typical strategy for single-bit³ Clock Domain Crossing (CDC) is double flopping (Fig. 3.5a) [102]. In this circuit, the output of the first flip-flop might suffer metastability, but it has one clock period to settle before the second flip-flop registers it and the remaining logic at the *blk* domain can use it.

When an implementation having two asynchronous clock domains with the same nominal frequency uses double flopping for CDC, the signal faces a variable delay of one to two clock cycles depending on the phase shift between the clocks. When the rising edge of the source clock (*ack*) happens just before the rising edge of the capture clock (*blk*), the delay will be of one clock period (Fig. 3.5b). When the rising edge of the capture clock happens just before the one of the source clock, the capture misses the data transition, and the total delay is two clock periods (Fig. 3.5c).

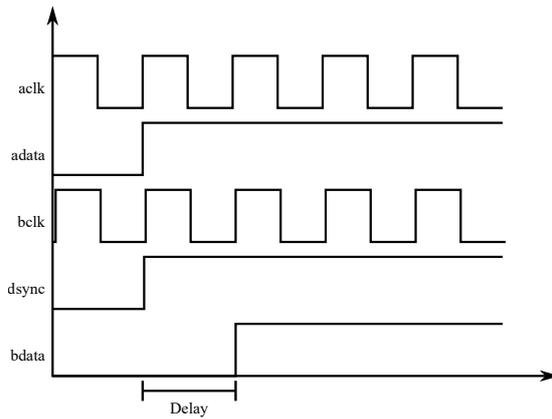
The consequence of this phenomenon is that each node has a variable delay to forward the incoming packet if the PHY transmitter uses a clock that is not phase-locked to the recovered clock of the receiver. To make this clear, consider the ring network with three

²High speed is a relative concept, but here we consider data rates equal to or higher than 100 Mbit/s.

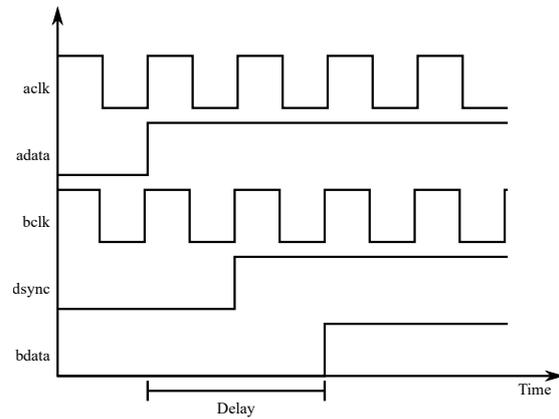
³The CDC inside the node involves multiple bits, but for the sake of explaining the consequence, a single bit CDC is sufficient.



(a) Double flopping.



(b) One clock period delay.



(c) Two clock periods delay.

Figure 3.5: Variable delay due to Clock Domain Crossing.

nodes depicted in Fig. 3.6. The PHY integrated circuits have a clock input pin that serves as the reference for the internal and the transmitter logics. A common design practice is to use the same crystal as a reference for both PHYs in a node, as in Fig. 3.6a. In such case, both transmitters on a node are in the same clock domain, but as the receiver locks to the clock of the neighboring node, CDC is necessary. Certain PHY implementations (e.g., PHYs with RGMII interface) handle the CDC internally. When it is not the case (e.g., PHYs with MII), the node logic, for example implemented in the Field Programmable Gate Array (FPGA), must take care of proper crossing.

On the other hand, some PHYs can output the recovered clock to a dedicated pin. Then, if this signal is forwarded to the PHY at the other port and employed as the reference for the transmitter, the network will have a single clock domain (Fig. 3.6b). With all the PHYs operating in sync, we avoid CDC as well as the variable delay associated with it. In Fast Ethernet mode, where the wire link speed is 125 Mbit/s and the Media Independent Interface (MII) outputs data with 25 Mbit/s, a variable error will still be present, but it remains constant while the link remains established [103]. Note that the implementation shown in Fig. 3.6b can only synchronize the PHYs in one direction because the first PHY in each node, i.e., the one at the left side of the node, has a local clock reference that it will use for transmitting data.

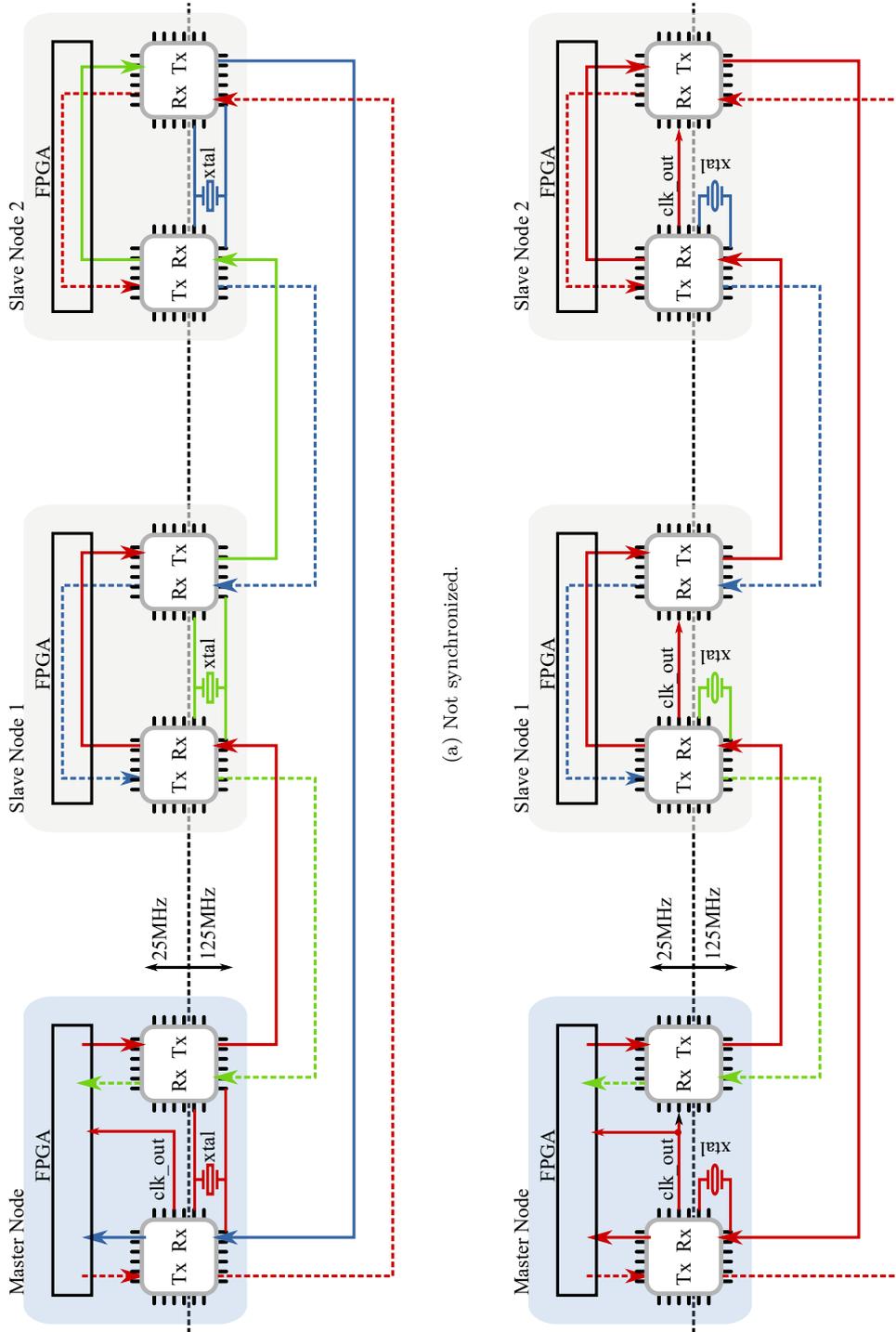


Figure 3.6: Physical Layers synchronization. The continuous lines represent one direction of the ring network and the dashed lines the other one.

3.1.6.2. AC Terminal Voltage Distortion

Once we discussed the synchronization of the physical layers, here we assess how it affects the harmonic voltage content of the output waveform.

When δ_k is constant, the cosine terms of (3.9) sum to zero, unless a is $[N, 2N, 3N\dots]$, in which case Υ is equal to N . To consider the synchronization error, we model δ_k in two ways. In the first model, the error in the k -th cell depends on the error of the previous cells (3.10), typical of implementations that need a clock domain crossing inside the node that causes packet retransmission jitter and an accumulation effect in the synchronization accuracy (indicated in [48]). In the second model, a simple uniform distribution (3.11) represents the case when the slaves share the same clock signal [104] and have a fixed forwarding delay. In either model, the clock reference (cell 0) has zero error (3.12).

In (3.10) and (3.11), the *uniform()* function returns a random sample following a uniform distribution, where the first argument is the mean value and the second the maximum delta from the mean value, i.e., the function outputs a number within the interval $[mean - \sigma, mean + \sigma]$. In both cases, the synchronization error has mean zero, because, otherwise, we would be able to measure and compensate it.

$$\delta_k = \delta_{k-1} + \text{uniform}(0, \sigma) \quad (3.10)$$

$$\delta_k = \text{uniform}(0, \sigma) \quad (3.11)$$

$$\delta_0 = 0 \quad (3.12)$$

After a synchronization process, e.g., using the Precision Time Protocol [105], the system will have a set of $[\delta_1.. \delta_k]$, and we can calculate Υ . As the clocks drift, a new synchronization is necessary; the system will have a new set of δ s and a new Υ . We simulated two thousand synchronizations, following (3.10) and (3.11), for three different standard deviations, 1%, 2%, and 5% of the switching period, in a network with 5 and 20 cells. Fig. 3.7 shows the histogram of Υ .

We can assess the influence of the synchronization error in the current harmonic content as follows: with zero synchronization error, i.e., $\delta_k = 0$, only harmonic content around $N\omega_c$ will appear. In this case, Υ is equal to N and, as the filter inductor impedance increases with the frequency, the combined effect in the load current is proportional to $N/N\omega_c = \omega_c^{-1}$. On the other hand, when synchronization errors are present, all the multiples of first switching harmonics (ω_c) and sidebands appear. These are the most representative components, because of the inductor lower impedance, but as long as $\Upsilon/\omega_c < \omega_c^{-1}$, i.e., $\Upsilon < 1$, we can expect a total harmonic content of the load current in the same order of magnitude as without synchronization error.

The results showed in Fig. 3.7 point to different limits for an acceptable error according

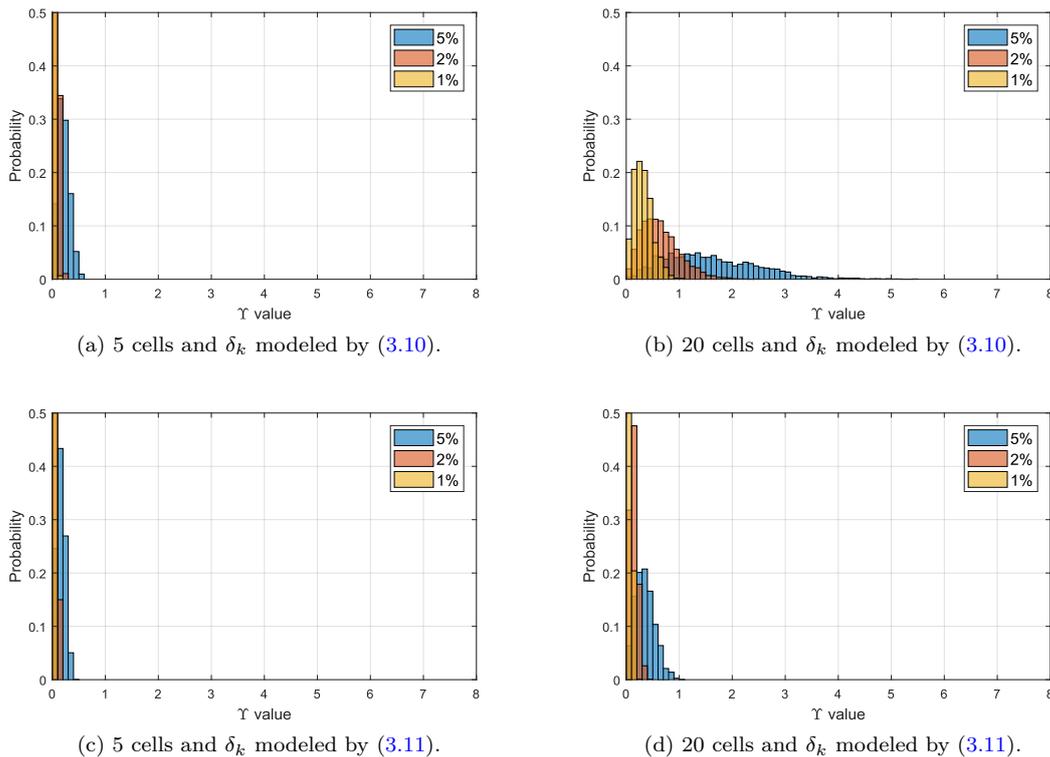


Figure 3.7: Histogram of Υ for the first harmonic and different σ 's.

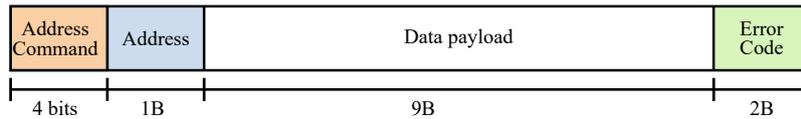
to the number of cells of the converter and their switching frequency. With five cells per arm and the accumulation effect of (3.10), an error of 5% is acceptable as Υ stays less than one, while with 20 cells, the synchronization has to be better than 1% to keep Υ lower than 1. On the other hand, without the accumulation effect in the synchronization accuracy, even an error of 5% leads to Υ lower than 1 in a converter with 20 cells. Thus, the exact use case has to be considered, as a fixed requirement (e.g., 20 ns or 1%) might lead to over tight requirements.

3.2. Existing Solutions

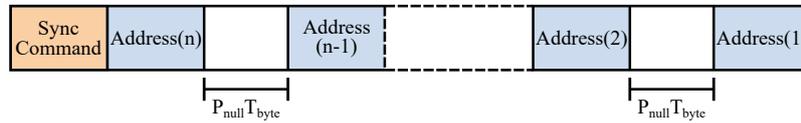
To the best of our knowledge, the first works to propose high-speed digital communication⁴ as a solution for the control of power electronics converters were developed under the Power Electronics Building Blocks program, sponsored by the Office of Naval Research, USA. This work resulted in a dedicated protocol called Power Electronics System Network (PESnet), first designed by Milosavljevic [47] and later reviewed by Francis [108].

Several authors embraced [EtherCAT](#) as an alternative to [PESnet](#). The firsts to look into it as an option were Toh and Norum, who compared [EtherCAT](#) with Profinet IRT and [PESnet](#), and found it to have better performance [37], lower synchronization error/jitter

⁴Previous to the works cited, other authors employed low-speed communications to control power electronics converters [106,107].



(a) Data frame.



(b) Synchronization frame.

Figure 3.8: PESnet data frames.

[38, 41], and fault tolerance [39]. Later, researchers from Aalborg University implemented a partly distributed control in an MMC prototype to reduce the data shared globally and demonstrated single fault tolerance when using EtherCAT [43–45, 85].

3.2.1. Power Electronics System Network

Milosavljevic [47] proposed a protocol based on MACRO (Motion and Control Ring Optical) and FDDI (Fiber Distribution Data interface)⁵ intended to exchange data between the central controller and the power stages. It transmits data over Plastic Optical Fiber in a ring topology, with a frequency of 125 MHz and 4b/5b encoding. It is a master/slave protocol, where the master sends every cycle 13-Bytes data frames (Fig. 3.8a) addressed to one of the 32 slaves in the network. When the slave receives a frame, it passes it along, unless the address field matches its own. In this case, the slave replaces the data with feedback information and forwards the packet to the next hop.

To synchronize the nodes, the master sends a long frame, starting with a synchronization identifier and followed by the slaves' addresses in reverse order (Fig. 3.8b), i.e., the address of the last slave at the first position and of the first slave at the last position [47]. Between two addresses, the master inserts empty words, such that the slaves receive their address simultaneously and update the local clock. The number of empty words, P_{null} , is an approximation to the measured propagation delay between two nodes, leading to a maximum error of 40 ns [49]. However, Celanovic [46] measured a synchronization jitter of 80 μ s in a network with three nodes.

The PESnet protocol establishes a direct link between the network and the PWM frequency because the master sends to each slave values corresponding to the turn on and turn off transition instant of each switch as a fraction of the time between synchronizations. Fig. 3.9 illustrates such a division when the pulse generation employs five bits. The slaves use the synchronization moment to update the turn on and turn off time with the last received value.

⁵FDDI was the basis for the specification of 100BASE-FX Ethernet.

As a consequence, the Minimum Cycle Time corresponds to the minimum period for the PWM generation and is equal to (3.13), where κ is the number of slaves, P and $P_{overhead}$ are the data payload (9 Bytes) and the protocol overhead (4 Bytes), T_{byte} is the time to transmit one byte (80 μs), T_{fw} is the forwarding delay (~ 460 ns [46]), and T_{sync} is the time to send the synchronization frame (3.14). Fig. 3.10 shows PESnet Minimum Cycle Time for different numbers of empty words and nodes.

$$MCT_{\text{PESnet}} = \kappa \cdot (P + P_{overhead}) \cdot t_{\text{byte}} + T_{\text{sync}} + \kappa \cdot T_{\text{fw}}, \quad (3.13)$$

$$T_{\text{sync}} = (\kappa + 1)T_{\text{byte}} + (\kappa - 1) \cdot P_{\text{null}}T_{\text{byte}}, \quad (3.14)$$

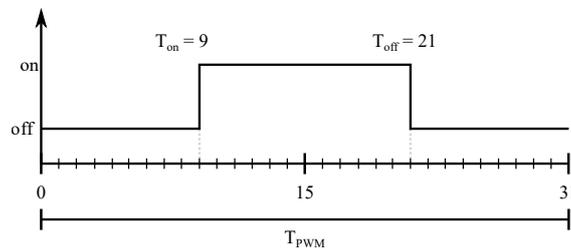


Figure 3.9: PWM generation using a time-division of the period.

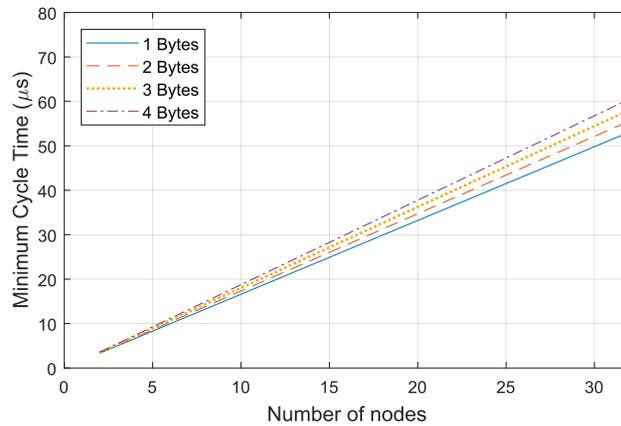


Figure 3.10: PESNet Minimum Cycle Time for various numbers of empty words (P_{null}).

PESnet main drawbacks are the high protocol overhead that leads to low efficiency and its rigid structure. It has a fixed packet size, fixed topology, fixed data rate, and the impossibility of communication between slaves. Thus it is difficult to improve performance using new control strategies. Also important, the maximum number of slaves is only 32, and the master cannot broadcast messages.

3.2.2. EtherCAT

The company Beckhoff introduced [EtherCAT](#) (Ethernet for Control Automation Technology) in 2003. Its innovative concepts, the on-the-fly processing of the frame and the summation frames are the main reason for its success.

The on-the-fly processing refers to the ability of the EtherCAT Slave Controller (ESC) to read and write data to the Ethernet packet while it passes through the controller. As the slave does not hold the entire packet for processing, the packet suffers only a short delay of a few bytes (typically 4-16 Bytes [109]) plus the delay of the [PHYs](#) as it passes through a node.

A single frame can address several slaves, thus the name summation frame. [EtherCAT](#) uses the standard Ethernet frame format, where the EtherType 0x88A4 indicates the [EtherCAT](#) protocol. The [EtherCAT](#) frame has a one Byte header and one or more [EtherCAT](#) datagrams. Each datagram has its header, payload, and a working counter that counts the number of slaves correctly addressed by the datagram [110].

[EtherCAT](#) has two addressing possibilities: Device Addressing and Logical Addressing. In Device Addressing mode, each datagram conveys data to/from a single slave, a rather inefficient method for a large number of slaves due to the 12 Bytes datagram overhead. In Logical Addressing mode, on the other hand, each slave maps the (global) logical process data into its local address space using Fieldbus Memory Management Units. Thus, one local address range can span several slaves, significantly reducing the protocol overhead.

Though [EtherCAT](#) supports a flexible connection topology, in fact it always operates as a logical line (or ring, if the last slave has a link with the master node). The reason for it is that the on-the-fly processing forces the frame to pass all the slaves in sequence. The processing occurs only in one direction; thus a slave can send data only downstream to other slaves. The exchange of data upstream must be through the master, adding a delay of one cycle.

[EtherCAT](#) Minimum Cycle Time corresponds to the format mentioned for ring topology in the previous section (3.1) and after replacing P_{overhead} with 50 Bytes and P_{max} with 1488 Bytes we obtain (3.15), where $T_{\text{byte}} = 80$ ns (it is limited to Fast Ethernet), and $T_{\text{fw}} \approx 0.7$ μ s. In (3.15), the first term corresponds to the time to transmit the payload; the second term to the overhead due to packet headers, interframe gap, and frame check sequence; the third to the sum of the nodes forwarding delays.

$$MCT_{\text{EtherCAT}} = P \cdot T_{\text{byte}} + \left\lceil \frac{P}{1488} \right\rceil \cdot 50 \cdot T_{\text{byte}} + \kappa \cdot T_{\text{fw}} \quad (3.15)$$

Fig. 3.11 shows the [EtherCAT](#) Minimum Cycle Time for payloads of 2, 4, 8, and 16 Bytes per slave, where we can see that [EtherCAT](#) also has problems with keeping the cycle time below 200 μ s as the number of nodes grows.

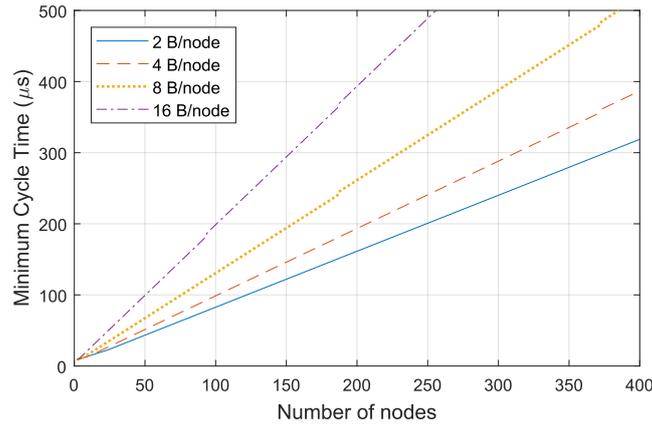


Figure 3.11: EtherCAT Minimum Cycle Time with a different number of bytes per node.

EtherCAT specifies a synchronization protocol named Distributed Clock. It achieves and maintains the same system time in the network⁶ with three mechanisms [111]: a) Propagation delay measurement, b) Offset compensation, and c) Drift compensation. In the first mechanism, the master sends periodically a synchronization datagram in which each slave inserts a time-stamp with its local clock. As the master knows the network configuration, it uses this data to measure the propagation delay of each segment. In the second mechanism, the master uses the time-stamp data to remove the offsets of the local free-running clocks, which started counting at different instants with different initial values. Note that both mechanisms are similar to IEEE Std 1588 [105]. In the third mechanism, a time control loop adjusts the local clock frequencies to reduce the drift caused by the different clock references (crystal or oscillator).

EtherCAT is widely deployed in factory automation networks; hence many master stacks exist. However, only a few reach cycle times lower than 100 μs [109] and none support cycles shorter than 50 μs . Recently researchers broke this barrier, first using a hardware accelerator in a Linux based implementation [112], then using zero-copy buffers, memory pre-allocation, and mapping of application variables directly onto **EtherCAT** telegrams [109]. In 2015, Cottet et al. [42] achieved cycles of 11 μs , but with a custom designed Master Stack.

Another factor influencing **EtherCAT** latency, though not the cycle time, is the low bandwidth of the Process Data Interface (PDI). The PDI is responsible for the connection between the ESC and the slave application processor. Several options are available for the PDI depending on the ESC implementation [113]: SPI, synchronous or asynchronous parallel bus, and on-chip bus (AXI for Xilinx Intellectual Property core). An SPI interface, with an address space of two Bytes, would be able to read bytes sequentially⁷ with a latency given by (3.16), where *MemoryRegions* is the number of accessed memory regions

⁶The Distributed Clock strategy excludes the master node.

⁷Typically SPI peripherals present in microcontrollers cannot read an arbitrary amount of data on each transaction. The consequence is that the effective read time would be considerably slower than that, not only due to addressing overhead, but also because of software intervention.

and $T_{byte} > 400$ ns ($f_{clk} < 20$ MHz) for the ET1100 ASIC. The read time is, therefore, larger than 400 ns/Byte, corresponding to a bandwidth of 20 Mbit/s.

$$t_{read} = 160 \text{ ns} + \text{MemoryRegions} \cdot (21 \text{ ns} + 2 \cdot T_{byte} + 240 \text{ ns}) + P \cdot T_{byte}, \quad (3.16)$$

The fastest available interface to exchange data between ICs is the synchronous 16-bit parallel type [113], which has a reading latency given by (3.17), where $T_{clk} > 25$ ns. The minimum read time is 190 ns/Byte, corresponding to a bandwidth of 41.6 Mbit/s.

$$t_{read} = P \cdot \frac{(315 \text{ ns} + 3 \cdot T_{clk})}{2} \quad (3.17)$$

If the ESC is implemented as an Intellectual Property block inside an FPGA, then higher performance is possible, though not as fast as the internal bus would allow [113]. The ESC has an internal limitation of 200 Mbit/s that makes the minimum read time slightly above 40 ns/Byte. Therefore, unless each slave reads a limited amount of bytes, the PDI becomes an important bottleneck.

Carstensen et al. [104] developed the protocol *SyCCo bus* specifically for modular converters. It uses EtherCAT's summation frame and on-the-fly processing, but brings a few innovations: it synchronizes the PHYs, as described before, to improve the synchronization accuracy. Second, each slave datagram has an error check code, hence the slave processes the data before the arrival of the Frame Check Sequence.

The SyCCo bus first reported implementation had a long forward delay of 2.1 μ s [104], but following improvements could bring it to the same level of EtherCAT, around 0.7 μ s [114]. Tu and Lukic [49] compared SyCCo bus with PESnet and found it to allow 28% higher switching frequency in a 30 cells converter.

To summarize, EtherCAT is a popular and mature protocol with several off-the-shelf components. It offers high performance in ring networks but falls short of being able to control MMCs with a large number of cells. As an industrial protocol, the user has limited flexibility to adapt it for a specific control strategy that could result in better performance. For example, the link speed is 100 Mbit/s, and higher capacity will only be available when the EtherCAT Technology Group decides to do so.

3.3. Future Perspectives

EtherCAT and PESnet fail to reach the MCTs lower than 200 μ s that are necessary for satisfactory control performance (Chapter 6) in networks with more than 200 nodes (Fig. 3.10 and 3.11); hence it is reasonable to consider communication technologies with higher link capacities as means to overcome this limitation. However, such communication

technologies are constrained to short distance when using electrical media or to Glass Optical Fiber (GOF). In industrial systems, Plastic Optical Fiber (POF) is preferred to GOF, because of [115]:

- Simpler installation, thanks to a much larger core (1 mm for POF, 50 μm for multi-mode glass fibers);
- Use of visible light rather than infrared, resulting in simpler “visual” check of the integrity of the connections;
- Higher mechanical robustness and tolerance to bending and dusty environments.
- Overall, glass fiber requires skilled technicians for the installation, while for POF do-it-yourself approach for the final user can be envisioned.

Unfortunately, manufacturers have struggled to push POF data rates beyond the gigabit. Only a few PHY (e.g., KD1011 [116]) and optical transceivers (e.g., Broadcom AFBR-59F3Z) are available.

Next, we explain two communication technologies that have a higher link capacity and discuss their ability to reduce the Minimum Cycle Times of the network: Gigabit Ethernet and SERDES.

Today, it is hard to imagine designs for Modular Multilevel Converters based solely in those technologies, because of higher cost, lower robustness, and the already pointed drawbacks of glass optical fiber. Nevertheless, such technologies could be used to speed up MMC control networks in the higher levels of a tree or hybrid networks, i.e., in the links between the central controller and the switches and among switches. Moreover, in such nodes, as explained before, electrical media is an option.

Gigabit Ethernet is the natural (and overdue) path for Real-Time Ethernet protocols to increase data rate and improve performance. Jarsperneit et al. [117] and Prytz [118] discussed more than a decade ago which benefits Gigabit Ethernet could bring to EtherCAT and PROFINET. The main gain is in the reduction of packet transmission time by a factor of ten. Another benefit is the possibility of using Jumbo frames with up to 9000 Bytes [118], thus reducing protocol overhead. Both works forecast that Gigabit Ethernet would result in lower forwarding delays in the slaves, but today the difference in Gigabit PHYs operating in Gigabit Ethernet (336 ns) or Fast Ethernet (312 ns) is marginal [119]. Still, some gain is possible due to the higher clock frequency (125 MHz against 25 MHz) inside the node logic.

Given these points, in a line/ring topology, Gigabit Ethernet reduces the contribution of the packet transmission time to the Minimum Cycle Time, but the dominant factor becomes the forwarding delays of each node, where the higher capacity has almost no effect. On the other hand, the increase in performance in tree networks is more relevant,

widening even more the gap between ring and tree topologies. As already mentioned, the forwarding delay in cut-through switches falls below $1\ \mu\text{s}$, what, together with the 10-times lower packet transmission delay, shortens the cycle times. As an example, the **MCT** of a tree network (3.2) with 400 nodes, six hops between the root and the leaf nodes, and a payload size of 1000 Bytes, would be $11\ \mu\text{s}$ with Gigabit Ethernet compared to $102\ \mu\text{s}$ with Fast Ethernet.

One may wonder why the migration to the faster data rate in the factory floor has been so slow. The literature has little on this topic; thus we speculate some reasons:

- The industrial environment has harsher conditions, so the number of components and manufacturers providing ICs that can tolerate them is smaller. As a consequence, the availability of such components is lower, and the price is higher.
- The reduced number of **PHYs** and transceivers that allow the use of **POF**, as stated before.
- The faster data rate leads to faster logic and the necessity of more careful hardware design, again impacting costs.
- The previous point may cause manufacturers to be wary of migrating established protocols to Gigabit.
- Lastly, the number of applications demanding such high performances are a small fraction of the total, so probably the market is not yet pushing manufacturers to make the transition.

SERDES stands for multi-gigabit Serializer/Deserializer and is also known as Multi-Gigabit Transceiver (MGT). Xilinx manufactures MGTs that support wire speeds up to 12.5 Gbps for a single differential pair [120]. MGTs are essential building blocks of several interface standards, such as Fibre Channel, Infiniband, Serial RapidIO, Gigabit Ethernet, and 10 Gigabit Ethernet [121].

SERDES employs many technologies to achieve high speeds, such as differential signaling, multi-bit signal encoding, clock correction, channel bonding, pre-emphasis and de-emphasis, and line equalization [120, 121].

Optical media has prevalence over wire media in this technology, where Small Form-factor Pluggable transceivers (SFP) or Quad SFP (QSFP) make the conversion from wire to single mode or multi-mode glass optical fiber. As already pointed, this is a limiting factor; so is the high cost of the transceivers.

3.4. Network Induced Latency

The adoption of a real-time protocol translates into a deterministic delay and **MCT**. When the controller outputs individual references to the cells or a single one per arm, the **MCT** defines the minimum possible actuation delay, because the cells must simultaneously apply the new references to avoid a variable loop delay.

Often, authors take the **MCT**, the sampling-to-actuation delay, and the control sampling frequency as similar concepts, but it must not be so. Consider Ethernet frames traveling over a network when the master node sends a new packet just after the inter-frame gap has elapsed (Fig. 3.12). The **MCT** includes the propagation delay between the sender node and the last node to receive the frame, including the forwarding delay of all cells along the path; hence the **MCT** is longer than the time between the transmission of two consecutive packets to a node. As the central controller can only send packets with at least this minimum delay, it corresponds to the minimum sampling period h of the control cycle.

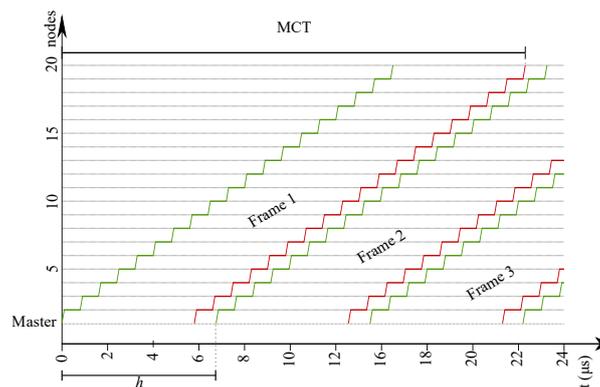


Figure 3.12: Ethernet frames traveling in a network. The green and red lines indicate the start and end of a frame, respectively.

EtherCAT MCT of a frame with minimum payload is equal to $10.2 \mu\text{s}$, $13.7 \mu\text{s}$, $41.7 \mu\text{s}$ or $76.7 \mu\text{s}$ for a network with 5, 10, 50 or 100 nodes, respectively. However, independently of the network size and the forwarding delay (see Fig. 3.12), the minimum sampling period is $6.7 \mu\text{s}$, as it depends only on the minimum payload size and the interframe gap that are 72 and 12 Bytes long, respectively [122].

In practice, the tasks execution time and scheduling prevent the controller of reaching such low sampling periods. For example, the central unit has to convert the analog signals and calculate the outputs of the control algorithms; the protocol stack has to prepare the data and handle them to the Media Access Control (MAC); then, the **MAC** has to command the physical layer to send the packet. In our experimental set-up, for example, measurement conversion and control calculations take $27 \mu\text{s}$. An optimized **EtherCAT** master stack can transmit small frames (~ 72 Bytes long) within $11 \mu\text{s}$ [42], hence, in this particular case, the minimum sampling period would be $38 \mu\text{s}$, unless the controller can

run some of these tasks simultaneously.

The sampling to actuation delay, ζ , includes all the delays since the data is sampled, processed by the controller, and applied to the cells, including all associated communication latencies, resulting in a value that can be longer than the sampling interval. Thus, we can define the loop delay n as the number of samples taken during the sampling-to-actuation delay (3.18). This allows quickly associating an actuation value to the sample that was used to generate it.

$$n = \left\lceil \frac{\zeta}{h} \right\rceil, \quad (3.18)$$

As an illustration, the diagram of Fig. 3.13 shows a network with the control running at two different rates. In the first case (a), the network introduces a delay of one sample, but in the second case (b), it introduces a delay of two samples due to the shorter sampling period. In an implementation that has a sampling-to-actuation delay just below 100 μs , the loop delay is one sample, just like any discrete controlled system, as long as the sampling period is more than 100 μs . The communication latency would represent a delay of one or two samples, if the sampling period is in the range [50 μs , 100 μs) or [33 μs , 50 μs), respectively, and the sampling-to-actuation delay would be two or three samples.

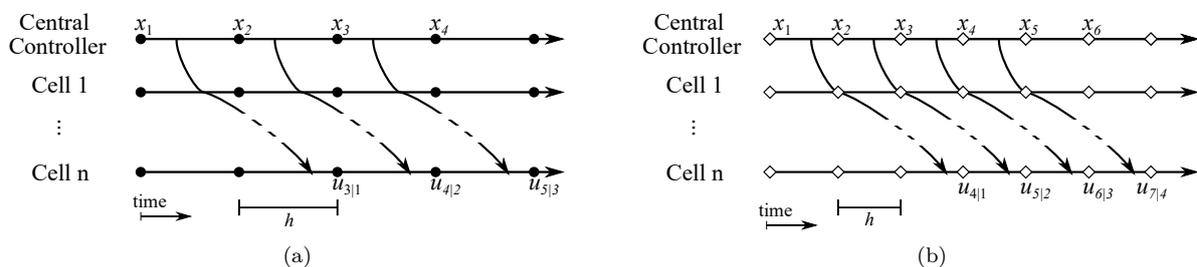


Figure 3.13: The network introduces additional delay to the actuation. As the sampling period reduces from (a) to (b), the loop delay in number of samples increases. \mathbf{x}_k is the state vector in instant k , and $\mathbf{u}_{k|k-n}$ is the plant input vector in instant k calculated in $k - n$.

3.5. Conclusions

Digital communication is an exciting alternative for improving scalability, and facilitating implementation and maintenance of power electronic converters. In this chapter, we discussed some aspects of the communication network and explained the characteristics and limitations of the two leading solutions proposed to date, namely [PESnet](#) and [EtherCAT](#). Additionally, we described how the communication influences the loop delay, and we discussed the differences between the network Minimum Cycle Time, the control period, and the actuation delay.

In the network selection, end-to-end latency is a crucial aspect to define whether a given communication strategy is suitable or not for controlling the converter; thus link capacity and data payload are only factors that influence it. We showed that the impact of synchronization grows with the number of cells and that the choice between ring and tree topology needs to cover multiple dimensions since none dominates the other. As the number of nodes increases, ring networks have difficulty to keep the Minimum Cycle Time below the required values.

The use of fiber optics is mandatory in certain circumstances, while in others electrical media is also adequate. Note that electrical media can facilitate the migration to higher link speeds. Today, several manufacturers have electrical media [PHYs](#) for industrial use that support Gigabit Ethernet (such as TI DP83867), and they are readily available and have a low cost.

Regarding [EtherCAT](#) and [PESnet](#), both use ring topology, hence they have problems to cope with large networks. [PESnet](#), in particular, has a limited address range and is too rigid for us to consider it as a suitable solution for large [MMCs](#). [EtherCAT](#) seems a better alternative, but for using it in a converter with a large number of cells, we would need to split the internal network into several smaller ones (e.g., one per phase or arm). In this case, several master stacks would need to run in parallel, increasing software and hardware complexity at the central controller. Another drawback is that [EtherCAT](#) has not yet evolved to Gigabit Ethernet and we are not sure if and when it will.

Chapter 4

Network and Control Co-design

In this chapter, we propose two internal control networks for Modular Multilevel Converter (MMC) that seek to reduce the network Minimum Cycle Time (MCT) to the lowest value possible with the chosen link speed and network topology.

The communication technology of choice is Ethernet, because of its high link capacity, availability of components, reliability, pervasiveness, and low cost. As usual in this application domain, we adopt the ring topology (Fig. 4.1a) because it is the simplest one to offer two disjoint paths between any two nodes; in other words, it is the simplest to offer single fault tolerance at the link level. Nevertheless, the proposals are also valid to hybrid networks that have the first level of hierarchy, i.e., the central controller side, built as a ring (Fig. 4.1b).

An example of this topology is a converter where one node controls several cells, in contrast to the one node per cell of the ring topology. In this case, the final link to the cells could use a different (and possibly slower) digital communication technology, or even on/off commands to the gate-drivers of the power switches. The main drawback of the hybrid topology is that a defective node causes several cells to come out-of-operation; therefore a design using this approach must consider not only the cost reduction of having less control hardware and the gains in end-to-end latency of having fewer nodes but also how it affects reliability.

The diagram of Fig. 4.2 shows the control topology where a central controller acts as the master and communicates with all the cells via the network.

Up to the present time, the literature refers to two main strategies to the networked control of MMCs. In the first strategy, the central controller sends commands for the power switches over the network employing the time-division principle, as in the PESnet protocol (Fig. 3.9) [47] or, in some of the works, using EtherCAT [40]. The nodes send back their capacitor voltage that will serve as input to the balancing algorithm in the central controller. This approach leads to a heavy packet payload, even when the nodes reuse the fields with command data to send the voltage information [104]. Typically,

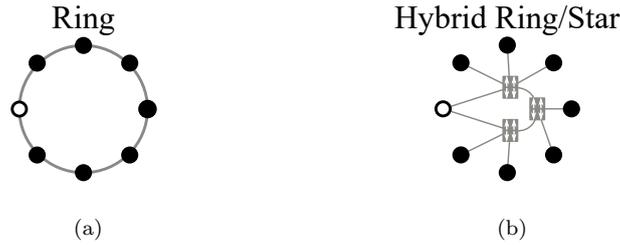


Figure 4.1: The protocols proposed are for networks with (a) ring or (b) hybrid ring/star topology.

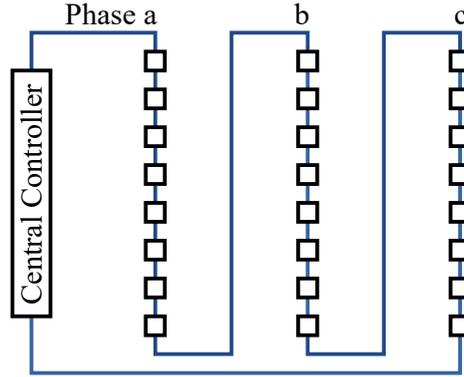


Figure 4.2: MMC controlled through a ring network. The small boxes represent the cells of the three phases.

each slave needs at least four Bytes to operate [41], for example, two Bytes for capacitor voltage and two Bytes for status feedback [45]; hence a ring network with 100 Mbit/s link capacity, two hundred nodes, and a forwarding delay of $0.7 \mu\text{s}$ will need $208 \mu\text{s}$ to update all slaves with new data.

In the second strategy, the proposal is to employ a partly decentralized control strategy, characterized by a central controller that has enough information to run the MMC outer control-loops, e.g., the load current and the DC link voltage loops. The central controller, though, lacks the necessary data to control the system alone; therefore the slaves are co-responsible for stabilizing the converter.

A partly decentralized control can reduce the payload of the packet in this application domain considerably, mainly because it avoids the transfer of the capacitor voltages to the central controller. It achieves this by adopting the Phase-Shifted Carrier (PSC) Pulse Width Modulation (PWM) and the control-loop balancing [7]. Though both strategies combined reduce the packet payload, because they decentralize the balancing to the cells and reduce the setpoint to one per arm, they restrict the design to a single modulation and balancing strategy.

The first protocol proposed, the Time-Triggered Ring (TTRing), targets the partly decentralized controls described in [8, 86] and was thought such that it minimizes the Minimum Cycle Time. If we analyze the ring network MCT (3.1), we see that the payload size and the forwarding delay are the two factors influencing it the most for a given number

of nodes and link capacity. As the partly centralized control reduces the payload, the forwarding delay becomes then the dominant factor. Therefore, to minimize the MCT we reduce the forwarding delay to a minimum.

The second protocol we propose, Distributed Sorting Network (DiSortNet), seeks to remove the modulation type limitation and to allow the adoption of popular sorting strategies and their variations. As we will see, DiSortNet reaches an MCT similar to the TTRing by the adoption of three innovative strategies: the dual insertion sorting, the decentralized identification of the cells with minimum and maximum voltage, and the reduction of the data necessary to command the power switches.

Both protocols need that the nodes share a common time base, but we will not cover synchronization strategies in this text. Essentially, synchronization schemes based on the IEEE 1588 Precision Time Protocol [105], where the master node is also the *grandmaster clock*, can reach the necessary synchronization accuracy with a carefully designed hardware. The IEEE 1588 principle is as follows: during start-up, or when a broken link is re-established, the master and slave nodes exchange the messages *Sync*, *Delay_Req*, and *Delay_Resp* to measure the network delays. After this initial process, the master time-stamps the packets, and the slave nodes keep the local clock in sync using this information. Note that the network load is stable during normal operation.

After the presentation and comparison of the protocols, we address the quick broadcast of fault information, which is crucial for power electronics converters to tolerate faults. Finally, we discuss the relevance of an accurate simulation of networked Modular Multi-level Converters and propose a co-simulation strategy that has a reduced runtime when compared with traditional methods.

4.1. TTRing

We designed TTRing initially for the partly decentralized strategy proposed in [86], in which the authors remove the necessity of sending capacitor voltage measurements to the central controller when using a closed-loop balancing strategy. They propose that each cell can use the mean value of the capacitor voltages of the neighboring cells as the setpoint for the balancing loop. Only one cell receives a setpoint from the central controller, but with their strategy, the other cells end up following this reference.

The TTRing protocol is time-triggered and has two different phases, one dedicated to the global control, and the other to the local, decentralized control. In the first phase, named *Cycle*, the master node sends a set of references and measurements to the slaves (Fig. 4.3a). In this phase, the slaves only read the master packet and forward it. In the second phase, named *Interslave*, each slave shares information with its neighbors (Fig. 4.3b). Depending on the nature of the master, it may act as a slave in this second phase or just forward the packet, like a transparent node. Note that during this phase,

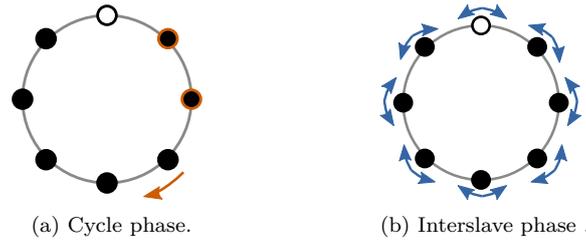


Figure 4.3: Representation of TTRing phases.

all transmissions occur in parallel in the full-duplex links taking full advantage of the network bandwidth.

Due to limited precision of the global clock, a mismatch between the local clocks is still present. A *Guard Window* is necessary to accommodate it and guarantee that every node is in the expected phase when a packet arrives. The *Guard Window* must be the minimum necessary to provide correct operation of the network since it represents wasted time.

We modeled the protocol using OMNeT++ and included several characteristics, such as channel occupation and propagation time, forwarding delay, clock offset and drift, synchronization (except the initial delay measurement), bit errors and loss of packets. Fig. 4.4 presents the result of a network with 43 nodes during two network periods. In this figure, the red dots indicate when the node started (or finished, in the app layer) to receive a packet and the green dots indicate a node internal event, like a change of phase and start of transmission. The shades of blue represent the occupation of the channel between two nodes, and its length is influenced by the link speed and length, and packet payload size.

The simulation result shows the two phases of the protocol: in the first phase, *Cycle*, starting when $t = 100 \mu\text{s}$, the master node sends a packet that rotates through all nodes and comes back to the sender. Note how the transmission coincides in several nodes. In the second phase, *Interslave*, just before $t = 200 \mu\text{s}$, all nodes transmit at the same time, and the network has full bandwidth utilization. In the second period shown, the twenty second node loses the *Cycle* message, and the downstream nodes do not receive it. This event does not disrupt the operation of the network and the nodes switch to the *Interslave* at the correct time.

Yang et al. [8] proposed another partly decentralized control strategy that can also benefit from the TTRing protocol. It is also based on the closed-loop balancing and **PSC PWM**, but avoids the data flow from cells to the central controller altogether; thus, we can suppress the protocol *interleaved* phase and only use the *cycle* phase to quickly broadcast information coming from the master. Fig. 4.5 shows the complete control strategy of the converter, including the network and the decentralized part executed at the cells. In this case, TTRing has the lowest **MCT** possible for a ring network that uses Ethernet as

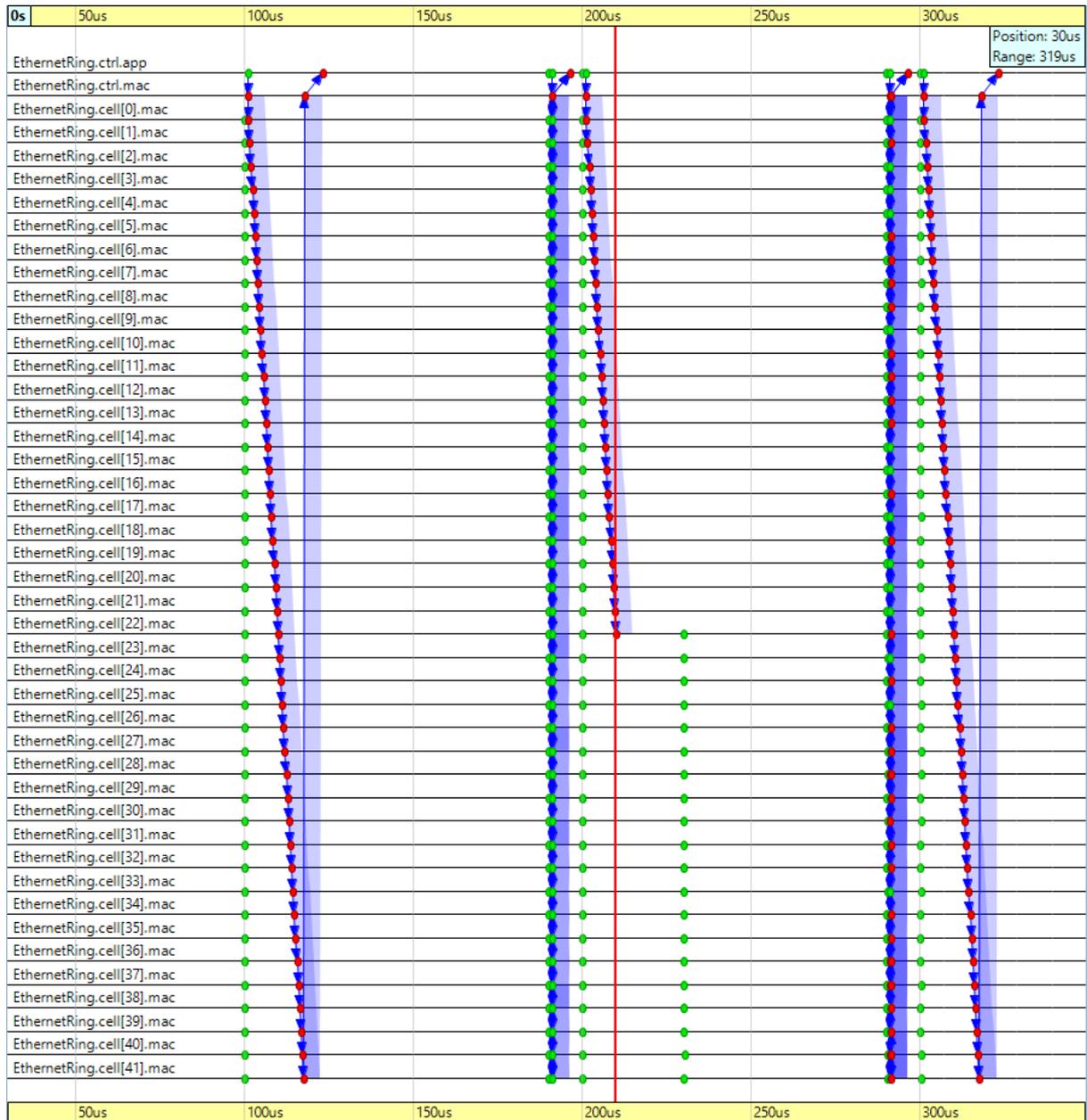


Figure 4.4: Two periods of the TTRing network with 43 nodes and modeled using OMNeT++.

communication technology.

4.1.1. Fast Forwarding

The Time-Triggered nature of the protocol allows ultra short forwarding delay of the cycle messages since reading data from the packet is unnecessary to forward it. Hence the slave nodes can immediately send the incoming packets to the next slave. To reduce the forwarding delay to a minimum, we connect the receiver data of the first port to the transmitter input of the second port and use the receiver data valid signal to enable the transceiver. No clock domain crossing is needed because the clock that drives the transmitter is the one recovered by the receiver, as explained in Subsection 3.1.6.1 (Fig. 3.6b).

We show in Fig. 4.6 details of the implementation of the slave nodes using a Field Programmable Gate Array (FPGA). The first block after the receiver is a data rate conversion block (IDDR), needed due to the Physical Layer (PHY) Reduced Gigabit MII (RGMII). The output of a multiplexer (MUX) drives the transmitter. It has two inputs: the received data and the output of a *First In First Out (FIFO)* buffer. The *FIFO* stores the data coming from the Media Access Control (MAC) layer, because it is in a different clock domain. A scheduler controls the multiplexer and the command to send data (Tx_En) when the slave is in the *Interslave* phase.

This implementation allows a single clock delay in the internal logic, i.e., 40 ns with Fast Ethernet and 8 ns with Gigabit Ethernet. The remaining components of the forwarding delay are the *PHY* transmitter and receiver latencies. We list some values found in public documents in Table 4.1. The forwarding delay of a node can be as low as 282 ns when using the quickest *PHY*.

Table 4.1: Latency of PHYs operating with 100BASE-T RGMII

Device	Manufacturer	Max. Latency
DP83867 [123]	Texas Instrument	90 ns (Tx) + 288 ns (Rx)
88E1510P/Q [124]	Marvell	362 ns (Tx + Rx)
88E1510 ^a	Marvell	1.2 μ s (Tx + Rx)
KSZ8091MLX [125]	Microchip	72 ns (Tx) + 170 ns (Rx)
VSC8601/VSC8641 [126]	Microsemi	200 ns (Tx) + 380 ns (Rx)

^a Measured.

4.1.2. Minimum Cycle Time

The *MCT* for the TTRing protocol is expressed in (4.1), where P_{master} and P_{slave} are master and slave payloads, $P_{overhead}$ is the number of bytes necessary for a proper packing of the frame (38 Bytes¹), P_{max} is the maximum payload that fits in one frame (1500 Bytes

¹Sum of preamble (7 Bytes), start of frame delimiter (1 Byte), source and destination addresses (12 Bytes), EtherType (2 Bytes), frame check sequence (4 Bytes), and interframe gap (equivalent to 12 Bytes).

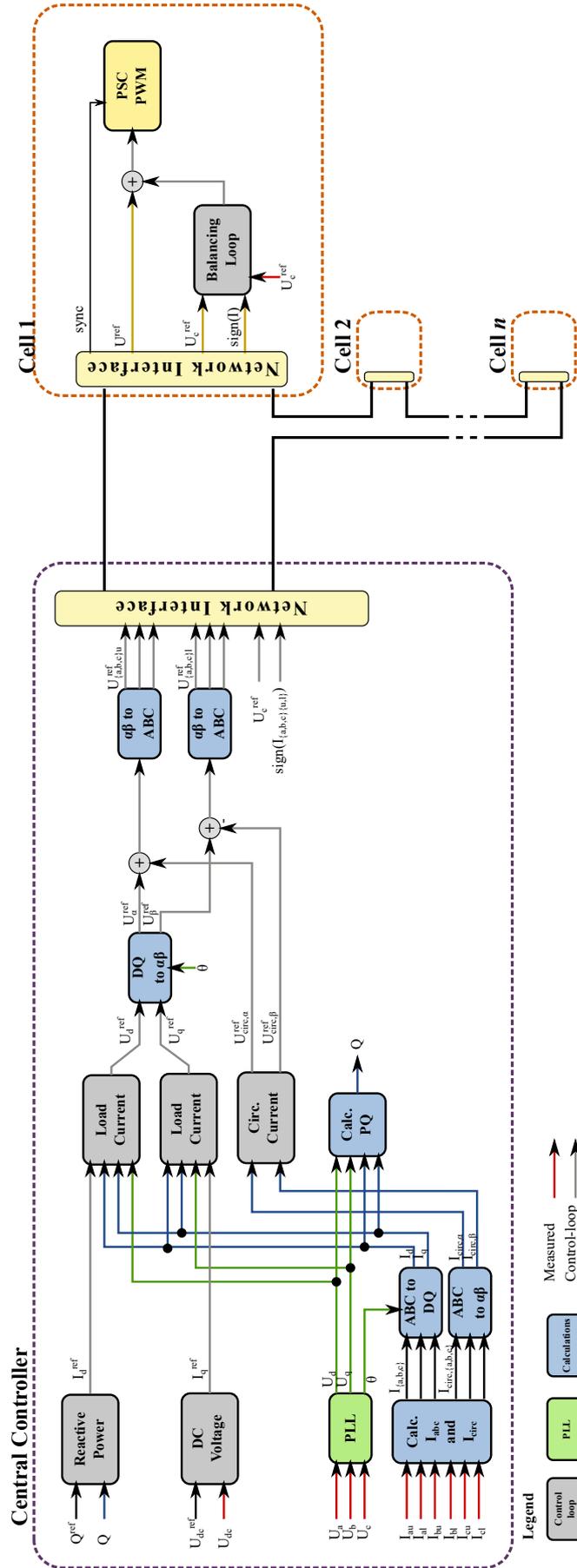


Figure 4.5: Block diagram of the control when using TTRing.

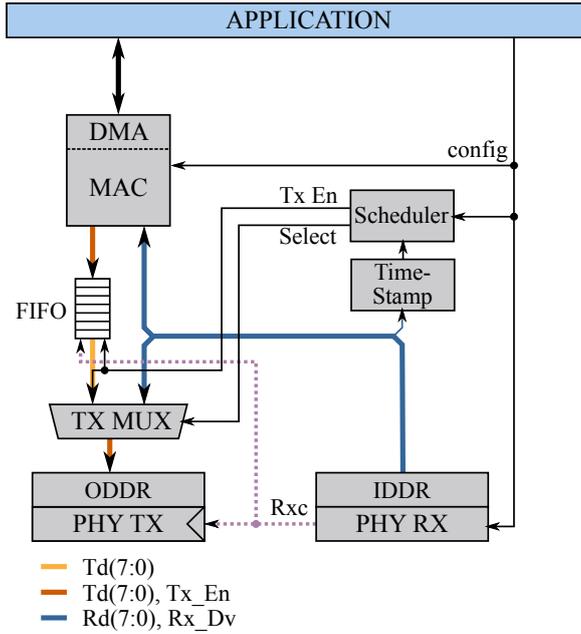


Figure 4.6: Details of the slave implementation. It uses the receiver clock Rxc as the transmitter clock.

in Fast Ethernet), GW are the guard windows necessary for the two phase changes per cycle, T_{byte} is the time to transfer one byte, T_{fw} is the forwarding delay, κ is the number of nodes, and $\lceil x \rceil$ returns the nearest integer not smaller than x . Later in this chapter, we will use this expression to compare the TTRing performance with [EtherCAT](#), [PESnet](#), and the DiSortNet protocols.

$$\begin{aligned}
 MCT_{TTRing} &= (P_{\text{master}} + P_{\text{slave}}) \cdot T_{\text{byte}} + \kappa \cdot T_{\text{fw}} + 2 \cdot GW \\
 &+ \left(\left\lceil \frac{P_{\text{master}}}{P_{\text{max}}} \right\rceil + \left\lceil \frac{P_{\text{slave}}}{P_{\text{max}}} \right\rceil \right) \cdot P_{\text{overhead}} \cdot T_{\text{byte}}
 \end{aligned} \tag{4.1}$$

4.2. DiSortNet

Several works on networked [MMCs](#) opted for phase-shifted carrier [PWM](#) and closed-loop balancing as a mean to reduce the [MCT](#) and achieve high update rates [8, 45, 86], but they mostly target converters with a reduce number of cells.

However, when we look into the installed [MMCs](#) worldwide, we recognize that they are large converters with hundreds of cells. These converters are more commonly controlled with Nearest Level Control or other non carrier-based modulation [35] that are unsuitable for the closed-loop balancing methods.

Only a few works regarding networked [MMCs](#) employ the sorting algorithm and have some flexibility in the choice of the modulation [40, 104]. Nevertheless, they fail to deliver acceptable update rates if the number of cells is higher than a few tenths.

In this subsection, we introduce the DiSortNet protocol, which pursues closing the gap between the requirements of real-life converters and what is available in the technical literature.

The DiSortNet protocol resorts to three strategies to reach Minimum Cycle Times in the same range of the TTRing, while still keeping a flexible modulation strategy. The strategies are the dual insertion sorting, the distributed Minimum/Maximum identification, and the compact modulation; following, we explain each of them.

4.2.1. Dual Insertion Sorting

Some authors proposed to identify only the cells with the minimum and maximum voltages in an arm as a strategy to balance the capacitor voltages, such that sorting implementation is computationally less expensive [61, 81]. This approach explores the fact that a single cell switches every modulation period as long as the modulation frequency is above the critical value, $f_{critical}$ (2.21). Thus, only the information at the top (maximum voltage) or bottom (minimum voltage) of the list is necessary in steady state if the min/max identification runs at the same rate as the modulation. When the current is positive, the controller needs to turn off the cell with the maximum voltage or turn on the one with the minimum voltage; when the current is negative, the controller needs to turn on the cell with the maximum voltage or turn off the one with the minimum voltage.

However, during transients, the reference may change faster, causing more than one cell to switch in a modulation cycle. Authors dealt with this situation by identifying the next list element(s) every time the controller needs to switch more than one cell [61, 81]. Hence, the identification must occasionally run at a faster rate than the modulation.

Our strategy takes a different approach because we want to avoid transmitting all the voltages values to the central controller. For this reason, we propose to execute a distributed max/min identification by means of an insertion sorting of the list every control cycle but moving only two elements at a time. It works as follows: the central controller receives at each control period the cell numbers that correspond to the maximum and the minimum capacitor voltages for each arm. It uses this information to modify the sorting list from the previous control period, moving the cell with minimum voltage to the bottom of the list and the one with maximum voltage to the top of the list (Fig. 4.7).

We simulated MMCs with 5 and 20 cells per arm operating as a Static Synchronous Compensator (STATCOM) in MATLAB/Simulink to verify the performance of this strategy. Fig. 4.8a to 4.8e and Fig. 4.9a to 4.9e show the position of five cells in the list ordered using the Dual Insertion Sorting, their actual position, and their switching function (1 for inserted; 0 for bypassed) for the 5 and 20 cells case, respectively. Ideally, the red lines should superpose the blue ones, i.e., the list using the insertion sorting should match the actual order cell. Fig. 4.8f and 4.9f show the arm current polarity (non-zero represents

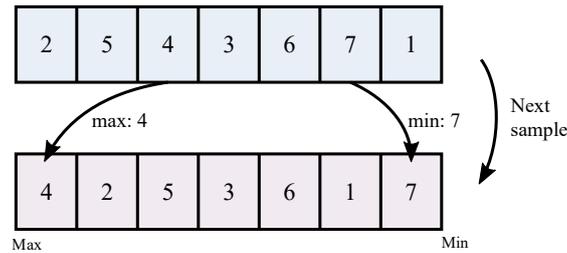


Figure 4.7: Dual Insertion Sorting. It reorders the list by moving the element with maximum value to the top of the list and with the minimum value to the bottom. The remaining elements are shifted right or left, to accommodate the new max/min, respectively.

a positive current; zero represents a negative current), the arm insertion index, and the AC component of the capacitors voltages of the converters with 5 and 20 cells per arm, respectively. All the cells are at the upper arm of phase A. Decoupled PWM generates the commands, the carrier frequency is equal to 750 Hz, and the control period is 100 μ s.

The results show that the Dual Insertion Sorting can keep the list reasonably ordered when the number of cells is low (Fig. 4.8a–4.8e), but it leads to significant deviations once the number of cells increases (Fig. 4.9a–4.9e). Nevertheless, if we look into the capacitor voltages of the converter with 20 cells and compare them when using the Dual Insertion Sorting or the regular sorting strategy, we recognize that the voltage ripple is only slightly higher (Fig. 4.10).

We have also simulated a larger converter with 200 cells per arm and controlled with Nearest Level Control (NLC) [35, Tab.6.9, E1]. In Fig. 4.11 we show the capacitor voltages of this converter when using the Dual Insertion Sorting with an update rate of (a) 100 μ s and (b) 25 μ s, and compare them with the (c) regular sorting. Note that 100 μ s and 25 μ s are, respectively, above and below the critical sampling frequency (equal to 31.5 kHz). As expected, if the update rate is not fast enough, the Dual Insertion Sorting is unable to keep the capacitor voltages well balanced.

The Dual Insertion Sorting is unable to increase its update rate during transients, as can the centralized controllers explained before, but nevertheless it can overcome the problems with transients described by Ricco et al. [73], where the lack of a complete list provokes current spikes. As the DiSortNet maintains a list (though not a perfectly-ordered one), the cells close to the bottom and the top of the list have adjacent voltages, because the capacitor voltages are states of the system, i.e., they have a limited rate of change. Therefore, the modulation can use the list to switch more than one element at once without compromising the balancing or limiting the modulator ability to follow the reference.

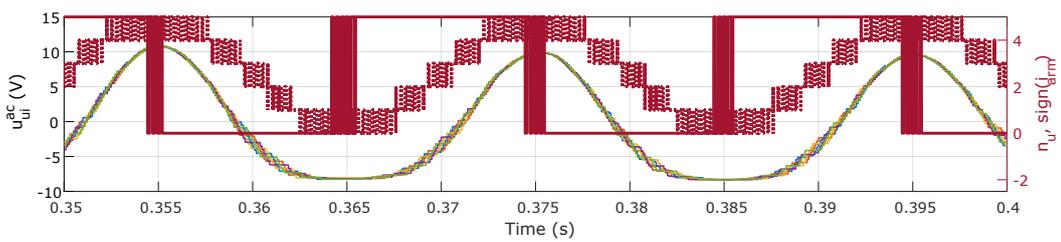
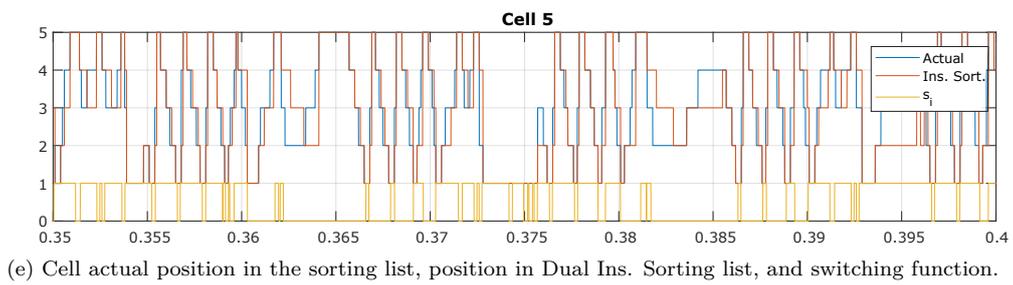
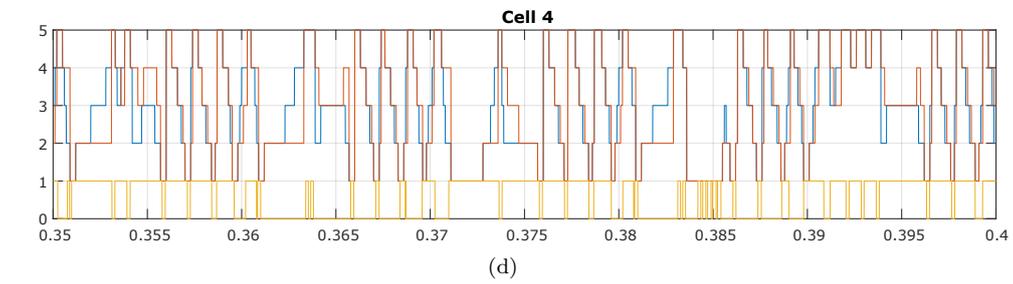
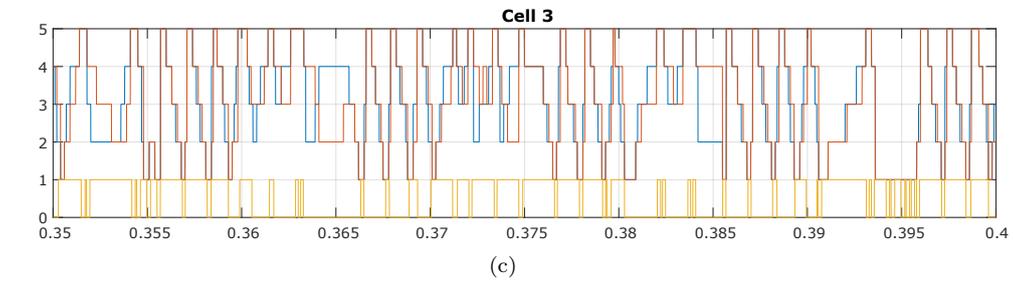
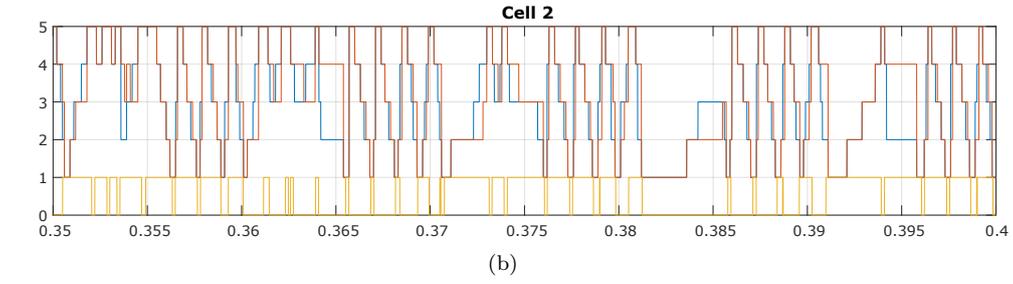
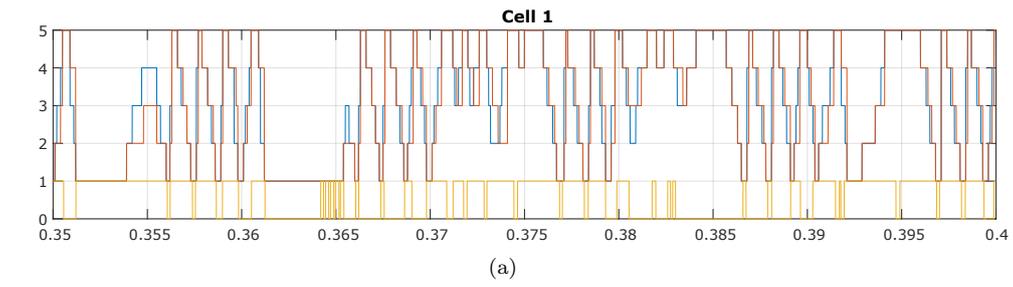
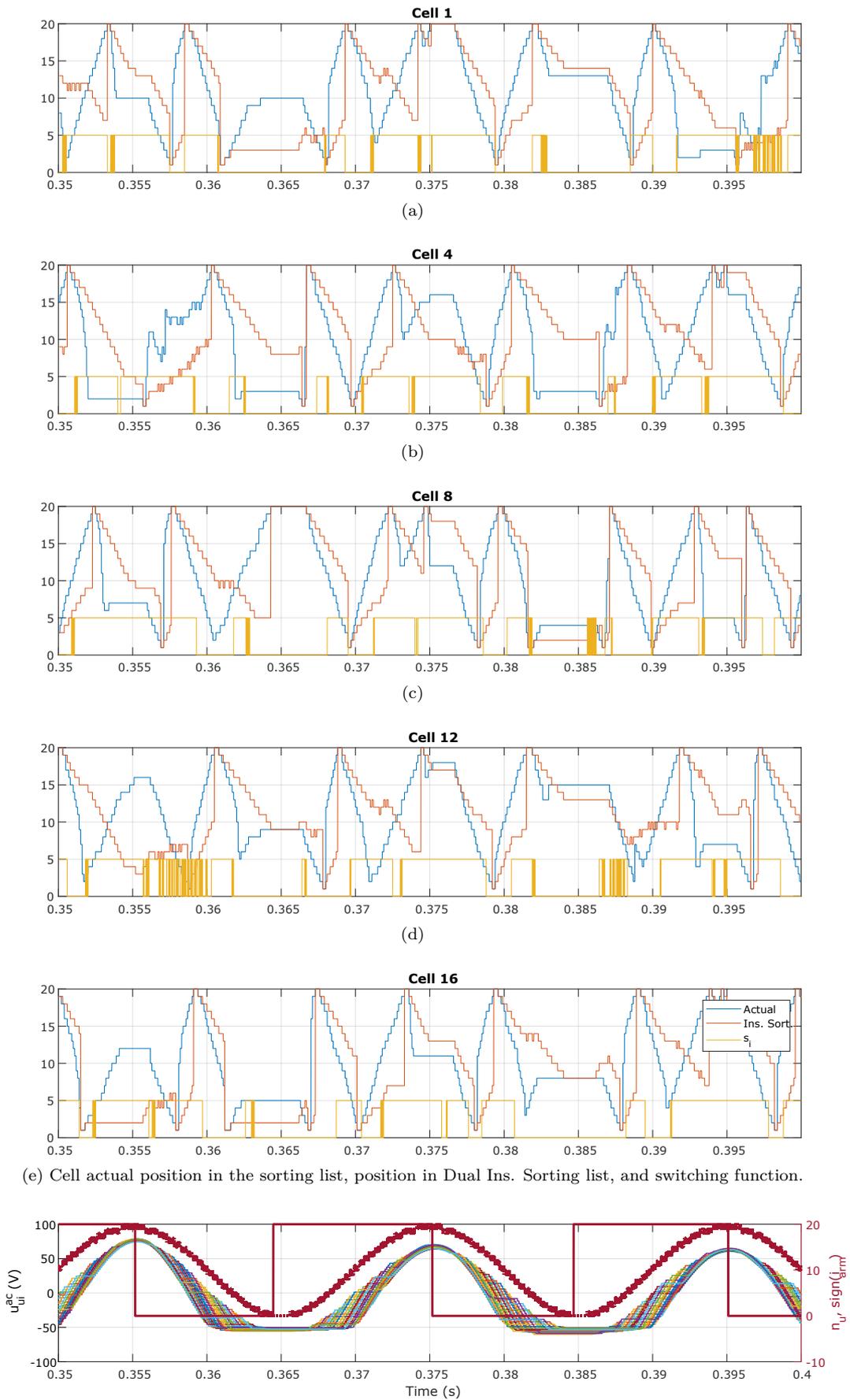


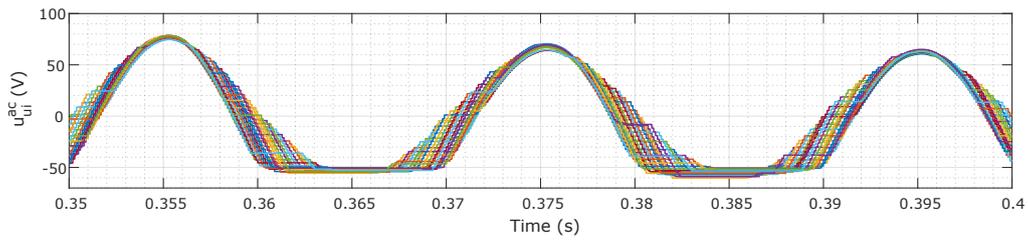
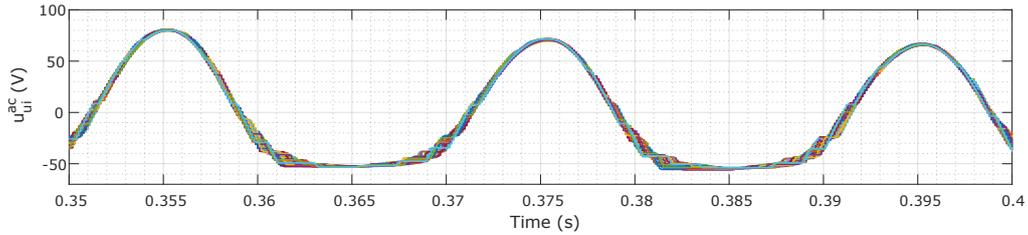
Figure 4.8: Dual Insertion Sorting in a converter with five cells per arm.



(e) Cell actual position in the sorting list, position in Dual Ins. Sorting list, and switching function.

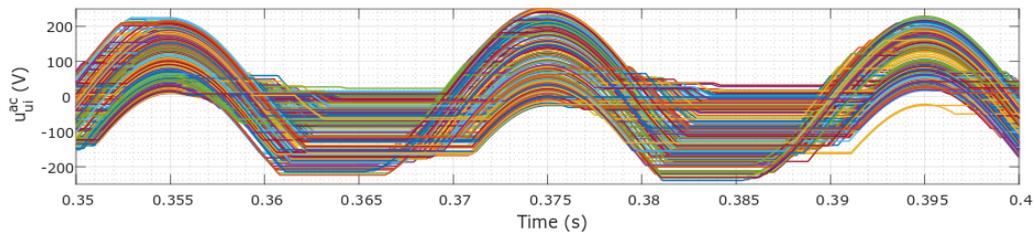
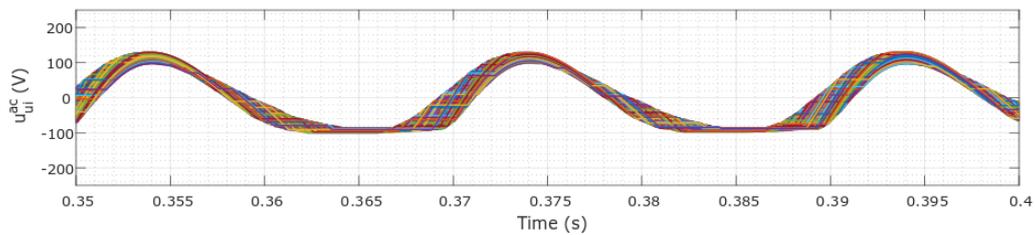
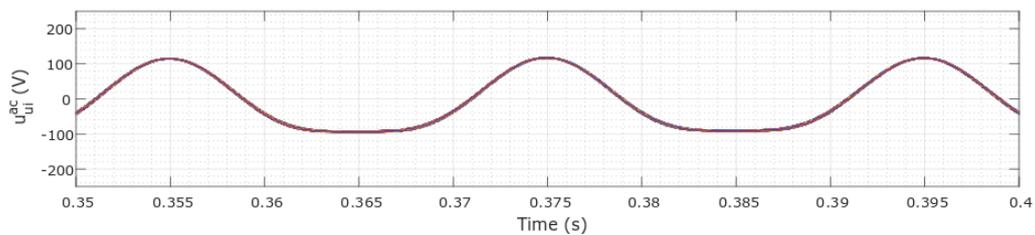
(f) AC component of the capacitor voltages, insertion index, and current polarity. Capacitor rated voltage of 1500 V.

Figure 4.9: Dual Insertion Sorting in a converter with 20 cells per arm.

(a) Dual Insertion Sorting strategy, update rate = 100 μ s.

(b) Standard sorting strategy limited to switching a single cell per modulation period.

Figure 4.10: Simulation of the capacitor voltage, phase A, upper arm, 20 cells per arm. Capacitor rated voltage of 1500 V.

(a) Dual Insertion Sorting strategy, update rate = 100 μ s.(b) Dual Insertion Sorting strategy, update rate = 25 μ s.

(c) Standard sorting strategy limited to switching one cell per modulation period.

Figure 4.11: Simulation of the capacitor voltage, phase A, upper arm, 200 cells per arm. Capacitor rated voltage of 2000 V.

4.2.2. Distributed Minimum/Maximum Identification

Besides solving the transient problems, the distributed Minimum/Maximum identification and the Dual Insertion Sorting together reduce considerably the packet payload, as we will explain next.

This strategy explores EtherCAT's on-the-fly processing and summation frame. In the DiSortNet protocol, the master node sends a single Ethernet frame that flows through the network and contains data to all slaves (Fig. 4.12a).

In this frame, a defined region, $P_{sorting}$, has fields reserved for the distributed Min/Max identification (Fig. 4.12b), and other two regions for the modulation, P_{int} and P_{pwm} (Fig. 4.12c). The first region carries the minimum and maximum voltage measurements of each arm and the indexes of the corresponding cell (at startup the master assigns indexes to the cells in an increasing order, starting from one). The master always transmits the frame with the largest value in the range (say, 0xFF) as the minimum voltage and the lowest value in the range (say, 0x00) as the maximum voltage. As the frame passes, each slave compares the values at the minimum and maximum fields that correspond to its arm with its measurement. If the local value is higher than the one in the maximum field or lower than the one in the minimum field, the slave replaces the content, both value and index, into the moving frame.

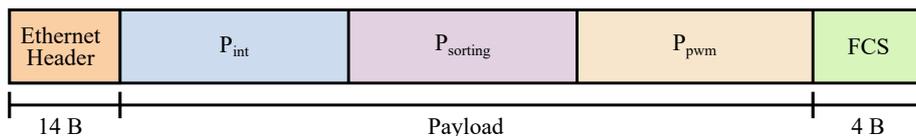
Since all slaves do the same, the master receives an edited version of the original frame with the minimum and maximum voltage of each arm and to which cells these measurements belong, which is the input data for the Dual Insertion Sorting. Note that this strategy removes the dependence of the balancing payload on the number of cells.

We have implemented a protocol node in an FPGA and have found some implementation details worth noting². First, the comparison and replacement of the voltage make necessary that the slave holds the frame a bit longer than just directly forwarding it, as we did in TTRing. The number of clock cycles added to the forwarding delay depends on the capacitor voltage measurement resolution (Res_{meas} , in bits) and is given by (4.2) for Fast Ethernet, where T_{clk} is equal to 40 ns and T_{phy} is the PHY latency (see Table 4.1).

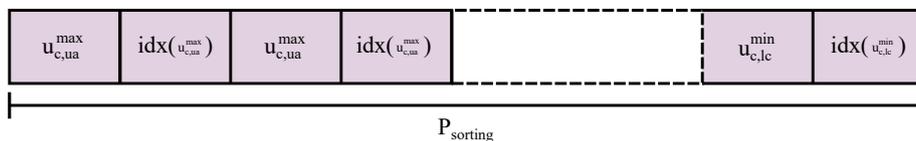
$$T_{fw}^{DiSortNet} = T_{phy} + T_{clk} \cdot \left(1 + \left\lceil \frac{Res_{meas} - 4}{8} \right\rceil \right) \quad (4.2)$$

Second, and a consequence of the first, we must reduce the measurement resolution (at least of the value transmitted to the central controller) to keep a low forwarding delay and MCT. We propose to do that by restricting the measurement range to the region of interest, e.g., between 85% and 115% of the rated capacitor voltage (Fig. 4.13). If we consider that the typical Analog/Digital Converter (ADC) resolution is twelve bits, that it will measure the entire voltage range, e.g., from 0% to 120% of the rated voltage, and also

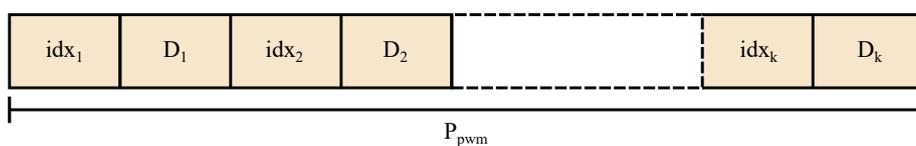
²The implementation is available for download at <https://github.com/tpcorrea>.



(a) Summation Frame.



(b) Detail of summation frame part related to the sorting.



(c) Detail of the summation frame part related to the PWM generation.

Figure 4.12: DisortNet protocol frames.

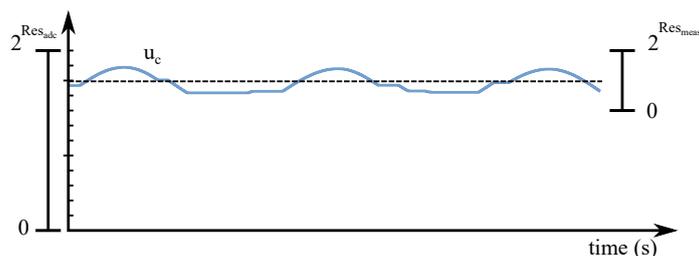


Figure 4.13: Capacitor voltage representation with a reduced number of bits without loss of accuracy.

that such ADCs lose at least the least significant bit due to noise, we can transmit only eight bits ($Res_{meas} = 8$), losing only one bit of precision as we save two bits restricting the measurement range to 85%–115% and one bit more ignoring the least significant bit.

Third, when a slave modifies the payload content, it must update the Frame Check Sequence (FCS) field at the end of the frame, so the next nodes recognize it as valid. Not only this is necessary but also the slave must check the incoming frame validity before sending a correct FCS; otherwise, the downstream nodes would take a corrupted frame as valid only because the upstream node that has modified it re-validated the frame with a new FCS.

4.2.3. Compact Modulation

The unified PWM strategy, discussed in 2.3.1, splits the pulse generation into an integer (2.17) and a fractional part (2.18) (repeated below as (4.3) and (4.4), respectively,

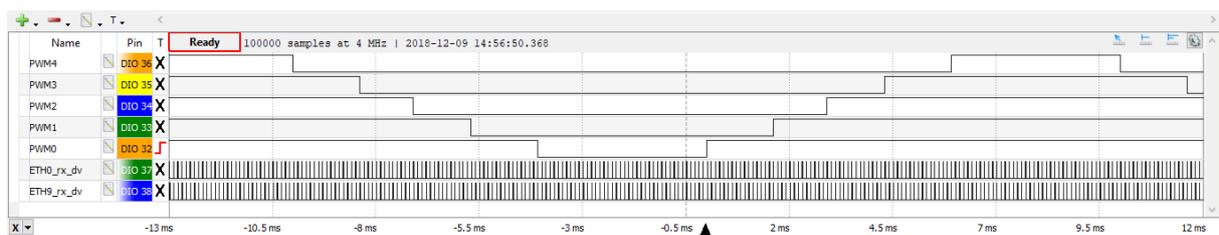
where $\lfloor x \rfloor$ returns the greatest integer less than or equal to x and $frac(x)$ represents the decimal part of x). The unified PWM allows generating not only the pulse patterns of any carrier-based PWM but also NLC and other modulation types [61].

$$n_{\{u,l\}}^{int} = \lfloor n_{\{u,l\}} \rfloor \quad (4.3)$$

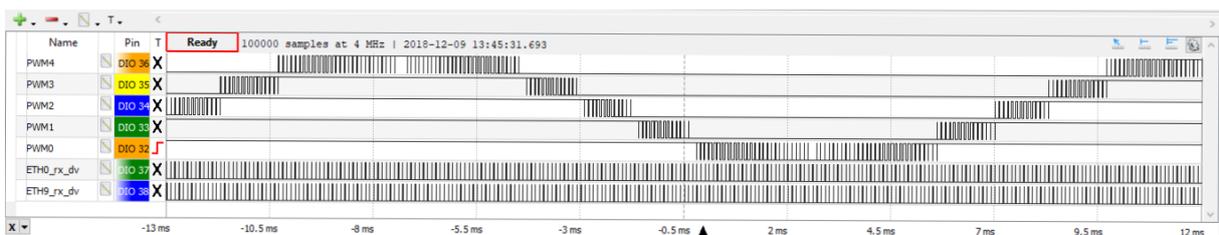
$$D_{\{u,l\}} = frac(n_{\{u,l\}}) \quad (4.4)$$

The compact modulation strategy employs the unified PWM and reduces the amount of data necessary to synthesize the reference waveform at the output. It splits the modulation information into two regions, represented by P_{int} and P_{pwm} in Fig. 4.12a. In the first region, corresponding to the integer part, each bit controls the cell with the corresponding index to be inserted (bit = 1) or bypassed (bit = 0), e.g., bit one controls cell one to insert its capacitor when one, or bypass the capacitor when zero. In the second region, the central controller addresses an arbitrary number of cells by sending their index (idx_i) followed by a PWM reference (D_i) (Fig. 4.12c). If $cell_i$ recognizes its index in the PWM region of the frame, it will load D_i in the compare register and generate a square waveform.

We implemented a network with five slaves and one master in our Node Carrier board (see Appendix A for more details) that corresponds to one arm of the MMC prototype available. We present the result of the cell commands for a pure NLC modulation, Fig. 4.14a, and a carrier-based modulation with a frequency of 10 kHz, Fig. 4.14b. In both figures, $PWM0-4$ correspond to the pulse of the top power switch of cells 1-5, respectively, and the signals $ETH0_rx_dv$ and $ETH9_rx_dv$ are the PHY data valid signal of the first node (cell 1) and the second port of the master (that signals that the frame has returned to it).



(a) Nearest Level Control.



(b) Decoupled PWM.

Figure 4.14: Experimental results of pulse commands over the network using the Compact Modulation Strategy.

4.2.4. Control Architecture

When using DiSortNet, the control strategy is centralized and the cells function is both of an actuator and an intelligent sensor that only output its value if it is higher or lower than the one informed by the network. Fig. 4.15 shows the block diagram of this solution.

4.2.5. Minimum Cycle Time

The MCT for the DiSortNet protocol is expressed by (4.5), where:

- P is the packet total payload;
- P_{int} and P_{pwm} are the payload data due to the integer and PWM parts of the modulation, respectively;
- $P_{sorting}$ is the payload data due to the sorting algorithm;
- $P_{overhead}$ is the number of bytes necessary for a proper packing of the frame (38 Bytes);
- P_{max} is the maximum payload that fits in one frame (1500 Bytes in Fast Ethernet);
- n_ϕ is the number of phases;
- Res_{pwm} and Res_{meas} are, respectively, the PWM and the capacitor voltage measurement resolution in Bytes;
- the term $\left\lceil \frac{\kappa}{255} \right\rceil$ represents the number of Bytes necessary to address all the cells.

$$MCT = P \cdot T_{byte} + \left\lceil \frac{P}{P_{max}} \right\rceil \cdot P_{overhead} \cdot T_{byte} + \kappa \cdot T_{fw} \quad (4.5)$$

$$P = P_{int} + P_{pwm} + P_{sorting} \quad (4.6)$$

$$P_{int} = \left\lceil \frac{\kappa}{8} \right\rceil \quad (4.7)$$

$$P_{pwm} = 2 \cdot n_\phi \left(Res_{pwm} + \left\lceil \frac{\kappa}{255} \right\rceil \right) \quad (4.8)$$

$$P_{sorting} = 4 \cdot n_\phi \left(Res_{meas} + \left\lceil \frac{\kappa}{255} \right\rceil \right) \quad (4.9)$$

In (4.5), the first term corresponds to the frame duration due to the payload, the second term to the delay caused by the frame header and interframe gap, and the last term to the forwarding delays of the slaves.

As we noted previously, the Dual Insertion Sorting strategy requires the update rate to be faster than the critical sampling frequency; hence, as both the critical sampling frequency (2.21) and the DiSortNet MCT (4.5) increase with the number of nodes, the protocol, as proposed in this text, has satisfactory balancing performance in networks up to 300 nodes (Fig. 4.16), corresponding to a three-phase double-star MMC with up to

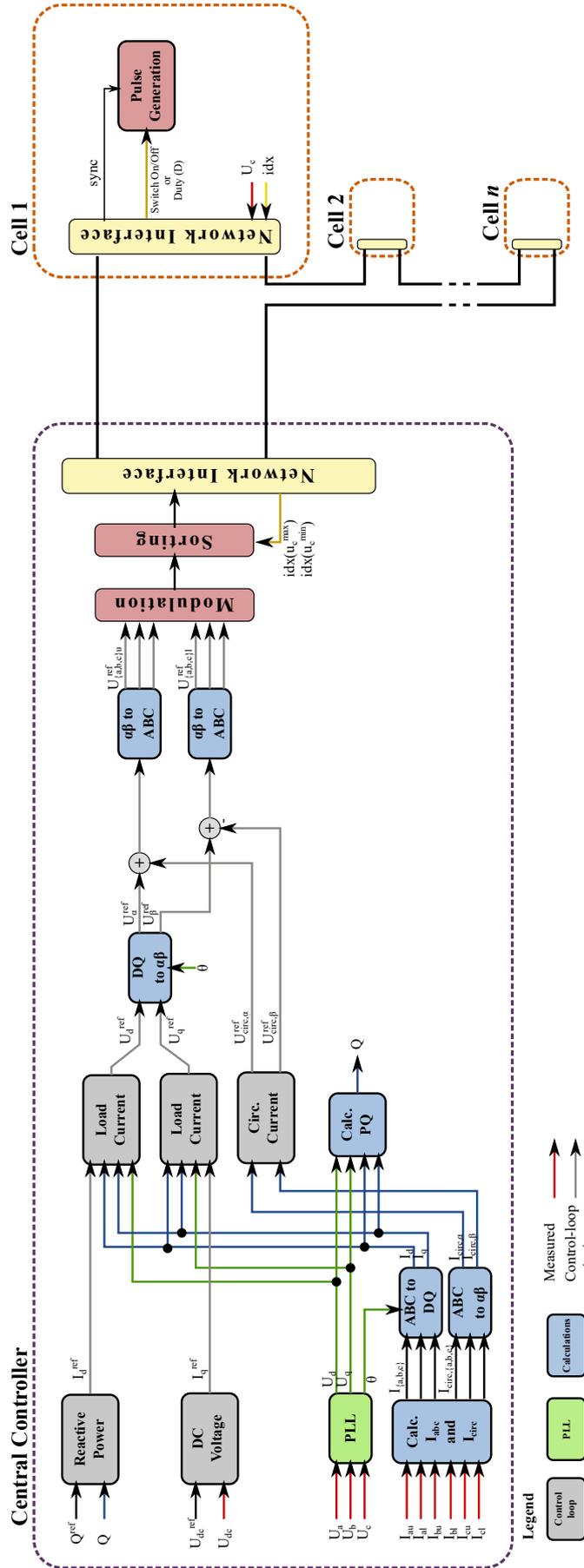


Figure 4.15: Block diagram of the control when using DiSortNet.

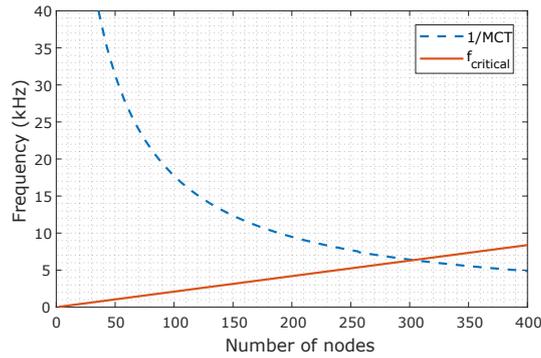


Figure 4.16: Minimal critical frequency and the inverse of the DiSortNet MCT. The necessity to run the Min/Max identification faster than the Minimum Critical Frequency limits the protocol coverage to networks with less than 110 nodes approximately.

50 cells per arm (the number of nodes κ in (4.5) is equal to $6 \cdot n$ in (2.21) due to the converter six arms).

4.3. Performance Comparison

In this subsection, we compare the Minimum Cycle Time of PESnet, EtherCAT, TTRing, and DisortNet. We considered that each node adds 4 Bytes to the payload when the protocol is EtherCAT [40], but the payload³ size remains constant at the minimum (46 bytes) when using TTRing, either in its Fast Ethernet (100BASE) or Gigabit Ethernet (1000BASE) versions, and obeys Equation (4.6) for DiSortNet.

As already mentioned, the forwarding delay is a crucial parameter for this analysis. For EtherCAT, it is not easy to put a number into it: Prytz [118] mentions a delay of 500 ns; Vitturi *et al.* [127] measured an average delay of 1 μ s; Orfanus *et al.* [109] wrote that it is lower than 1 μ s. This divergence is mainly due to different implementations, PHYs and configurations adopted. We opted to take the information from the latest Application Specific Integrated Circuit (ASIC) from Beckhoff that, as EtherCAT inventor, can arguably be considered one of the fastest alternatives. According to the ET1100 Hardware datasheet [113], the maximum forwarding delay for Media Independent Interface (MII) to MII is 335 ns plus the PHY delay, when the Rx Buffer is set to the default size of 7. For PESnet, we assumed a forwarding delay of 460 ns [46]; for TTRing a delay of 430 ns; for DiSortNet a delay of 470 ns. The TTRing and DiSortNet forwarding delay correspond to a PHY latency of 390 ns, which is not the quickest PHY listed in Table 4.1 but matches the values measured in our Node Carrier board.

Fig. 4.17 shows the estimated MCTs using (3.13), (3.15), (4.1), and (4.5) for PESnet, EtherCAT, TTRing, and DiSortNet, respectively.

For small networks, the TTRing protocol has the worst result due to the overhead

³Note that the payload here excluded the headers.

generated by sending two packets per cycle. As soon as the number of nodes increases

, PESnet lags behind the others, mainly due to its high protocol overhead. For networks larger than 31 nodes, TTRing 100BASE is faster than EtherCAT but remains behind DiSortNet.

As the number of nodes increases further, the gap between TTRing and DiSortNet closes because the latter has higher forwarding delay. The DiSortNet protocol MCT is half as long EtherCAT's when the network is larger than 86 nodes and is equal to 205 μs when the network has 400 nodes, against 209 μs of TTRing 100BASE, 183 μs of TTRing 1000BASE, and 446 μs of EtherCAT. The Gigabit implementation of TTRing is faster than all the other protocols for any network size.

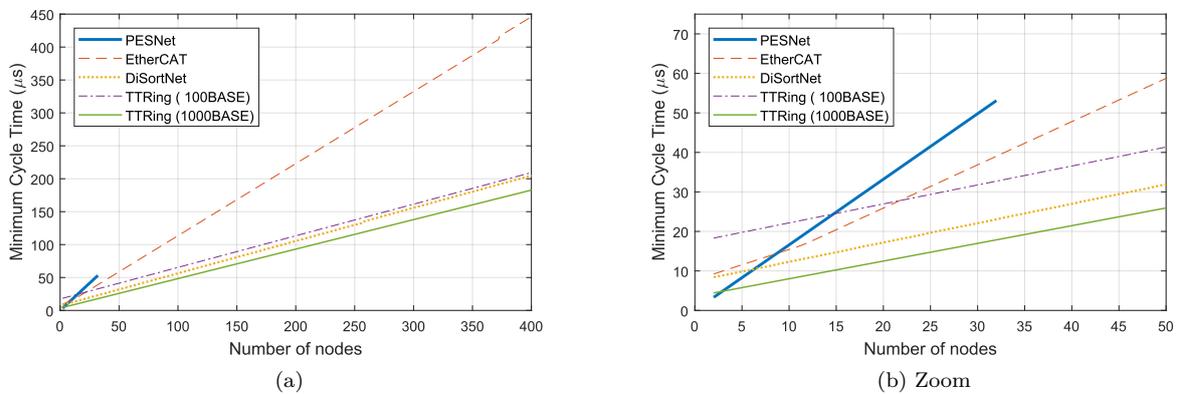


Figure 4.17: Minimum Cycle Time depending on the network size, when the payload is 48 bytes (TTRing), 4 bytes/node (EtherCAT), or following (4.6) (DiSortNet).

4.4. Fault Signal

In power electronics converters, it is essential to quickly suppress the pulses to the power switches once detecting a fault. While in a star, centralized control architecture, this is easily accomplished by merely stopping the pulses to the power switches, but in a network controlled converter it is more challenging to come to a complete halt within a short delay. The most straightforward strategy would be to broadcast a packet with a fault message to stop all the nodes. However, if the controller detects the fault condition just after it starts transmitting a long packet, this packet could cause significant interference in the transmission time of the fault message; the latency can be beyond a few hundred of microseconds. In the following subsections, we suggest some possibilities for reducing the latency of fault messages.

4.4.1. Error signal

A simple solution is to equip the cells with an additional (low speed) optical receiver and transmitter and connect all the cells in a ring. The transmitter emits light if both the cell has no critical fault and is receiving light on the receiver. Therefore, any device under fault, or if the ring is open due to a broken or unconnected fiber, would bring the system to a complete stop.

4.4.2. Bit Encoding

Designers employ bit encoding to help the receiver in clock recuperation, but we can use it for signaling fault conditions, too. It works as follows: for every x -bits of data, the transmitter sends y -bit ($y > x$) and maps all the 2^x bits in the new 2^y space. As an example, consider the 4b/5b encoding shown in Table 4.2.

Table 4.2: 4b/5b Bit Encoding.

Data			Data		
(Hex)	(Binary)	4B5B	(Hex)	(Binary)	4B5B
0	0000	11110	8	1000	10010
1	0001	01001	9	1001	10011
2	0010	10100	A	1010	10110
3	0011	10101	B	1011	10111
4	0100	01010	C	1100	11010
5	0101	01011	D	1101	11011
6	0110	01110	E	1110	11100
7	0111	01111	F	1111	11101

Since the number of transmitted bits is higher, some codes do not map to valid data, e.g., 11000 ('J') and 10001 ('K'). The user can use these symbols to preempt the on-going transmission and tell the next receiver that the following Byte(s) is(are) an error code. Consider the example below: the data to transmit is '536584FA543'; after transmitting '536584F', an error occurs, and it is necessary to tell the other nodes about it. Instead of transmitting 'A', the node which has detected the fault would transmit 'J', then the error code followed by 'K'. After 'K', it can resume transmitting 'A543'.

Though this method has minimum latency, it reduces the effective data transfer by x/y and, more important, needs support from the Physical Layer. Unless the designer is also responsible for the implementation of this layer, it is difficult to find integrated circuits that allow the higher layers to use these unmapped codes (TAXIchip is one of them [47]).

4.4.3. Magic Number

An alternative to bit encoding is a magic bit sequence that tells the receiver that the following data is an error code. This magic number could be, for instance, a sequence of five '1's. As the normal data may perfectly have '11111', the transmitter must insert a

"dead" zero to prevent normal data of triggering a fault. The receiver, then, ignores the "dead" zero. For clarity: the data to transmit is '1100 1101 1111 0011'. This data has five '1's in sequence, so the transmitter would add a '0' after that sequence and transmit '1100 1101 11110 0011'. When the receiver detects five '1' followed by a '0', it knows that the data has a dead '0' and removes it.

This method also provides minimum latency, as the previous one, but has a lower effect in the data transmission rate. It could, in the limit case when the data transmitted is only '1's, reduce the data rate by the same amount as the data encoding, but this is an unlikely case.

A more serious handicap to its use is that it breaks the Ethernet frame structure. The only field that the user has the freedom to insert arbitrary data is the payload, but as its size must be coherent with the total frame length written in the header length/type field, it is not possible to insert "on-the-fly" the magic number without losing data. Additionally, the higher layers have to handle the payload data to the [MAC](#) already with the dead '0's. If the transmitter must wait for the payload field to transmit the magic number, it can have a latency of 22 Bytes⁴ or 1.76 μ s with Fast Ethernet.

4.4.4. Gigabit Ethernet

The fourth option is to employ a faster communication technology, like Gigabit Ethernet. With the faster data rate, the time to transmit a packet falls proportionally; thus the system can tolerate the delay caused by the interference of the packet being transmitted when the fault occurs.

Though Gigabit Ethernet is standard in commercial devices, its deployment in industrial networks has been slow. As already discussed, in this application field, in particular, the low offer of Plastic Optic Fiber transceivers is a limiting factor. Plastic Optic Fibers are much easier to handle and more robust than the Glass option; therefore its use is preferable.

4.5. Simulation of MMC With Communication Network

Though several authors have explored the use of a digital communication network in a Modular Multilevel Converter, the simulation of such systems specifically accounting for the network impact has been overlooked. Simulations, however, are helpful to analyze the system behavior under specific operational conditions. In networked [MMCs](#), it can help to observe the impact that events in the network, e.g., delays, errors, losses, may have on the control performance. To close this gap, we adopted two simulation environments:

⁴Sum of the preamble (7 Bytes), start of frame delimiter (1 Byte), source and destination addresses (12 Bytes), and the length/type (2 Bytes) field.

OMNeT++ for modeling the network and MATLAB/Simulink for modeling the control and power circuits.

OMNeT++ is a C++-based discrete event simulation framework, developed by András Vargas since 1997. It is well documented and free for education and research use at non-profit research institutions. Rather than a simulator for a specific purpose, OMNeT++ provides infrastructure and tools for writing models in a variety of domains, such as wired and wireless networks, protocols, queuing networks, and multiprocessors systems [128]. Thus, it is a powerful tool to model a network and the underlying embedded hardware.

The usual approach to cosimulation is to run both simulators in lock-step [129], reflecting the events of one side on the other and vice-versa. However, in our case, the time-triggered protocol is not influenced by the plant. On the contrary, it is only the network that influences the plant. Thus, we propose a novel approach to similar situations.

The first step in our cosimulation workflow is to model the network and analyze the results with OMNeT++ tools. The communication packets should be grouped under labels that will represent values, e.g., references and measurement data, transported together. Once a correct model is ready, the next step is to record the time instants when the packets reach the nodes. OMNeT++ provides the class *cOutVector* with this purpose and generates a .vec file with the data from all nodes. With this step, the work in OMNeT++ is complete, so that we can move to MATLAB/Simulink environment.

In MATLAB, we need to convert the OMNeT++ .vec file into vectors, one for each node and data group. To each vector, we add a second column with alternating zeros and ones, creating matrices, so we can modify a working model in Simulink to account for the network behavior. We do that by introducing a triggered sub-system between the source of information, e.g., a sensor or control loop output, and the actuator, namely the cell in the MMC. The sub-system connects the output to the input (Fig. 4.18), but, because of its triggered nature, the update happens only with a transition in the trigger signal.

The trigger signal is the output of the *source: From workspace* block, used to import the matrix that represents the corresponding input signal and node. The modified model includes now the impact of exchanging information over the network. This approach has significant advantages in managing the cosimulations and reducing its execution time, as it is not necessary to run the network simulation every time and no data moving across the simulators takes place at runtime.

Each simulation used to produce the results shown in the next subsection took 2 minutes to run. On the other hand, this strategy is limited to network delays smaller than one control sampling period, a limitation we are working to overcome.

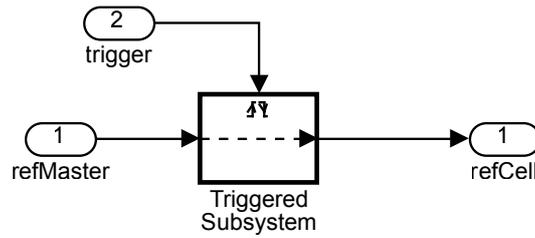


Figure 4.18: Use of a triggered sub-system to include the network behavior.

4.5.1. Simulation Results

The result of the OMNeT++ simulation (Fig. 4.4) shows an important characteristic of the ring topology: when a packet is lost or corrupted (around $t = 420\mu\text{s}$), all the following nodes are affected, increasing the node probability of not receiving a new setpoint the farther it is from the central controller.

The cosimulation allows assessing the influence of the loss of packets in the system. We have simulated a medium voltage STATCOM with seven cells per arm, $42 + 1$ nodes in total. We show in Fig. 4.19 the reference voltage sent by the central controller (the dashed line) and the local reference of the last seven nodes of the network (the solid lines). These nodes correspond to the cells of the lower arm of phase C and are the ones with the highest probability of not receiving a packet. We modeled the network with two probabilities of losing a packet: 0.0001 (Fig. 4.19a) and 0.005 (Fig. 4.19b). These waveforms illustrate how the network can have a significant impact on the system. For example, the Total Harmonic Distortion (THD) of the grid current is 2.9% when a model of the network is not included, 3.7% and 5.5% when the network is included for cases (a) and (b), respectively.

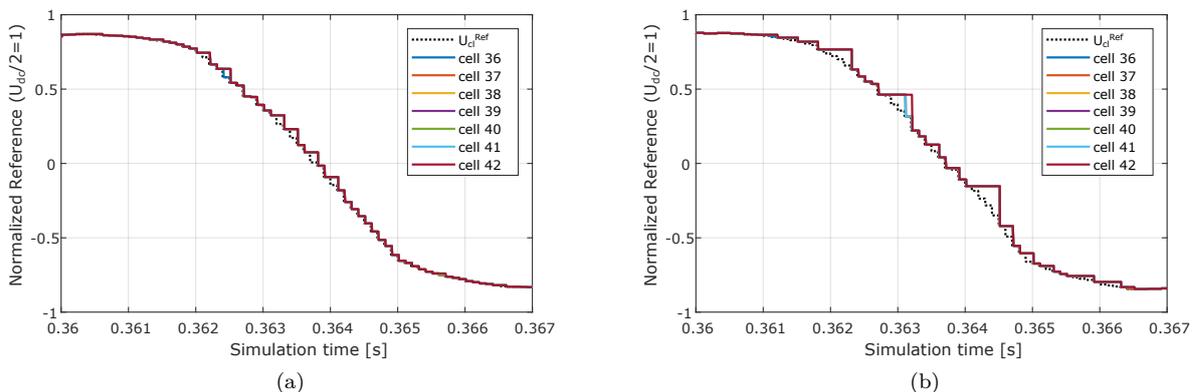


Figure 4.19: Central controller sends a reference to the nodes (dashed line), but due to failures in the network, deviations occur (solid lines). We modeled the network with two loss probabilities: (a) 0.0001 and (b) 0.005.

4.6. Conclusions

In this chapter, we have proposed two Ethernet-based protocols for the control of MMCs that have an internal control network with a ring topology. The proposed protocols are tailored to converters with specific control strategies.

The first protocol, TTRing, reduces the MCT of the network to values close to the minimum by adopting the Time-Triggered paradigm. As the slaves know in advance that the incoming frame must be forwarded to the next node, it can reduce the forwarding delay to the lowest value possible with the adopted link technology. The TTRing protocol covers converters that use the closed-loop balancing strategy and the phase-shifted carrier PWM. We argue that both strategies apply to converters with a low number of cells and that this is only a small part of the MMCs constructed worldwide.

As an alternative to this limitation, we propose a second protocol, named DiSortNet. The most important characteristic of this protocol is that it distributes part of the sorting algorithm among the network nodes. The implementation of this kind of sorting at the central controller is more straightforward and requires fewer data. Hence, the reduction of data transferred from the cells to the master node, together with a short forwarding delay, enables the DiSortNet protocol to outperform the TTRing in a wide range of converters, while supporting a flexible modulation strategy and the more common sorting algorithm.

Furthermore, we have discussed possibilities of quickly stopping the cells in case the control detects a critical fault condition, though some of the strategies are difficult to implement when the protocol employs the Ethernet frame format. Specifically, as a first choice, we suggest the use of the Error signal, because it is easy to implement and has low latency.

Lastly, we have also introduced a strategy to accurately simulate a network controlled MMC, accounting for the communication artifacts in a more efficient way than traditional cosimulation approaches. Specifically, we showed how our cosimulation allows assessing the impact of realistic error patterns on relevant MMC performance metrics, such as the Total Harmonic Distortion.

Chapter 5

Minimal Reception Delay for Ethernet Interfaces

In the previous chapters, we employed the Minimum Cycle Time (MCT) as a performance indicator to compare real-time protocols. However, in high-performance applications, like drives [127] or the control of Modular Multilevel Converters studied here, not only the cycle time has to be low but also the end-to-end latency must be as short as possible. The reason, as already explained in Chapter 3, is that the communication latency adds to the loop delay.

In our research, we found little information on the time the nodes need to make incoming data available to the application layer or the time necessary to effectively start transmitting data through the link. As protocols inexorably move to higher data rates, these delays will become more relevant, so it is essential to know their magnitude and what are the main aspects influencing it.

Orfanus et al. [109] list some strategies they adopted to optimize the implementation of an EtherCAT master, such as zero-copy buffers, memory pre-allocation, and mapping of application variables directly onto EtherCAT telegrams, but they omit timing figures. We wanted to characterize the delay inside the nodes and understand how both hardware and software implementations affect it. For that, we run experiments on Xilinx Zynq System-on-Chip (SoC), adopting different Media Access Control (MAC) implementations, data copy strategies, and using or not the User Data Protocol (UDP) and Internet Protocol (IP) stacks.

Though these experiments were performed in a defined platform, the results can easily be translated to other devices from different manufacturers because of the variety of MAC implementations that are possible in the Zynq SoC, as we will see next. For example, TI's Keystone architecture, employed in the C667x and C665x families, uses a MAC peripheral that has a local First In First Out (FIFO) buffer and transfers the information from/to the main memory using a Direct Memory Access (DMA) engine [130], just like the Zynq Gigabit Ethernet MAC (GEM).

Furthermore, the measurements obtained with the Zynq device allowed the identification of the main reasons for the transmission and reception delay. Thus, after discussing the results at the end of Section 5.3, we propose hardware accelerators¹ that can be easily implemented in Field Programmable Gate Array (FPGA) technology to minimize the reception time of packets and remove the dependence on the packet size. The key strategy adopted is to move the packet directly to the final memory destination as the Physical Layer (PHY) receives it, thus not waiting for the complete reception, as it is the norm. This strategy keeps the MAC layer intact; hence the node remains fully compliant with the Ethernet Standard. We present details of the accelerator implementation and experimental results to demonstrate its effectiveness. We achieved a uniform delay just above 1 μ s for any packet size, which is significantly quicker than standard implementation and represents a substantial reduction both in the master and slave nodes.

5.1. Media Access Control

Before we look into the implementation possibilities of the Media Access Control and how they affect the reception latency, let us first remember why do we need the MAC in the first place. The MAC performs, together with the Logical Link Control, the functions described by the Open Systems Interconnection (OSI) model for the Data Link Layer. Its main functions are [131]:

1. Data encapsulation (transmit and receive)
 - a) Framing (frame boundary delimitation, frame synchronization)
 - b) Addressing (handling of source and destination addresses)
 - c) Error detection (detection of physical medium transmission errors)
2. b) Media Access Management
 - a) Medium allocation (collision avoidance)
 - b) Contention resolution (collision handling)

Therefore, it is the MAC sublayer that defines Ethernet packet format, including its fields and size, the addressing possibilities (unicast, multicast, and broadcast), the order of bit transmission, and the mode of operation (half- or full-duplex). It is also responsible for generating and verifying the Frame Check Sequence, for supporting nodes isolation in a same physical network with VLAN tags, and for enforcing the minimum and maximum frame length, the interframe gap, and the media access rule – Carrier Sense Multiple Access with Collision Detection (CSMA/CD, ignored in full-duplex mode).

¹The implementation is available for download at <https://github.com/tpcorrea>.

5.2. Implementation Details and Possibilities

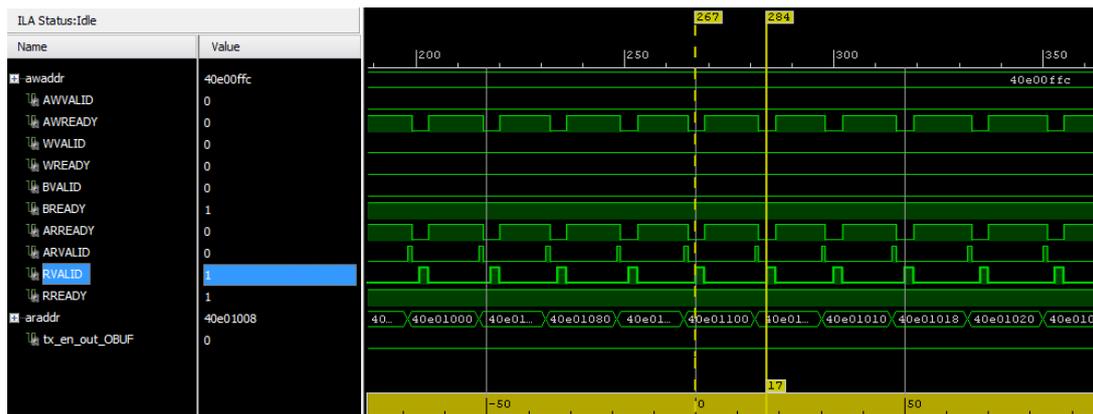
The Zynq System-on-Chip combines a single or dual-core ARM processor with an FPGA fabric. It includes up to two hard Media Access Controller, named GEM. The GEM interface with the PHY is either the Gigabit MII (GMII) or the Reduced Gigabit MII (RGMII) and with the core and memory it uses a 32-bit AHP bus. A DMA engine, operating at a maximum frequency of 150 MHz (IC speed grade -2) [132], controls the flow of data to/from the main memory.

Besides the GEM, Xilinx provides two types of MAC as Intellectual Properties for synthesis and implementation inside the FPGA: the Ethernet Lite, free of charge but limited to 100 Mbps, and the soft Tri-Speed MAC (TEMAC), supporting up to 2.5 Gbps.

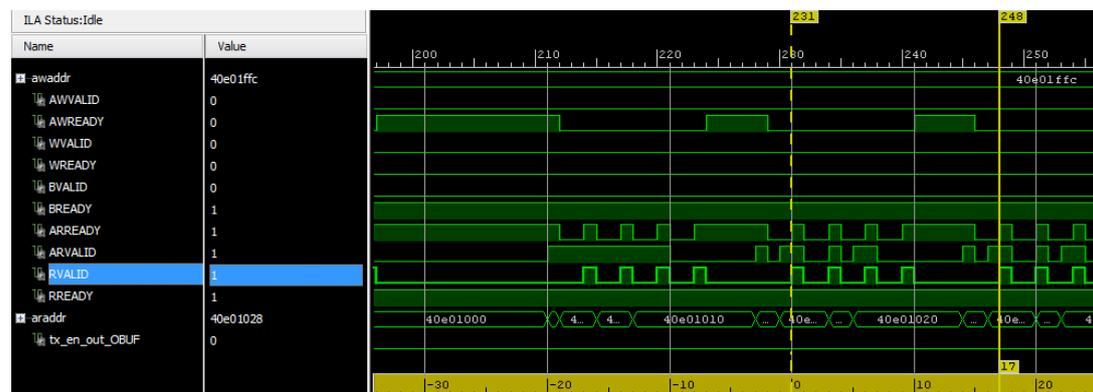
The Ethernet Lite soft MAC connects to the main memory via either AXI4-lite or AXI4 slave interfaces. The former does single transactions, only, what limits performance, and has a maximum clock of 150 MHz. The latter supports burst transactions of 256 words with a single addressing phase [133]. The AXI Master, though, does not use data bursts when running the demo echo server application, so the AXI4 performance is similar to AXI4-lite, with a minor gain due to the higher clock rate (up to 180 MHz). In both cases, the CPU transfers data with higher bandwidth by configuring the Ethernet Lite memory address region as *device memory*² instead of the default *strong-ordered* (see [134], chapter 3): the number of clocks between valid write responses reduces from 18 to 3 and between reading transactions from 17 to 4 (Fig. 5.1), according to our measurements. The impact in the delay is significant: 35 μ s against 7.73 μ s to receive an 1024 Bytes UDP packet. We tried using DMA engine to accelerate the data transfer, but its rate was the same as when the core managed the data movement, and the total delay increased, due to the time spent configuring and triggering the DMA. These observations show the crucial role of the effective data transfer bandwidth in the transmission and reception delay of the nodes, independent of which device it is implemented in.

The other type of MAC available, TEMAC, supports AXI4-stream. AXI4-stream removes the addressing phase altogether [133] and, combined with an AXI DMA, can connect to the main memory using either of the high bandwidth interfaces, AXI ACP or AXI HP. In both cases, the maximum clock is also 180 MHz, but the bus width is 64-bits so that one can expect twice the data transfer rate than with the 32 bit wide AHP bus. However, our design was able to meet the timing constraints only with a clock of 150 MHz.

²Unlike strong-ordered memory, a write to device memory is allowed to complete before it reaches the peripheral accessed by the write.



(a) Strong-Ordered



(b) Device Memory

Figure 5.1: AXI Read transaction from the Ethernet Lite MAC to the processor main memory. When the memory range is configured as Device Memory, the AXI Master reads data four times faster (the *RVALID* signal indicates a read transaction).

5.2.1. Lightweight Internet Protocol

Light-Weight Internet Protocol (*lwip*) is an open-source Transmission Control Protocol (TCP)/IP stack developed from the beginning to be modular and use little Random Access Memory (RAM), so even small processors can run it. Adam Dunkels started *lwip* at the Swedish Institute of Computer Science in the early 2000s [135], and today a worldwide group of programmers maintains and further develops it. Several processor manufacturers ported it to their devices and Xilinx is no exception. During our work, we found *lwip* to be an excellent starting basis, not only due to the several protocol stacks themselves but also because it implements the drivers for the MAC and the routine to configure the PHY.

Lwip main feature to reduce RAM footprint is to avoid copying data as it moves up and down the protocol layers. For that, it defines a data structure called PBUF that can be allocated dynamically, but to improve performance, *lwip* pre-allocates PBUFs for incoming packets, only trimming the size according to the amount of incoming data. The PBUFs make *lwip* efficient, as the experimental results show, even though *lwip* design was not targeting real-time applications.

When sending a packet, the following processing sequence takes place (Fig. 5.2): after the user moves data to a PBUF and calls the routine `UDP_sendto()`, the stack adds the UDP header, selects an interface to send from, includes the IP and Ethernet headers, resolves the destination MAC address based on the destination IP address. Then, it copies the packet to an intermediate memory location. The last steps depend on the MAC type: if using Ethernet Lite, the driver copies the packet to the MAC buffer and handles control to the MAC hardware to start transmission; with the GEM or TEMAC, the driver passes the control to the MAC hardware that transfers data to the internal buffer using DMA and starts transmission.

When receiving a packet, the initial steps differ according to the MAC type: the Ethernet Lite MAC checks data integrity and immediately calls the Interrupt Service Routine (ISR); the GEM and TEMAC first transfer the packet to an intermediate position inside the main memory and, upon completion, call the ISR. Then, in all cases, the MAC driver identifies the origin of the interrupt (transmitter, receiver, or error) and calls the corresponding handler. The receiver handler copies the data to a PBUF structure, puts it into the receive queue, and exits (see Fig. 5.3a). The processing of the packet then happens outside the interrupt context, by pooling the receive queue for new data in a routine inside the infinite loop (Fig. 5.3b).

From the description above, the reader can identify that Xilinx implementation of `lwip` does copies that could be avoided if the user needs to reduce the delay. In the next section, where we show experimental results, we modify the implementation to assess the influence of these options in the whole sending and receiving delays, as well as the performance of the different MACs.

5.3. Measurements

The experimental platform was a 7020 Zynq-based board with a Microchip Fast Ethernet PHY connected to a host PC. The host PC sends packets with a total length of 64 Bytes, 256 Bytes, and 1024 Bytes, without considering the preamble, the start of frame delimiter, and the frame check sequence. The embedded processor runs the echo server demo application configured to echo UDP packets, i.e., it sends back packets received in a given port.

To measure the different delays, we designed a capture unit in VHDL to count the number of clock cycles between the positive edges of the start and stop ports. The time resolution was ten nanoseconds. When measuring the incoming delays, the inverted `Rx_dv` signal triggered the capture unit and a software-controlled output stopped it. When measuring outgoing delays, the software-controlled output triggered the start, and the positive edge of the `Tx_en` stopped it. We verified the capture unit measurement by connecting the start and stop signals to an oscilloscope. For every test run, the program

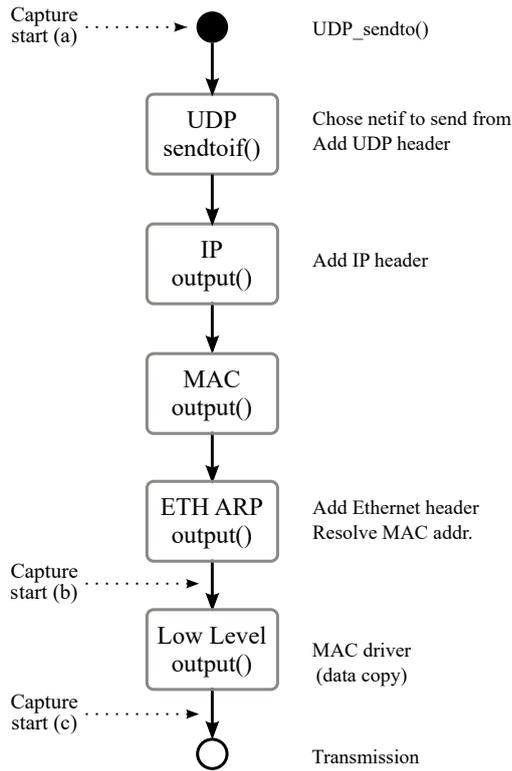


Figure 5.2: Outgoing packet sending. The dashed lines with arrow indicate the measurement points.

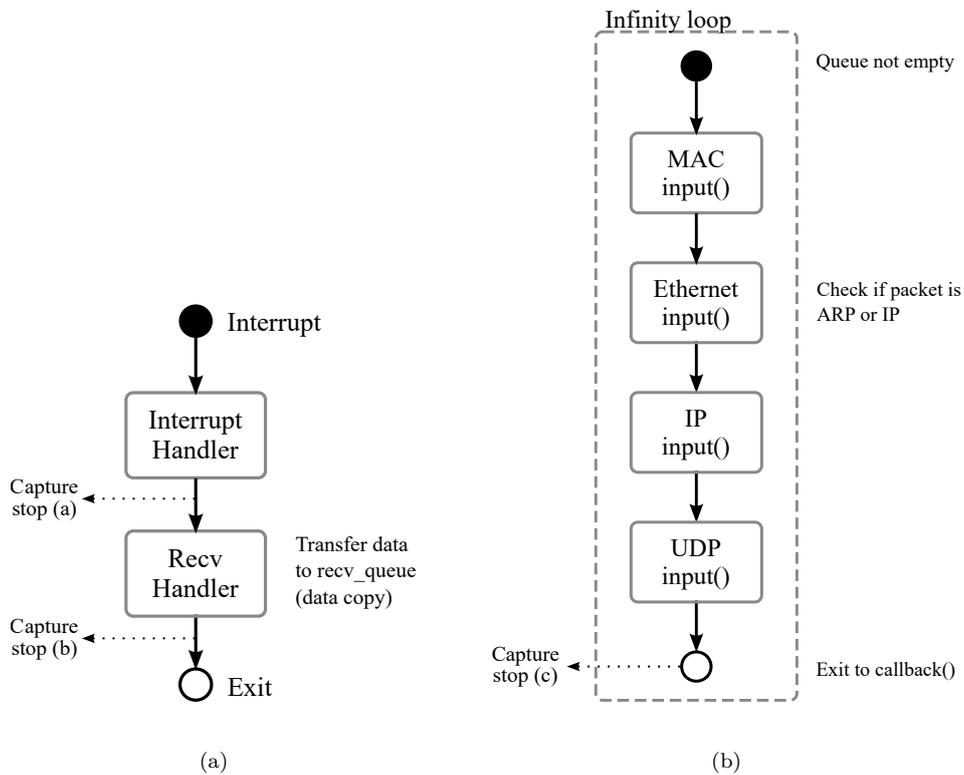


Figure 5.3: Incoming packet processing. The dashed lines with arrow indicate the measurement points.

logged 512 measurements and sent them to the host PC for processing.

For incoming packets, we measured the time delay between when the **MAC** finished receiving a packet and three points in the software: when the processor entered the interrupt service routine (Fig. 5.4); when the processor finished copying the data to the PBUF structure and queued it (Fig. 5.5); and when entering the user-defined **UDP** receiver callback (Fig. 5.6).

For outgoing packets, we measured the time delay from three points in the software to when the **MAC** starts sending data to the **PHY**: when the user calls the `UDP_sendto()` routine (Fig. 5.7); when the processor starts copying the packet from the original PBUF (Fig. 5.8); and when the software triggers the **MAC** to send the packet (Fig. 5.9). Table 5.1 summarizes the delays obtained in terms of mean value and standard deviation σ .

5.3.1. Discussion

The results show a shorter latency to enter the **ISR** when using Ethernet Lite (Fig. 5.4). This **MAC** requests an interrupt just after the packet reception, because it saves the data internally and henceforth further data movement needs the participation of the processor core. The delay around 680 ns is expected because the minimum interrupt latency of the (Cortex A9) processor core is 360 ns [136] and the **MAC** takes around 150 ns to flag the interrupt. In contrast, the **GEM** and **TEMAC** have longer latencies, which lengths depend on the packet size, because the **DMA** transfers the packet to the main memory before flagging the interrupt.

When triggering the **MAC** to send a packet, again the Ethernet Lite shows a more predictable behavior (Fig. 5.9), due to the same reason (absence of **DMA** transfer). In applications where accurate timing and low jitter are desired, e.g., when implementing a master stack or using time-triggered protocols, this is a considerable advantage of the Ethernet Lite **MAC**, moreover considering the low jitter for a pure software implementation (Table 5.1).

As explained in Section 5.2, the configuration of Ethernet Lite address range as *device memory* has a significant impact in the delays, both for receiving (Fig. 5.6) and sending data (Fig. 5.7). The results show Ethernet Lite outperforming **GEM** and **TEMAC** independently of the packet size and direction when using delay as a figure of merit. This result is surprising, as one would expect a dedicated hard peripheral, integrated to the processor, or the higher performance AXI-stream interface to be faster. The price to pay for Ethernet Lite shorter delays is the higher utilization of the CPU that is responsible for moving data to/from the **MAC**³. Another consequence of the utilization of the CPU to move data is that an implementation with Ethernet Lite is more sensible to the CPU utilization by other tasks.

³Though the programmer can use the **DMA** engines to do it

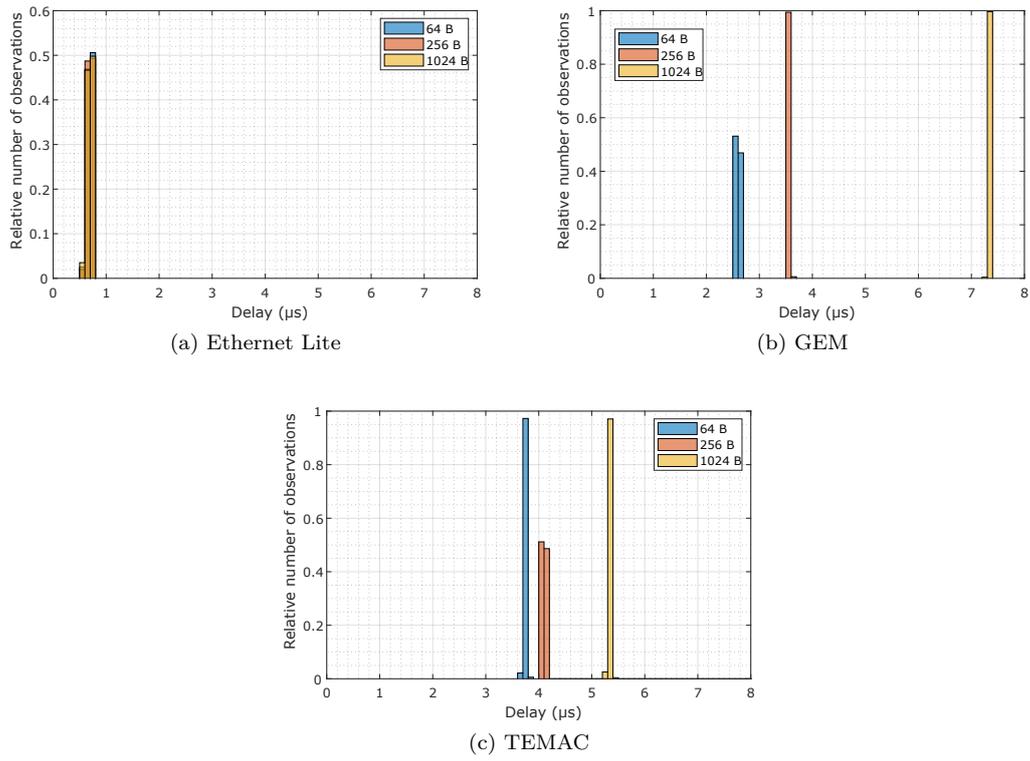


Figure 5.4: Incoming packet: delay to enter ISR after receiving packet

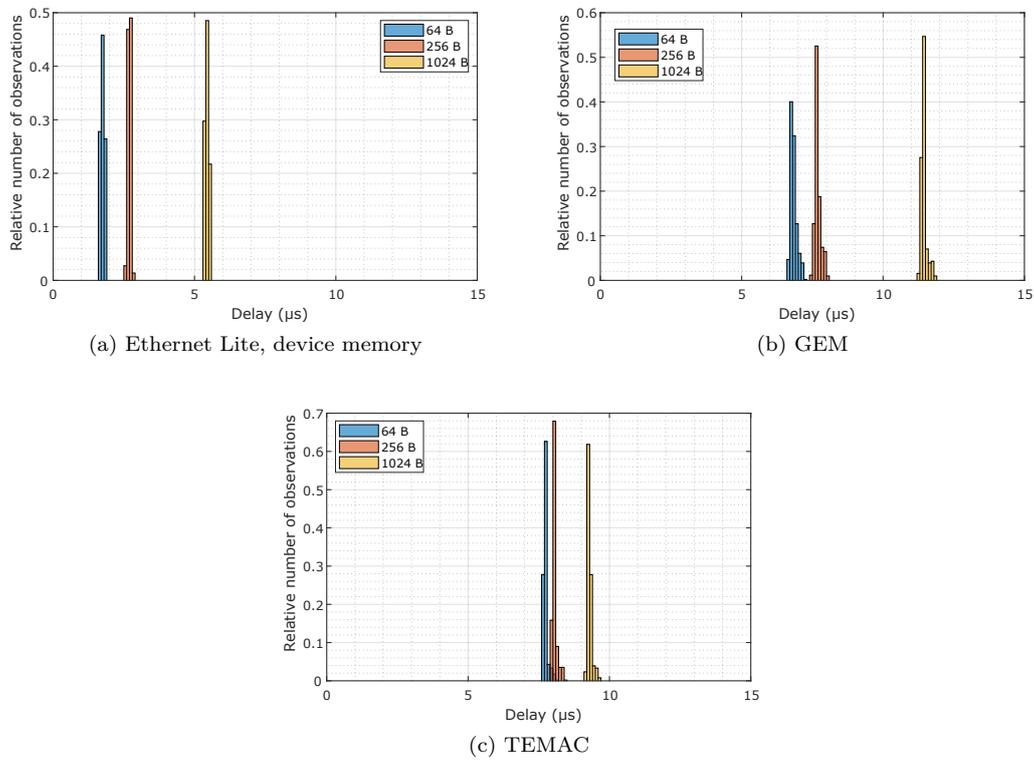


Figure 5.5: Incoming packet: delay to transfer data to PBUF after receiving packet

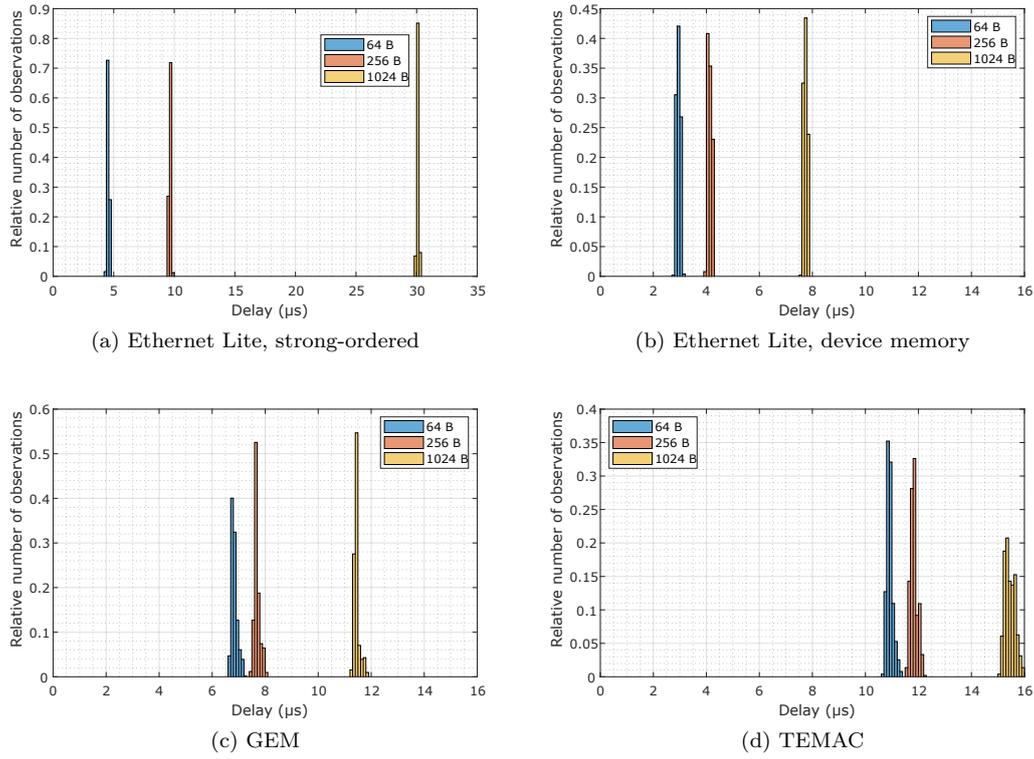


Figure 5.6: Incoming packet: delay to enter UDP callback after receiving packet

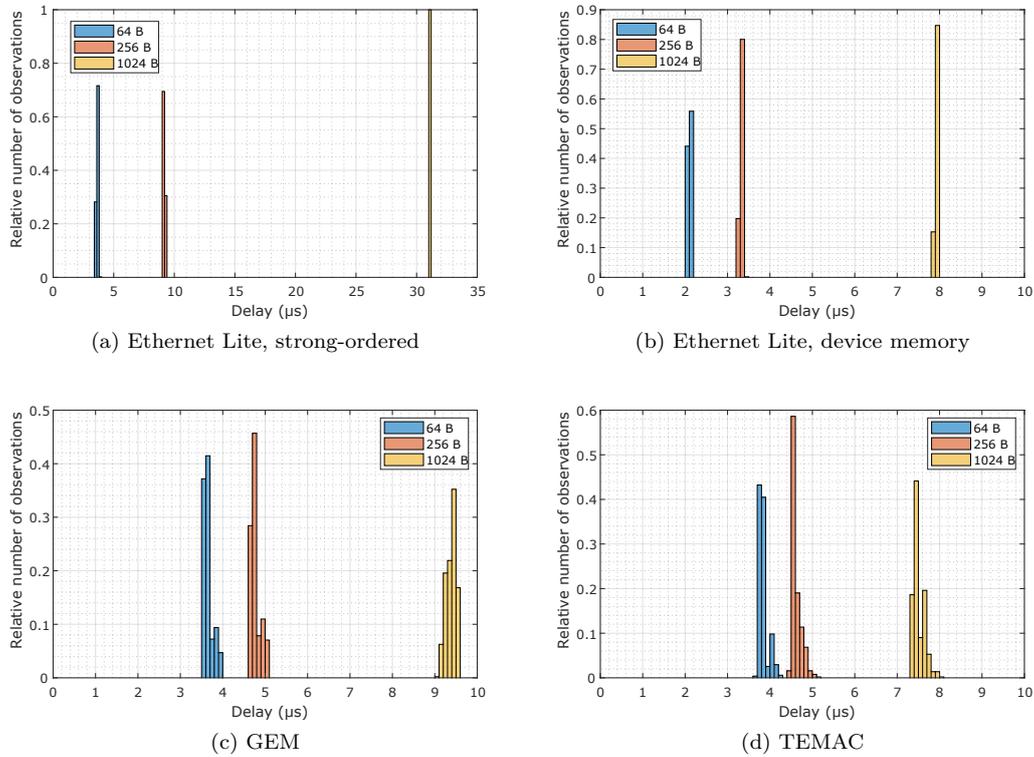


Figure 5.7: Outgoing packet: delay to start sending packet after UDP command

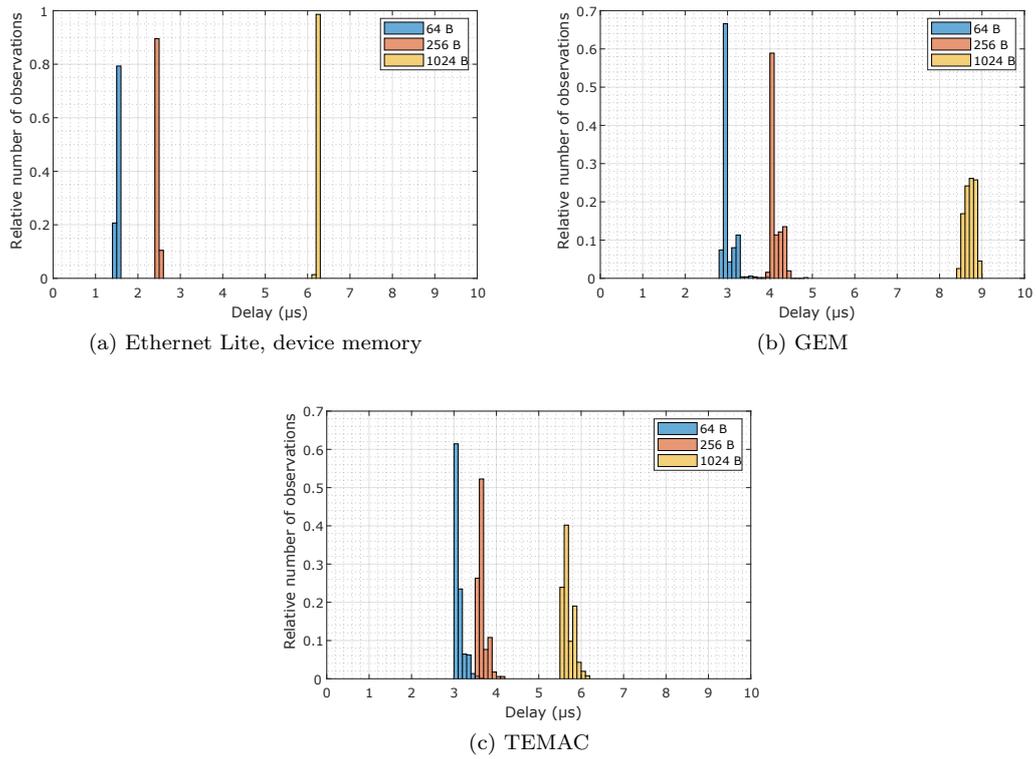


Figure 5.8: Outgoing packet: delay to send packet after starting copying from PBUFs

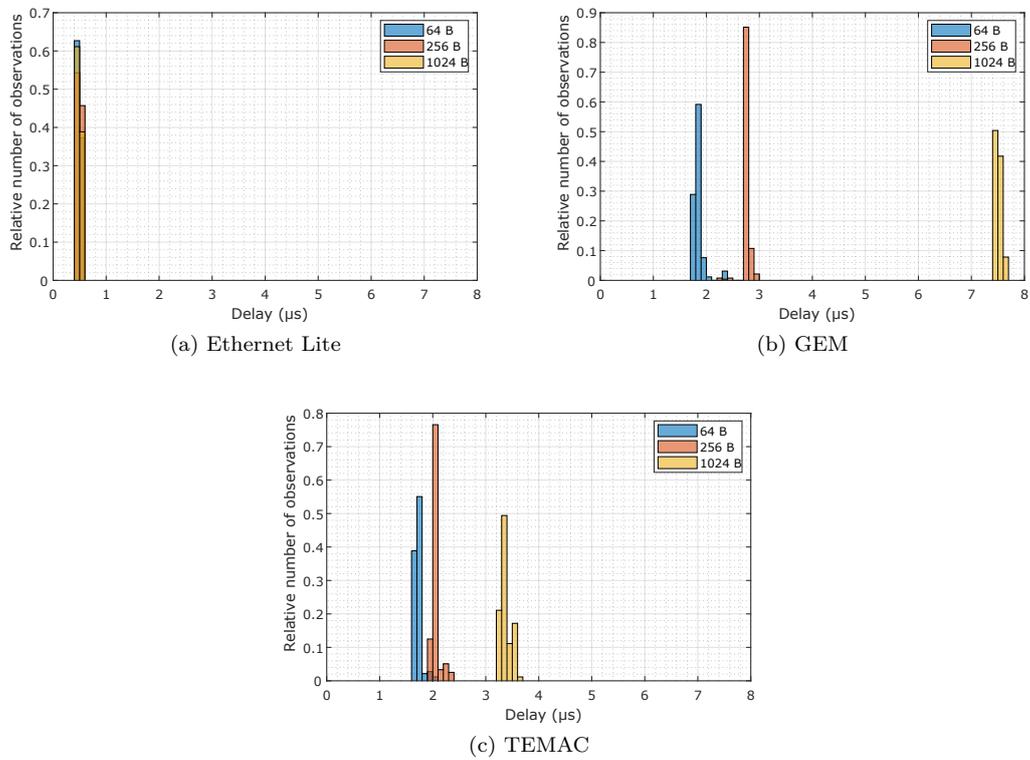


Figure 5.9: Outgoing packet: delay to send packet after triggering the MAC hardware

Table 5.1: Incoming and outgoing packet mean delay and deviation, in μs .

	Packet size	GEM		TEMAC		EL, DM		EL, SO	
		\bar{x}	3σ	\bar{x}	3σ	\bar{x}	3σ	\bar{x}	3σ
Rx PBUF	64 B	6.84	0.33	7.74	0.22	1.75	0.17	3.32	0.19
	256 B	7.69	0.32	8.06	0.25	2.70	0.18	8.22	0.18
	1024 B	11.45	0.31	9.29	0.23	5.44	0.19	27.80	0.18
Rx UDP	64 B	7.48	0.35	10.9	0.36	2.95	0.20	4.53	0.23
	256 B	8.44	0.38	11.8	0.39	4.13	0.21	9.65	0.2
	1024 B	12.09	0.34	15.5	0.57	7.74	0.20	30.09	0.21
Tx PBUF	64 B	3.01	0.45	3.11	0.31	1.52	0.05	2.74	0.06
	256 B	4.13	0.38	3.65	0.31	2.48	0.05	8.02	0.05
	1024 B	8.71	0.35	5.69	0.38	6.23	0.05	29.05	0.06
Tx UDP	64 B	3.65	0.32	3.84	0.32	2.10	0.07	3.62	0.08
	256 B	4.77	0.34	4.62	0.33	3.31	0.07	9.18	0.08
	1024 B	9.38	0.34	7.50	0.41	7.92	0.08	31.07	0.08

The experiments conducted allow quantifying the costs of using **IP** and **UDP** protocols by computing the difference between the values after copying the packet to the PBUF and the **UDP** callback (see Table 5.1). Besides the additional payload to accommodate the protocol headers, less than 2.2 μs are necessary to process the packet and call the user-defined routine.

5.4. Hardware Accelerators

When the traffic is mixed, i.e., it has real-time and best effort data, certain packets have higher priority and must be processed faster. Hardware accelerators can minimize the nodes reception delay handling packets according to their requirements.

The first hardware accelerator proposed is a *Packet Identifier* that will read all incoming packets and check their headers for some predefined characteristics, like a certain EtherType (e.g., **IP**), a specific source/destination **IP** address, and an **UDP** packet with a given port number. The receiver handler can verify the register where the accelerator puts its findings and, if the packet meets all filtering conditions, the program jumps directly to the user application. Adopting this strategy we could reduce the reception delay of 64 Bytes and 256 Bytes from 2.95 μs and 4.13 μs to 1.51 μs and 2.85 μs , respectively (Fig. 5.10). Unexpectedly, the delay of 1024 Byte packets stayed the same, i.e. 7.74 μs . We verified the reason for such result and found that the CPU implements the read transactions over the AXI bus differently, with a slower data transfer when adopting the Packet Identifier accelerator. This slower data transfer counterbalances the “jump” to the application layer as the payload increases.

Though the Packet Identifier bypasses the protocol stacks for the critical packets, it only partly avoids the data copying that is the main responsible for the reception latency.

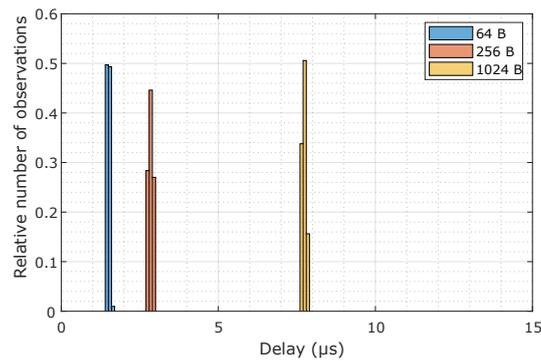


Figure 5.10: Fast-track for incoming UDP port 1026 packets.

Equally important, the data copying only starts after the **MAC** has completely received the frame and confirmed that the Frame Check Sequence is valid. To reduce the reception time to a minimum, we designed an Ethernet Direct Copy (EDC) hardware accelerator. It listens to data coming from the **PHY** and saves them directly to the main memory, to a position predetermined by the node User Application Layer.

The diagram of Fig. 5.11 represents the **EDC** structure. On the **PHY** side, the accelerator receives the clock, data valid and data signals. Once the data is valid, it waits for the start of frame delimiter to fill an asynchronous **FIFO** buffer, collecting the data into 32-bit words. The **FIFO** is responsible for the clock domain crossing of the data and, once it reaches a certain threshold signaled by the Almost Full signal, the accelerator starts writing the packet to the predefined location, either at the On-Chip Memory or the external Dual Data Rate **RAM**.

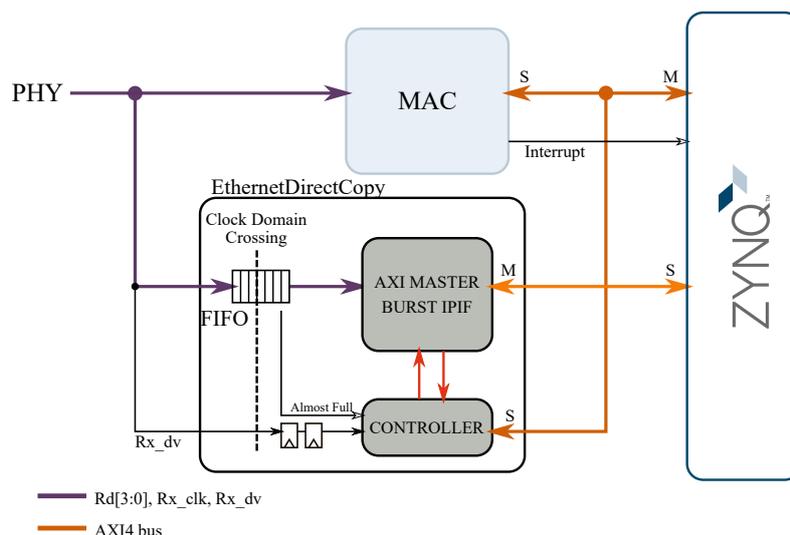


Figure 5.11: Ethernet Direct Copy hardware accelerator block diagram.

To illustrate the operation of the Ethernet Direct Copy with more details, we present signals acquired using Vivado Integrated Logic Analyzer in Fig. 5.12.

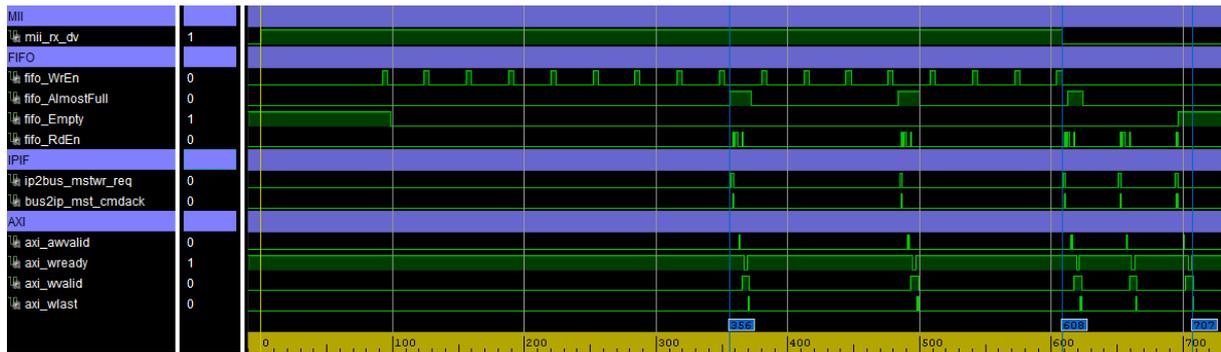


Figure 5.12: Capture showing EDC internal signals while receiving a packet with 64 Bytes. The x -axis is in samples, and the sampling period equal to 10 ns.

The rising edge of mii_rx_dv marks the start of packet reception ($t = 0$). The accelerator monitors the incoming data and waits for the start of frame delimiter to collect the packet into a 32-bit word and write it to the *FIFO* ($t = 92$), which happens every time the signal $fifo_WrEn$ is high. Once the *FIFO* reaches the almost full level ($fifo_AlmostFull$, $t = 356$), the controller triggers a write burst (16 Bytes, in this case) using the AXI Master Burst IPIF. This happens when $ip2bus_mstwr_req$ goes high and, after some handshake, the AXI master reads data from the *FIFO* ($fifo_RdEn$) and writes to the memory. Note that it has a unique address phase (the master assigns an address when $axi_awvalid$ is high) and writes several words to the memory (when axi_wvalid and axi_wready are simultaneously high).

As the IPIF reads data from the *FIFO* faster than the *PHY* writes, it comes out of the almost full condition. While data is arriving, the *FIFO* gets filled again, and triggers write transactions a number of times. Once the reception of the packet completes (mii_dv falling edge, $t = 608$), the controller performs the last bursts to empty the *FIFO*, finishing at $t = 707$.

Simultaneous to the accelerator operation, the *MAC* receives the packet and stores it internally. If the packet received is valid, the *MAC* flags an interrupt and the processor switches context to service it. At the entrance of the Interrupt Service Routine, the processor consults the Packet Identifier accelerator. If the packet is of the high-priority type, the processor disables the *EDC* accelerator, to prevent the *EDC* of overwriting the data while the CPU is processing it, and calls the user callback function. If the packet is invalid, the *MAC* simply will not flag an interrupt and the node will wait for the next packet arrival.

Note that the accelerator, unlike all the *MACs* investigated, transfers data to the memory during the packet reception. As a consequence, the time to complete the transfer and to enter the user callback is independent of the packet length (Fig. 5.13).

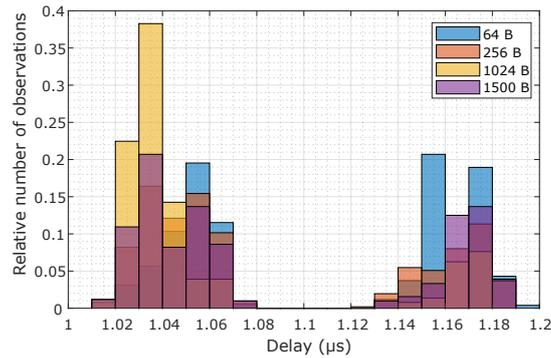


Figure 5.13: Delay to enter UDP callback after receiving packet using Ethernet Direct Copy accelerator. The bar fillings are transparent to show that the measurements are overlapping.

5.5. Conclusions

In this chapter, we investigated the delay in receiving and sending Ethernet packets with different [MAC](#) architectures and communication stack implementation details. We ran extensive tests and gathered several results to help the designer estimate more accurately the total end-to-end latency when using Ethernet. The results show the effective bandwidth of the [MAC](#) interface to the processor memory as a critical aspect. Other critical aspects are minimizing copying data and pre-allocating memory.

The platform chosen to do this investigation was a Zynq-based board using the Lightweight [IP](#) stack. First, we described the different [MACs](#) available in the selected platform and the packet processing implementation. Second, we measured the delays between the reception of packets at the [MAC](#) and selected points in the software to characterize the delay when adopting different strategies. We did the same for outgoing packets but measuring the delay from selected points in the software to when the [MAC](#) effectively starts sending data.

The results show Ethernet Lite to be faster than the hard [MAC](#) and the TEMAC for any payload and both incoming and outgoing packets. Short packets can be received and sent using the full [UDP/IP](#) stack within 3.3 μs and 2.1 μs , respectively.

We discussed how to reduce the delay by simplifying the protocol or adopting a Packet Identifier accelerator to fast-track high priority data to the application, avoiding unnecessary copies and the delay of the protocol stack. As this strategy lefts untouched an important source of delay, the fact that the [MAC](#) waits for the packet arrival to move it to the main memory, we presented the concept, implementation details, and experimental results of Ethernet Direct Copy accelerator that minimizes the reception delay of Ethernet packets. The main innovation of the proposed solution is to transfer the packet data directly to the main memory of the processor while it is being received, thus removing the delay dependence on packet length. The experimental results show the proposed solution to be 6.4 μs and 11 μs faster than [GEM](#) (likely the most popular choice) when the packet

size is 64 Bytes and 1024 Bytes, respectively.

The [EDC](#) preserves Ethernet layer structure and enables faster cycle times and lower end-to-end latencies that are important for high-performance applications while keeping the implementation of higher layers in software. As real-time Ethernet protocols move to higher data rates, the minimization of the internal delays becomes more critical, so the higher performance brought using the accelerator gains relevance.

Chapter 6

Model-Based Compensation for Network Delays

In the previous chapters, we explained the advantages of introducing an internal network to Modular Multilevel Converters. Among the advantages are the possibility to adopt other control architecture topologies than the traditional star and the opportunity to partly decentralize the control. On the other hand, the network inevitably increases the loop delay and imposes some constraints to the selection of the control tasks period (see Section 3.4). To limit the influence of this delay, we have proposed two Ethernet-based protocols that reach Minimum Cycle Times close to the lowest possible for ring networks.

In this chapter, we approach the influence of the network in the loop delay from a different side. A model-based predictor compensates for the latency of the network, so the control tolerates longer delays, achieving higher flexibility in the choice of the protocol and communication technology. An alternative benefit is a better dynamic performance by the use of a higher sampling rate and controller gains.

The use of a plant model to compensate for the loop delay dates back to 1957 when O.J. Smith proposed what today is known as the Smith predictor [137]. Recently, Cortes et al. [138] adopted a similar approach to compensate for the delay introduced by Model Predictive Control calculations. Nevertheless, this strategy has not yet been applied to network controlled Modular Multilevel Converters (MMCs).

In the next Sections, we discuss the influence of loop delay and sampling periods in closed-loop control. Then we model the proposed strategy in mathematical terms and investigate the effect of parameter variations in the closed-loop response. At the end of the chapter, we show MATLAB/Simulink simulations and experimental results to validate the strategy.

6.1. Influence of Sampling Rate and Delay on Control Performance

Sampling is a fundamental property of discrete-controlled systems. Its rate influences the poles and zeros of the discretized plant, hence it affects the controller design and closed-loop response [139]. In like manner, loop delays are a common phenomenon in several control applications and are almost inevitable in networked controlled systems [140]. Their presence imposes strict limitations on achievable feedback performance and has often a “destabilizing effect” [140].

The frequency-domain analysis is the preferred way of dealing with systems that have loop delays [137, 139, 140]. To visualize their effect, consider the open-loop Bode diagrams of the system (Fig. 6.1) with a Proportional-Integral (PI) controller tuned for 3 ms settling time t_s using the pole placement method for the converter with the parameters listed in Table 6.1. Each sample delay introduces a linearly increasing lag with frequency [140] (at the Nyquist frequency, the phase is -360° , -540° , -720° for delays of 1, 2 or 3 samples respectively), but the magnitude remains equal to the case without delay (Fig. 6.1a). A discrete-time controller has a delay of at least one sample. With two samples delay, the system has a narrow phase margin¹, and with three or more samples delay it becomes unstable (negative phase margin).

The Bode diagram of Fig. 6.1b shows the influence of the sampling frequency on the phase margin and crossover frequency² when the plant has a single sample delay. For the case analyzed, the three shortest sampling periods result in positive phase margins of 52° , 39° , and 16° , respectively, i.e., only them result in a stable closed-loop system.

Still, the control of the converter can be realized with lower sampling frequencies or longer delays, but the designer will need to reduce the controller gains to bring the system back to stability [137] (e.g., in Ziegler-Nichols method, the integral gain is inversely proportional to the deadtime [139]) and, consequently, slow down its response.

6.2. Proposed Estimation Algorithm

In this section, we explain the model-based predictor and how it compensates for loop delays. It works as follows: a model of the plant uses the measured state vector and the known future plant inputs to estimate what will be the state vector n samples ahead. The central unit feeds the estimated states back to the controller that calculates the plant inputs and places them into the communication queue. After n samples, the actuators apply them to the plant.

¹Distance between the system phase and -180° at the crossover frequency (see footnote 2).

²Frequency in which the system magnitude is equal to 0 dB.

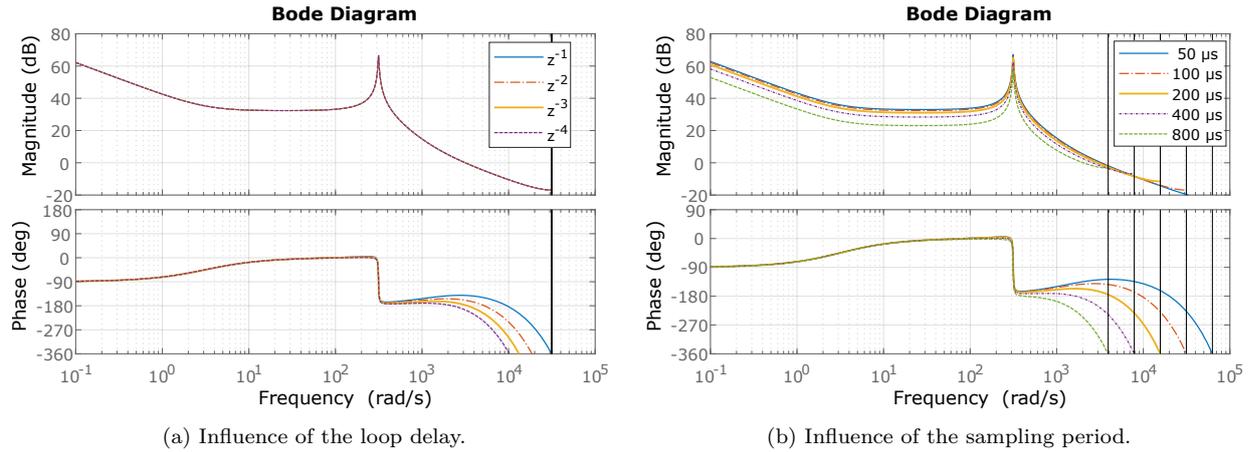


Figure 6.1: Open-loop Bode diagram of the plant with controller.

Consider the difference state-space equations that represent a discretized linear time-invariant model (6.1) and (6.2), where \mathbf{x}_k and \mathbf{y}_k are the state and output vectors in instant k , respectively, $\mathbf{u}_{k|k-n}$ is the plant input vector in instant k calculated in $k - n$, and \mathbf{A} , \mathbf{B} , and \mathbf{C} are the state, input, and output matrices with the necessary dimensions.

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_{k|k-n}, \quad (6.1)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k, \quad (6.2)$$

In a state-feedback control law, the plant input would be $\mathbf{u}_k = \mathbf{K}\mathbf{x}_k$, where \mathbf{K} is the gain matrix that can be designed, for example, using Linear Quadratic Gaussian control or pole placement [139]. To compensate the loop delay, we propose to predict the system state $\hat{\mathbf{x}}_{k+n}$ and use it for the calculation of the plant input (Fig. 6.2), as in (6.3).

$$\mathbf{u}_{k+n|k} = \mathbf{K}\hat{\mathbf{x}}_{k+n}. \quad (6.3)$$

Henceforth, we will drop the notation $\mathbf{u}_{k+n|k}$ and use only \mathbf{u}_{k+n} , because the network has a constant delay, guaranteed by design as explained in Section 3.4, and the controller always calculates the plant input n samples in advance.

We can express the estimated states $\hat{\mathbf{x}}$ in instants $(k+1)$ – $(k+n)$ with (6.4)–(6.6), respectively, where matrices $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$ correspond to the plant model. Alternatively, in a recursive manner, we could use (6.7).

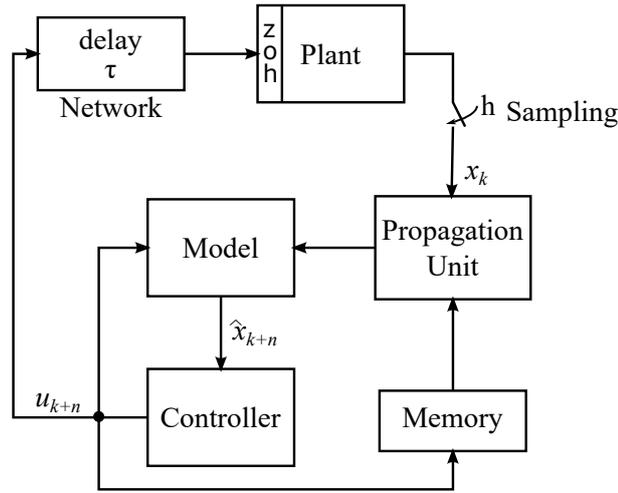


Figure 6.2: Configuration of the proposed control with long network delays.

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{A}}\mathbf{x}_k + \hat{\mathbf{B}}\mathbf{u}_k, \quad (6.4)$$

$$\hat{\mathbf{x}}_{k+2} = \hat{\mathbf{A}}\hat{\mathbf{x}}_{k+1} + \hat{\mathbf{B}}\mathbf{u}_{k+1}, \quad (6.5)$$

$$\vdots$$

$$\hat{\mathbf{x}}_{k+n} = \hat{\mathbf{A}}\hat{\mathbf{x}}_{k+n-1} + \hat{\mathbf{B}}\mathbf{u}_{k+n-1}, \quad (6.6)$$

$$\hat{\mathbf{x}}_{k+n} = \hat{\mathbf{A}}^n \mathbf{x}_k + \sum_{j=1}^n \hat{\mathbf{A}}^{j-1} \hat{\mathbf{B}} \mathbf{u}_{k+n-j}, \quad (6.7)$$

If we substitute (6.7) in (6.3), the plant input vector is expressed by (6.8).

$$\mathbf{u}_{k+n} = \mathbf{K}\hat{\mathbf{A}}^n \mathbf{x}_k + \sum_{j=1}^n \mathbf{K}\hat{\mathbf{A}}^{j-1} \hat{\mathbf{B}} \mathbf{u}_{k+n-j}. \quad (6.8)$$

The controller calculates (6.8) in every cycle and puts the results in the communication queue. A new measurement is only necessary to calculate the first term of (6.8), $\mathbf{K}\hat{\mathbf{A}}^n \mathbf{x}_k$. The controller can evaluate all the other terms as soon as the next plant input vector is available, as $\mathbf{u}_{k+n} \rightarrow \mathbf{u}_{k+n-1}, \dots, \mathbf{u}_{k+1} \rightarrow \mathbf{u}_k$ in the next sample. Note that the matrices $\mathbf{K}\hat{\mathbf{A}}^n$, $\mathbf{K}\hat{\mathbf{B}}$ to $\mathbf{K}\hat{\mathbf{A}}^{n-1}\hat{\mathbf{B}}$ are constants that can be stored in the memory at compilation time, so this method is not hard on the processing system.

As an illustration, we represented the closed-loop system with two actuation delays, a PI controller, and the model-based predictor in the block diagram of Fig. 6.3. In this system, the augmented state-space vector is $[\mathbf{x} \ \mathbf{u}_k \ \mathbf{u}_{k+1} \ \boldsymbol{\eta}]^T$ to include the plant input \mathbf{u}_k , the future plant input \mathbf{u}_{k+1} , and the controller integral part $\boldsymbol{\eta}$. The output of the PI controller will be equal to the plant input two samples ahead ($n = 2$) and (6.8) becomes $\mathbf{u}_{k+2} = \mathbf{K}\hat{\mathbf{A}}^2 \mathbf{x}_k + \mathbf{K}\hat{\mathbf{B}}\mathbf{u}_{k+1} + \mathbf{K}\hat{\mathbf{A}}\hat{\mathbf{B}}\mathbf{u}_k$. From the block diagram, it is possible

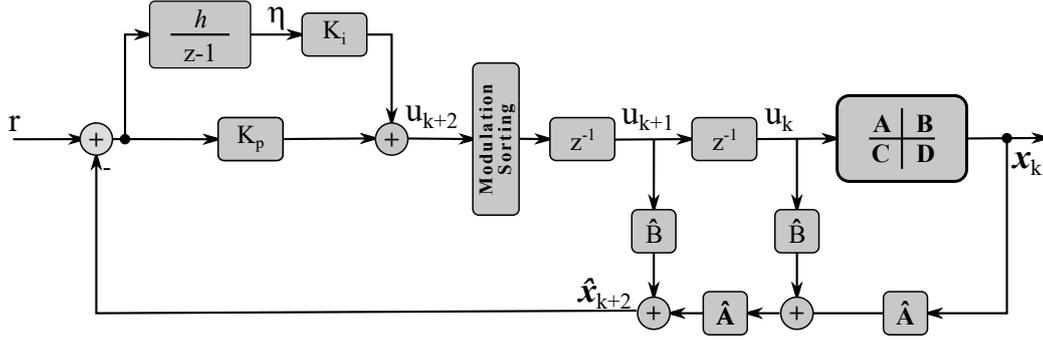


Figure 6.3: Block diagram of a closed-loop system with two loop delays, a PI controller, and the model-based predictor.

to deduce the state-space closed-loop response (6.9) in the format of (6.1), where \mathbf{r} is the reference vector, \mathbf{I} is the identity matrix of proper dimensions, h is the sampling period, and k_p and k_i are the proportional and integral gains, respectively.

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{u}_k \\ \mathbf{u}_{k+1} \\ \boldsymbol{\eta} \end{bmatrix}_{k+1} = \begin{bmatrix} \mathbf{A} & \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -k_p \hat{\mathbf{A}}^2 & -k_p \hat{\mathbf{A}} \hat{\mathbf{B}} & -k_p \hat{\mathbf{B}} & k_i \mathbf{I} \\ -\hat{\mathbf{A}}^2 h & -\hat{\mathbf{A}} \hat{\mathbf{B}} h & -\hat{\mathbf{B}} h & \mathbf{I} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x} \\ \mathbf{u}_k \\ \mathbf{u}_{k+1} \\ \boldsymbol{\eta} \end{bmatrix}_k + \begin{bmatrix} \mathbf{0} \\ 0 \\ k_p \mathbf{I} \\ \mathbf{I} h \end{bmatrix} \cdot \mathbf{r} \quad (6.9)$$

6.2.1. Estimation of Circulating Current

Though (6.7) is valid for both the load and the circulating current state vectors, the correct estimation of the latter is difficult. The reason is the appearance of parasitic components due to the difference between the divisors of (2.16) and the summed voltage of the inserted capacitors (see subchapter 3.5 of [59]). Because the voltage drop in L and R are small ($u_l + u_u \simeq u_{dc}$), even parasitic components with low magnitude produce significant errors between the reference and the applied internal voltage $u_{circ,abc}^*$. In Fig. 6.4 we show a simulation of the $u_{circ,a}$ (orange waveform) and its reference (blue waveform) when the circulating current control is enabled ($t > 0.4$ s) and disabled ($t < 0.4$ s). In both cases, the difference between reference and measurement is remarkable, though the control still reduces the circulating current RMS value (yellow waveform) when enabled.

Two solutions to this problem are possible. The first one is to use the arm voltage measurements as u_k in (6.7) to estimate the circulating current state vector that is fed back, but in this case, it is only possible to predict a single sample ahead. The second one is to limit the circulating current passively with a larger arm inductance. In our experiments, we adopted the first strategy, even when the network delay is higher than one sample. In this case, the resonant controller becomes unstable due to the partial compensation, but the repetitive controller could still reduce the circulating current.

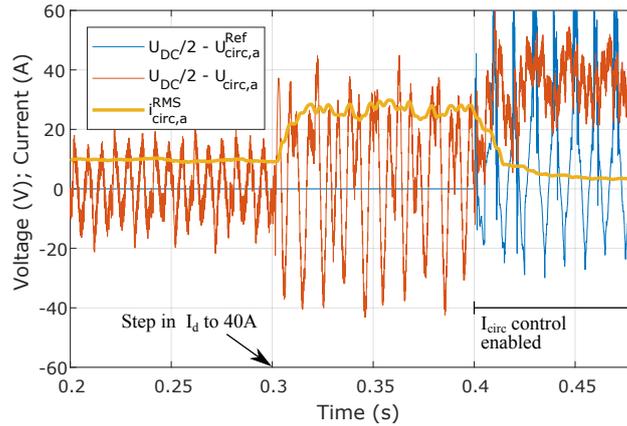


Figure 6.4: Mismatch between u_{circ} and its reference. As a consequence, an estimation of i_{circ} based on the references delivers poor results.

6.2.2. Modulation and Capacitor Balancing

Thus far, we only considered the calculation of the voltage references by the load and circulating current controllers. The references need to be translated into commands to the power switches and the capacitor voltages need to be balanced, as already explained.

The controller outputs either one reference per cell or single ones per arm. In the first case, the control is centralized, and the central unit is responsible for the modulation and capacitor balancing. Due to the network, the central controller receives data delayed by a few samples. We tested if the use of delayed capacitor voltage measurements could still allow an effective balancing when using the sorting algorithm. For that, we ran several simulations introducing delays in the capacitor voltage measurements that are fed back to balancing. The results show that the maximum deviation to the average capacitor voltage increases with the number of delayed samples, but the deviation stays as a small fraction of the ripple (Fig. 6.5). Therefore, the standard sorting algorithm is an effective balancing strategy when controlling a network controlled MMC.

In case the control outputs single setpoints per arm, as adopted in [45] using phase-shifted carrier Pulse Width Modulation (PWM), it configures a partly distributed control strategy. The central controller runs the current and outer control loops, while the cells execute the modulation and balancing. In such a case, the network does not affect the capacitor balancing because it is run locally at the cells.

6.3. Closed-Loop Stability

After the discretization of the MMC model (2.11) to the format in (6.1), we augment the state space equation to include the PI controller with the model prediction (6.3) and the calculation delay, resulting in (6.10) and (6.11), where $\mathbf{x} = [\boldsymbol{\eta} \quad \mathbf{x}' \quad \mathbf{x}'_r]^T$ is the new

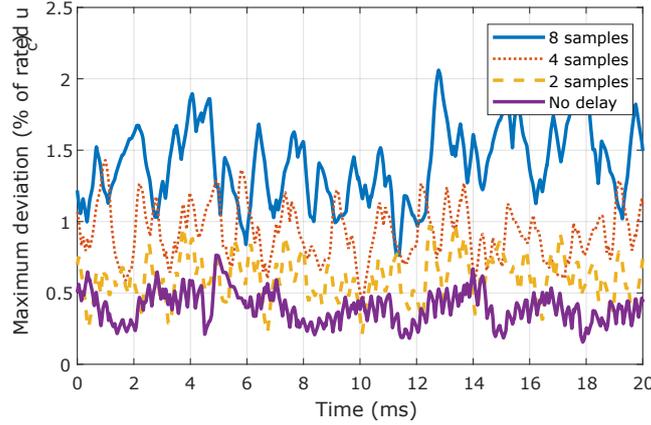


Figure 6.5: Maximum deviation to the average capacitor when the sorting uses measurements with and without delays. Capacitor rated voltage of 170V and sampling period of 100 μ s.

state vector, $\boldsymbol{\eta}$ is the controller integral part, \boldsymbol{x}'_r is the delayed state vector, k_p and k_i are the proportional and integral gains.

$$\begin{bmatrix} \boldsymbol{\eta} \\ \boldsymbol{x}' \\ \boldsymbol{x}'_r \end{bmatrix}_{k+1} = \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} & -\mathbf{I}.h \\ \mathbf{0} & \mathbf{A}_d & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \boldsymbol{\eta} \\ \boldsymbol{x}'_k \\ \boldsymbol{x}'_r \end{bmatrix}_k + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{B}_d \\ \mathbf{0} \end{bmatrix}}_{\mathbf{B}} \boldsymbol{u}_k + \begin{bmatrix} \mathbf{I}.h \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \boldsymbol{i}_{dq}^* \quad (6.10)$$

$$\boldsymbol{u}_k = \underbrace{\begin{bmatrix} k_i \mathbf{I} & \mathbf{0} & -k_p \mathbf{I} \end{bmatrix}}_{\mathbf{K}} \hat{\boldsymbol{x}}_k + k_p \mathbf{I} \cdot \boldsymbol{i}_{dq}^* \quad (6.11)$$

The closed-loop system modeled by (6.10) and (6.11) with actuation delay is equivalent to the state feedback case with large delays analyzed in [141]. In such systems, the network induced delay τ is larger than the update time h' , where both h' and τ are measured in number of samples and are therefore integers. The augmented system $\boldsymbol{z} = \begin{bmatrix} \boldsymbol{x}^T & \check{\boldsymbol{e}}^T & \hat{\boldsymbol{e}}^T & \check{\boldsymbol{e}}_1^T & \cdots & \check{\boldsymbol{e}}_n^T \end{bmatrix}$ is globally exponentially stable if and only if the eigenvalues of Σ (6.12) are inside the unity circle [141], where $\check{\boldsymbol{e}}_k = \boldsymbol{x}_k - \check{\boldsymbol{x}}_k^{n+1}$, $\hat{\boldsymbol{e}}_k = \check{\boldsymbol{x}}_k^1 - \hat{\boldsymbol{x}}_k$, $\check{\boldsymbol{e}}_k^i = \check{\boldsymbol{x}}_k^{i+1} - \check{\boldsymbol{x}}_k^i$, for $i = 1, 2, \dots, n$, $\tilde{\mathbf{A}} = \mathbf{A} - \hat{\mathbf{A}}$, $\tilde{\mathbf{B}} = \mathbf{B} - \hat{\mathbf{B}}$, $\check{\boldsymbol{x}}_k^i$ are the $n-1$ propagation state variables updated using the expression $\check{\boldsymbol{x}}_{k+1}^i = \hat{\mathbf{A}} \check{\boldsymbol{x}}_k^i + \hat{\mathbf{B}} \boldsymbol{u}_k$, and the matrix $(n+2) \times (n+2)$ Λ is expressed by (6.13).

$$\Sigma = \begin{bmatrix} I & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & I & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & I & \dots & 0 & 0 \\ \vdots & & & \ddots & & \vdots & \\ 0 & 0 & 0 & 0 & \dots & I & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & I \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \end{bmatrix} A^{\tau-(n-1)h'} \begin{bmatrix} I & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & I & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & I & & 0 & 0 \\ \vdots & & & \ddots & & \vdots & \\ 0 & 0 & 0 & 0 & \dots & I & 0 \\ 0 & I & 0 & 0 & \dots & 0 & I \end{bmatrix} A^{nh'-\tau} \quad (6.12)$$

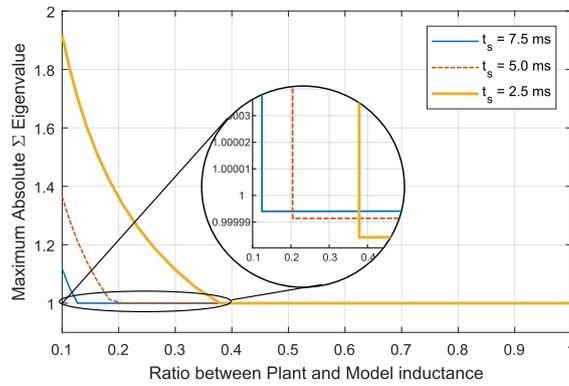
$$A = \begin{bmatrix} \mathbf{A} + \mathbf{BK} & -\mathbf{BK} & -\mathbf{BK} & -\mathbf{BK} & \dots & -\mathbf{BK} \\ \tilde{\mathbf{A}} + \tilde{\mathbf{B}}\mathbf{K} & \hat{\mathbf{A}} - \tilde{\mathbf{B}}\mathbf{K} & -\tilde{\mathbf{B}}\mathbf{K} & -\tilde{\mathbf{B}}\mathbf{K} & \dots & -\tilde{\mathbf{B}}\mathbf{K} \\ 0 & 0 & \hat{\mathbf{A}} & 0 & \dots & 0 \\ 0 & 0 & 0 & \hat{\mathbf{A}} & \dots & 0 \\ \vdots & & & & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \hat{\mathbf{A}} \end{bmatrix}. \quad (6.13)$$

6.4. Parameters Sensitivity

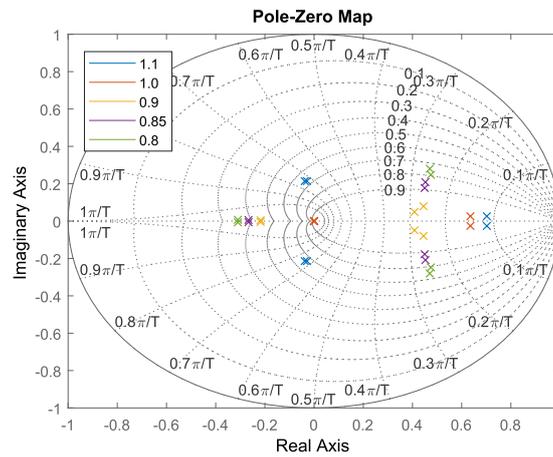
The proposed estimation algorithm uses a model of the plant. Naturally, using models raises concerns over the system performance and stability as the model and plant deviate from each other.

It is possible to verify the system stability, study mismatches between plant and model or the effect of the controller tuning by calculating the Eigenvalues of (6.12). To exemplify, we discretized the plant (2.11) using a Zero-Order Hold and considered the closed-loop system with a delay of two samples (Fig. 6.3). The maximum absolute Eigenvalues of the matrix Σ for different plant/model inductance ratios (L_{eq}/\hat{L}_{eq}) and controllers gains, represented here as the settling time used in the controller design, are shown in Fig. 6.6a. As expected, a faster controller is more sensitive to models mismatch, but the closed-loop plant is stable with all three controllers as long as the plant-to-model inductance ratio is greater than 0.38. Note that high voltage MMCs typically use dry-type air-core reactors [59], so errors of this magnitude are unlikely. Furthermore, they are associated with expensive projects that have long implementation periods; hence a good characterization of the filter components is feasible.

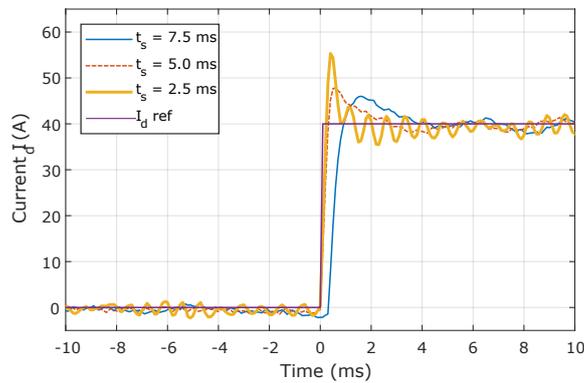
A second method for the verification of the sensitivity to parameters variation is to plot the root locus of the closed-loop plant (6.9), as in Fig. 6.6b for the controller with $t_s = 2.5$ ms. Finally, we show simulation results of a step response for three controllers when the equivalent plant inductance is 0.5 of the model value (Fig. 6.6c).



(a) Study of sensibility to the ratio between plant and model inductance with different controller gains (t_s is the controller settling time). The system is stable if and only if all the eigenvalues of Σ are within the unitary circle [142].



(b) Pole-Zero map of closed-loop plant with controller tuned for $t_s = 2.5$ ms and several plant to model inductance ratios.



(c) Step response with a plant to model inductance ratio of 0.5.

Figure 6.6: Stability analysis and influence of model error in the system closed-loop poles and zeros.

6.5. Simulation and Experimental Results

We simulated a Static Synchronous Compensator (STATCOM) using a detailed equivalent circuit model (type 4 [35]) in MATLAB/Simulink as proposed in [143]. The simulation time with this modeling is about the same as with a full detailed model (type 2 [35]) for small converters, but it simplifies the parametrization of the number of cells and reduces the simulation time of larger converters dramatically.

The experimental results were taken using a reduced scale prototype (Fig. 6.7) also working as a STATCOM. It is a quasi-industrial three-phase converter, rated to 50 kVA/400 V, with five half-bridge cells per arm (see Table 6.1). The control is based in Xilinx Zynq System-on-Chip (SoC) and can simultaneously measure 43 analog signals, command 64 power switches through fiber optics, and receive 32 fault signals (refer to [144] for a detailed description of the prototype). The Programmable Logic (Field Programmable Gate Array (FPGA)) has the following tasks: measure the grid voltage angle with a PLL; acquire the data from the Analog/Digital converters; do the modulation that controls the power switches; keep the capacitor voltages balanced (sorting algorithm). The Processing System (ARM cores) has two tasks: execute the control loops and run the server of the data acquisition and supervision system. In our implementation, the time necessary for data conversion is 12.5 μ s; for control calculations is 6.5 μ s; for system protection is 4.8 μ s; for all the tasks is 27 μ s.

Table 6.1: Simulation and prototype parameters.

Parameter	Value	Parameter	Value
Rated power	50 kVA	Arm inductor (L)	0.5 mH
Grid line voltage	400 V	Arm resistance (R)	1 m Ω
Cells/arm (N)	5	Filter inductor (L_f)	5 mH
Cell capacitor (C)	2.2 mF	Filter resistance (R_f)	14 m Ω
Sampling period (h)	100 μ s	Grid inductance	0.4 mH
Carrier frequency	750 Hz	DC voltage	750 V

The control architecture of this converter is centralized, i.e., a single hardware unit controls all the cells. For the validation of the proposed method and the sake of simplicity, we emulated the digital communication network by adding a delay of one sample between the output of the control and the PWM. From the control point of view, this is an accurate emulation of a distributed implementation based on a real-time network that guarantees meeting its deadlines. In total, the system experiences two samples delay between sampling and actuation, one owing to the digital nature of the controller and the other to the delay introduced by the network. Fig. 6.8 represents the complete control structure emulated in the reduced scale prototype. Note that the predictor of the circulating current needs additional sensors to measure the arm voltages, as explained before.

We present the simulation and experimental results in Fig. 6.9 under two different



Figure 6.7: Modular Multilevel Converter prototype with five cells/arm.

scenarios: (a) with and (b) without the model-based latency compensation. The blue and the dashed orange waveforms are the simulated and experimental response, respectively, to a reference step in current producing reactive power I_d . In a [STATCOM](#), it makes no sense doing steps in the reference of the active current I_q , because the active current compensates the converter power losses.

As expected, the system has a narrower phase margin and the closed-loop response is less damped when the latency is uncompensated, so the overshoot is higher and the settling time is longer. On the other hand, the step response with the compensation has less overshoot and is faster, demonstrating the benefits of the proposed approach.

The simulated and experimental responses are almost identical (see Fig. 6.9), validating both the model and the model-based compensation strategy. The only tweak needed in the simulation was to include the grid fifth and seventh harmonic, with the same magnitude measured, and to adjust their relative phase to the fundamental.

Additionally, in Fig. 6.10a and Fig. 6.10b, we show, respectively, the reactive and the grid currents when the controller commands a negative current step in a system with a delay of three samples and which uses the model-based predictor. Note that the response has the same settling time and overshoot as before. Note that we do not consider this a real use-case of the proposed method because the designers have the freedom to choose the sampling period and the communication network deployed; we believe that it is better to have a higher sampling period than a loop delay higher than two samples. Despite this consideration, the system response has the same settling time and overshoot as when the network adds a single delay, but without the prediction, the system becomes unstable.

In our experimental setup, the control balances the cell voltages employing the sorting algorithm. For this, it receives from each cell its voltage, organizes the cells in ascending

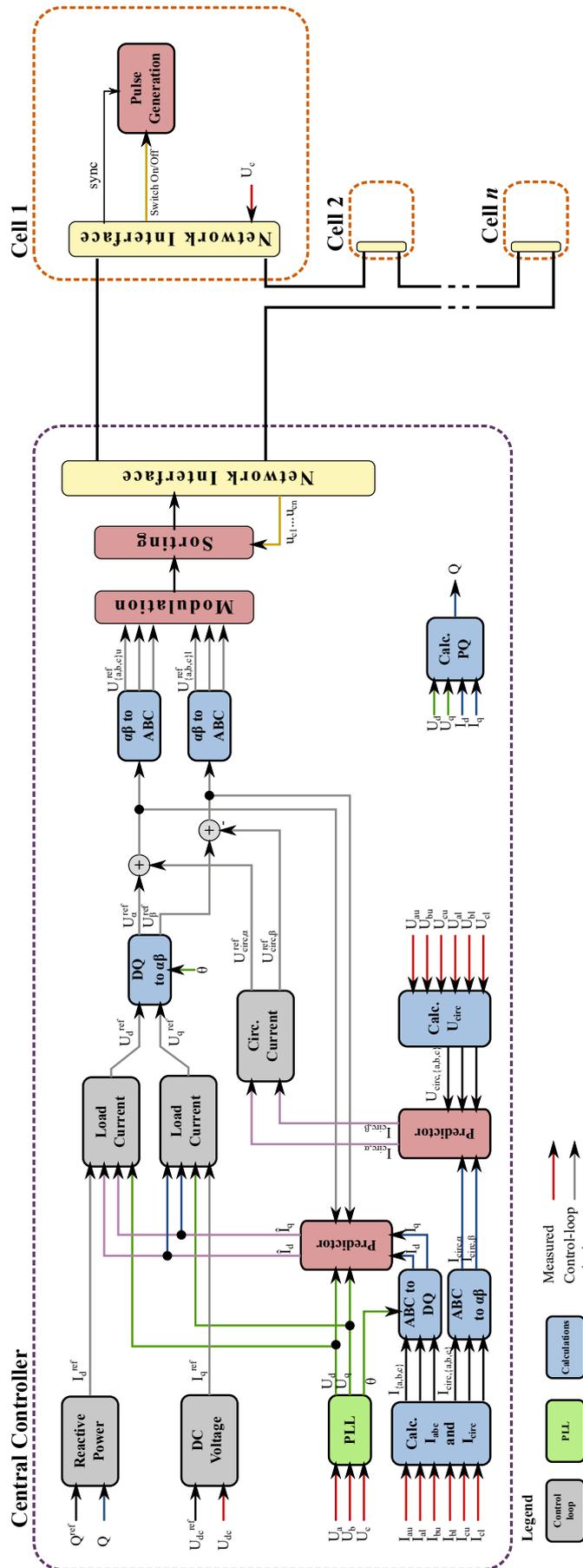


Figure 6.8: Block diagram of the control when using DiSortNet.

order, and selects the ones with the highest or lowest voltages depending on the arm current polarity. When the converter control has an internal network, the capacitor voltage will reach the central controller after the delay introduced by the network. We have tested two scenarios, one without delay and the other with two samples delay, and show the capacitor voltage of one arm in Fig. 6.11, where we see that the delay of only two samples has no influence on the capacitors balancing.

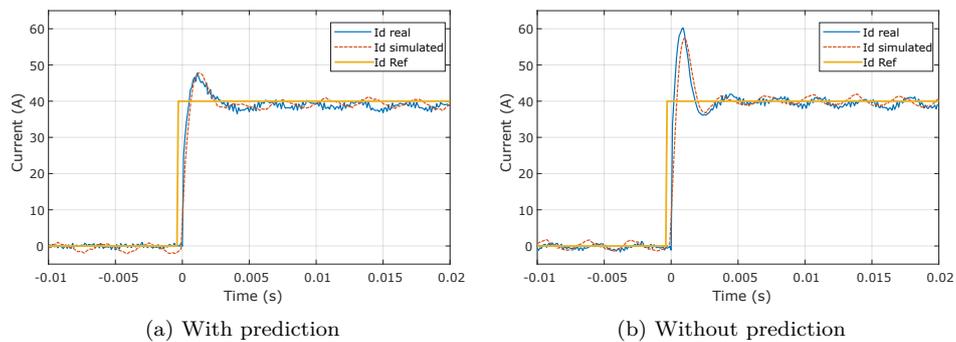


Figure 6.9: Experimental and simulation results for a system with delay of two samples.

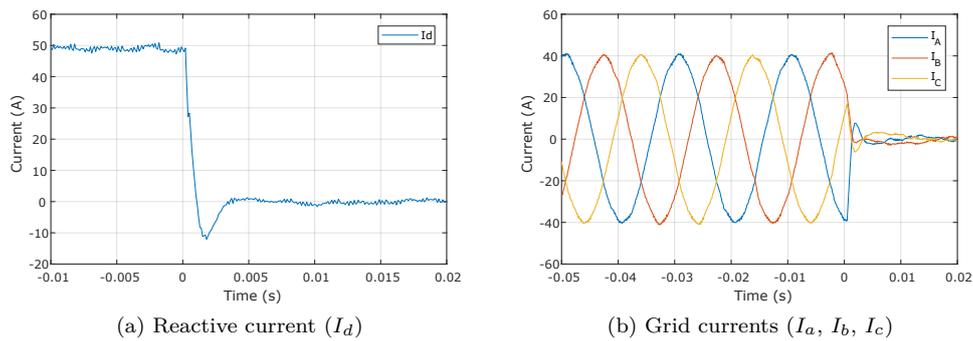


Figure 6.10: Experimental results for a system with a delay of three samples and the model-based prediction.

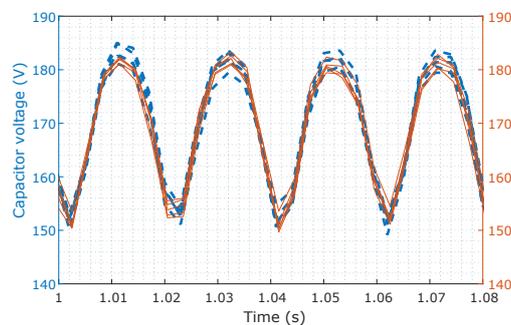


Figure 6.11: Comparison of the arm capacitor voltages when the modulation uses a sorted list based on measurements without delay (dashed blue line) and with a delay of two samples (continuous orange line).

6.6. Conclusions

The use of a digital communication network in the design of Modular Multilevel Converters introduces a loop delay in the plant, making the controller design harder, reducing dynamic performance, and restricting the minimum sampling period. Till now, researchers have to cope with these consequences in this application domain by minimizing the introduced loop delay. In this work, we adopted a different approach and used a model-based predictor to compensate for the loop delay.

By analogy with the Smith Predictor, we can state that the proposed method removes the loop delay from the denominator of the closed-loop. The PI controller considers only the delay-free plant [140]; hence it can have higher gains, and the transient system response is faster. By accounting for the disturbance effect in the model, it can also improve the disturbance rejection [137].

Moreover, the proposed method allows decoupling the sampling period from the communication Minimum Cycle Time. It has two consequences: either the sampling rate can be increased, or a higher Minimum Cycle Time (MCT) can be tolerated. Thus, the method increases the flexibility in the system design.

We proved the feasibility and better performance of the proposed method and we validated the approach using MATLAB/Simulink simulation and implementing it in a reduced-scale prototype. Additionally, we showed the necessary and sufficient conditions for stability, as well as the influence of parameter change into the closed-loop response.

The results obtained reassure engineers that it is possible to use an internal digital communication network in a Modular Multilevel Converter, with all the advantages regarding hardware design and ease of implementation/maintenance, while still having flexibility and excellent dynamic response.

Chapter 7

Conclusions and Future Work

Forecasts show that electric demand will surge in next decades, mainly a consequence of the electrification of most direct fuel uses. To cope with this increased demand, generation, transmission, and storage of electrical energy will have to evolve accordingly.

In this context, the Modular Multilevel Converter (MMC) stands out as a crucial technology for transmitting and controlling the energy at high voltage levels. Invented in 2001, the [MMC](#) consists of a high number of cells that are connected in series to achieve the necessary voltage level. In this work, we argued that the introduction of a digital communication network between the central controller and the cells brings many benefits, such as simplified assemblage and maintenance, higher functionality, and the possibility to partly decentralizing the control strategy and modifying the control architecture.

We discussed in this dissertation some characteristics of the digital communication network in this application domain and the requirements that the [MMC](#) imposes to it. Reliability and fault tolerance are vital aspects, leading designers to prefer the ring topology because it is the simplest one to offer two disjoint paths between any two nodes. Another advantage of the ring network is the minimization of the link lengths.

On the other hand, the latency of ring networks is linearly dependent on the number of nodes. As the number of nodes increases, and we observed that a typical [MMC](#) has hundreds of them, the Minimum Cycle Time of ring networks becomes too long for high-performance control of the converter. It is a common problem with solutions adopting ring topology and the most prominent protocols employed in this domain, [PESnet](#) and [EtherCAT](#), are no exception.

We explored a co-design approach where the network and the control designs take into consideration some characteristics of both domains and explore them to increase the overall system performance. Markedly, the protocols proposed in this thesis reduce the delay to values close to the minimum possible in ring networks by reducing both the packet payload and forwarding delay in the nodes. As the network is faster, the control algorithm can have a better performance by increasing its update rate.

The first proposed protocol, the TTRing, uses partly decentralized control strategies that reduce the packet payload by implementing the capacitor balancing inside the nodes; thus avoiding the necessity to transmit the capacitor voltage measurements to the central controller. The TTRing protocol minimizes the forwarding delay adopting a time-triggered paradigm, such that the node knows in advance when incoming packets must be forwarded to the next node, thus removing the need to read data from the packet before being able to transmit it. A second phase of the protocol, where the nodes share information with their immediate neighbors, can make use of the full capacity of the network.

The second proposed protocol, the DiSortNet, makes the co-design approach even more apparent. It has three pillars: the Dual Insertion Sorting, the distributed Min/Max identification, and the Compact Modulation.

The first pillar, the Dual Insertion Sorting, maintains an organized list of the cells, where the ones with the lowest voltages are close to the bottom, and the ones with the highest voltages are close to the top. It relaxes the ordering, such that it needs only the information of the minimum and maximum voltages of the cells to update the list.

The second pillar, the distributed Min/Max identification, uses the network to identify the cells in an arm that have the minimum and maximum voltage; thus removing the necessity to transmit the voltage measurements to the central controller and also off-loading part of the processing to the network. Note that both strategies combined remove the dependency of the balancing payload with the number of cells.

The third pillar, the Compact Modulation, aims in reducing the amount of data that the central controller sends to the cells to synthesize the desired voltage at the converter output. Still, the user can choose among most of the modulation strategies available in the literature.

All three strategies combined enable the DiSortNet protocol to reach a Minimum Cycle Time close to the minimum possible while implementing the widely adopted Sorting strategy and keeping a flexible modulation scheme.

Although both protocols have quasi-optimum performance, they do not represent the whole delay in transferring information across the network. The network interfaces also add delay when receiving and transmitting packets, associated to the data transfer to and from the protocol stack and application software. Thus, we also addressed this aspect and designed strategies to minimize the data transfer time upon packet reception and transmission. Namely, we studied the impact of using different Medium Access Control modules synthesized on Field Programmable Gate Arrays (FPGAs) and designed hardware accelerators that allow receiving packets with minimal and constant delay, independently of the packet size.

Finally, note that, no matter how small, the network introduces additional loop delay and limits the control sampling frequency and transient response. For this reason, we

introduced a model-based predictor to compensate for the loop delay and overcome these limitations. Two benefits of this approach are possible: either designers can increase the sampling rate and control performance or employ a slower communication protocol/technology, tolerating longer delays.

As [MMCs](#) are complex converters employed in mission-critical applications, we defended the relevance of accurate modeling of the converter, one that includes not only the control and power circuit domains but also the network one. With this purpose, we proposed a simplified co-simulation strategy that allows accounting for the effects of the network into the overall system while imposing an execution time that is significantly smaller than other common co-simulation approaches that work in lock-step. Results of such approach showed relevant aspects of networked controlled [MMCs](#), such as the higher probability of losing packets of the nodes at the end of the ring network and the effect that losing packets has on the converter current harmonic distortion. We also discussed the relevance of an accurate simulation that accounts not only for the control and power circuits but also the communication artifacts; a point overlooked this far.

Everything combined, this work contributed to the understanding of what are the advantages, limitations, possibilities, and difficulties that the use of digital communications brings to Modular Multilevel Converters. Moreover, it also provided a set of tools to circumvent the most important of those limitations and difficulties, potentially encouraging designers to implement network controlled [MMCs](#).

7.1. Future work

Three years for developing such research is sufficient for deepening our knowledge in the area of study and expanding the state-of-the-art, but it certainly falls short of what is necessary to explore all the details and possibilities identified along the way. Furthermore, the room for re-working or fixing problems, in particular those related to the practical experiments, is limited. Therefore, we list below the further developments that we plan or suggest as a continuation of this work:

- Investigate further the Insertion Sorting strategy, because it is the main limiting factor of the DiSortNet protocol in terms of network size. As the calculation of Minimum Cycle Time (MCT) showed, the DiSortNet protocol has the potential to be much faster than the current solutions as the number of nodes in the network increases.
- Implement the TTRing network and get experimental results to prove its effectiveness and check limits and drawbacks;
- Further validate the proposed protocols in a networked controlled converter prototype or a Hardware-in-the-loop platform. For this, we intend to build a network with

several nodes that is more representative of real converters (e.g., 30 nodes) ;

- Carry out a thorough reliability study of [MMCs](#) built with the protocols we proposed. In this work we addressed reliability only by the use of ring topology, but the topology per se is not a guarantee of fault tolerance. As reliability is one of the hard requirements of any communication solution for network controlled [MMCs](#), this is a vital point we must address;
- Review the Node Carrier board, as we faced problems with the TI's Ethernet Physical Layers (PHYs) output driver strength that limited the ability of using Gigabit Ethernet. As discussed, the increased speed of Gigabit Ethernet is a natural path for gains in performance and we would benefit from having a hardware platform to test it and develop new concepts;
- Explore other network topologies than ring, since this topology shows an [MCT](#) that has a linear dependency with the number of nodes, which becomes a limiting factor for very large numbers of cells;
- Extend the cosimulation strategy to cover networks with delays longer than one sample and implement the traditional lock-step cosimulation approach, so we can study cases of protocols that are influenced by the control plant. Note that in the solutions we proposed, the network influences the control plant but not the opposite.

Bibliography

- [1] D. Clark. (2012, Feb.) Has the Kyoto protocol made any difference to carbon emissions? Date last accessed Sep 07, 2018. [Online]. Available: <https://www.theguardian.com/environment/blog/2012/nov/26/kyoto-protocol-carbon-emissions>
- [2] C. Frisch, P. Donohoo-Vallett, C. Murphy, E. Hodson, and N. Horner, “An electrified nation: A review of study scenarios and future analysis needs for the United States,” *IEEE Power Energy Mag.*, vol. 16, no. 4, pp. 90–98, Jul. 2018.
- [3] V. K. Sood, “HVDC transmission,” in *Power Electronic Handbook*, 3rd ed. Butterworth-Heinemann, 2011, ch. 31, pp. 823–849.
- [4] H. Akagi, “Classification, terminology, and application of the modular multilevel cascade converter (MMCC),” *IEEE Trans. Power Electron.*, vol. 26, no. 11, pp. 3119–3130, Nov. 2011.
- [5] Siemens AG. HVDC plus IGBT converter modules. Date last accessed Sep 28, 2018. [Online]. Available: <https://www.siemens.com/press/IM2015040626EMEN>
- [6] A. Lesnicar and R. Marquardt, “An innovative modular multilevel converter topology suitable for a wide power range,” in *Proc. IEEE Power Tech Conference*, vol. 3, June 2003, pp. 6 pp. Vol.3–.
- [7] M. Hagiwara and H. Akagi, “Control and experiment of pulsewidth-modulated modular multilevel converters,” *IEEE Trans. Power Electron.*, vol. 24, no. 7, pp. 1737–1746, 2009.
- [8] S. Yang, Y. Tang, and P. Wang, “Distributed control for a modular multilevel converter,” *IEEE Trans. Power Electron.*, vol. PP, no. 99, p. 1, 2017.
- [9] T. M. Wigley, “The kyoto protocol: CO₂ CH₄ and climate implications,” *Geophysical research letters*, vol. 25, no. 13, pp. 2285–2288, 1998.
- [10] R. Henson, *The rough guide to climate change*. Dorling Kindersley Ltd, 2011.
- [11] United Nations. (2015) The paris agreement. Date last accessed Sep 07, 2018. [Online]. Available: <https://unfccc.int/process-and-meetings/the-paris-agreement/the-paris-agreement>

- [12] J. Rogelj, M. Den Elzen, N. Höhne, T. Fransen, H. Fekete, H. Winkler, R. Schaeffer, F. Sha, K. Riahi, and M. Meinshausen, "Paris agreement climate proposals need a boost to keep warming well below 2C," *Nature*, vol. 534, no. 7609, p. 631, 2016.
- [13] J. H. Williams, A. DeBenedictis, R. Ghanadan, A. Mahone, J. Moore, W. R. Morrow, S. Price, and M. S. Torn, "The technology path to deep greenhouse gas emissions cuts by 2050: the pivotal role of electricity," *Science*, p. 1208365, 2011.
- [14] T. Mai, D. Steinberg, J. Logan, D. Bielen, K. Eurek, and C. McMillan, "An electrified future: Initial scenarios and future research for U.S. energy and electricity systems," *IEEE Power Energy Mag.*, vol. 16, no. 4, pp. 34–47, Jul. 2018.
- [15] R. Fowler, O. Elmhirst, and J. Richards, "Electrification in the United Kingdom: A case study based on future energy scenarios," *IEEE Power Energy Mag.*, vol. 16, no. 4, pp. 48–57, Jul. 2018.
- [16] G. J. Kramer and M. Haigh, "No quick switch to low-carbon energy," *Nature*, vol. 462, no. 7273, p. 568, 2009.
- [17] M. P. Bahrman and B. K. Johnson, "The ABCs of HVDC transmission technologies," *IEEE Power Energy Mag.*, vol. 5, no. 2, pp. 32–44, Mar. 2007.
- [18] S. Kenzelmann, A. Rufer, D. Dujic, F. Canales, and Y. R. de Novaes, "Isolated DC/DC structure based on modular multilevel converter," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 89–98, Jan. 2015.
- [19] N. Flourentzou, V. G. Agelidis, and G. D. Demetriades, "VSC-based HVDC power transmission systems: An overview," *IEEE Trans. Power Electron.*, vol. 24, no. 3, pp. 592–602, Mar. 2009.
- [20] E. H. Watanabe, M. Aredes, P. G. Barbosa, F. K. de Araújo e Lima, R. F. da Silva Dias, and G. S. Jr, "Flexible AC transmission systems," in *Power Electronic Handbook*, 3rd ed. Butterworth-Heinemann, 2011, ch. 31, pp. 851–877.
- [21] C. Abbate, G. Busatto, and F. Iannuzzo, "High-voltage, high-performance switch using series-connected IGBTs," *IEEE Trans. Power Electron.*, vol. 25, no. 9, pp. 2450–2459, Sep. 2010.
- [22] P. K. R., *FACTS controllers in power transmission and distribution*. New Age International, 2007.
- [23] M. A. Perez, S. Bernet, J. Rodriguez, S. Kouro, and R. Lizana, "Circuit topologies, modeling, control schemes, and applications of modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 4–17, 2015.
- [24] A. Nabae, I. Takahashi, and H. Akagi, "A new neutral-point-clamped PWM inverter," *IEEE Trans. Ind. App.*, vol. IA-17, no. 5, pp. 518–523, Sep. 1981.

- [25] T. A. Meynard and H. Foch, "Multi-level conversion: high voltage choppers and voltage-source inverters," in *IEEE Power Electr. Spec. Conf. (PESC)*, vol. 1, Jun. 1992, pp. 397–403.
- [26] M. Marchesoni, M. Mazzucchelli, and S. Tenconi, "A non conventional power converter for plasma stabilization," in *IEEE Power Electron. Spec. Conf. (PESC)*, Apr. 1988, pp. 122–129 vol.1.
- [27] R. Marquardt, "Stromrichterschaltungen mit verteilten energiespeichern," *German Patent DE10103031A1*, vol. 24, 2001.
- [28] L. G. Franquelo, J. Rodriguez, J. I. Leon, S. Kouro, R. Portillo *et al.*, "The age of multilevel converters arrives," *IEEE Ind. Electron. Mag.*, vol. 2, no. 2, pp. 28–39, 2008.
- [29] S. Khomfoi and L. M. Tolbert, "Multilevel power converters," in *Power Electronic Handbook*, 3rd ed. Butterworth-Heinemann, 2011, ch. 31, pp. 823–849.
- [30] J. Rodriguez, S. Bernet, P. K. Steimer, and I. E. Lizama, "A survey on neutral-point-clamped inverters," *IEEE Trans. Ind. Electron.*, vol. 57, no. 7, pp. 2219–2230, Jul. 2010.
- [31] J. S. Lai and F. Z. Peng, "Multilevel converters – a new breed of power converters," *IEEE Trans. Ind. App.*, vol. 32, pp. 509–517, 1996.
- [32] I. Colak, E. Kabalci, and R. Bayindir, "Review of multilevel voltage source inverter topologies and control schemes," *Energy Conversion and Management*, vol. 52, no. 2, pp. 1114–1128, 2011.
- [33] A. Nami, J. Liang, F. Dijkhuizen, and G. D. Demetriades, "Modular multilevel converters for HVDC applications: Review on converter cells and functionalities," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 18–36, Jan. 2015.
- [34] T. Modeer, H.-P. Nee, and S. Norrga, "Loss comparison of different sub-module implementations for modular multilevel converters in HVDC applications," *EPE Journal*, vol. 22, no. 3, pp. 32–38, 2012.
- [35] Working Group B4.57, *TB 504: Guide for the Development of Models for HVDC Converters in a HVDC Grid*. CIGRE, dec 2014.
- [36] S. Debnath, J. Qin, B. Bahrani, M. Saeedifard, and P. Barbosa, "Operation, control, and applications of the modular multilevel converter: A review," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 37–53, 2015.
- [37] C. L. Toh and L. E. Norum, "A performance analysis of three potential control network for monitoring and control in power electronics converter," in *IEEE Int. Conf. Ind. Technology (ICIT)*. IEEE, 2012, pp. 224–229.

- [38] —, “A high speed control network synchronization jitter evaluation for embedded monitoring and control in modular multilevel converter,” in *IEEE PowerTech*. IEEE, 2013, pp. 1–6.
- [39] —, “Implementation of high speed control network with fail-safe control and communication cable redundancy in modular multilevel converter,” in *European Conf. Power Electronics and Applications (EPE)*. IEEE, 2013, pp. 1–10.
- [40] C. L. Toh, “Communication network for internal monitoring and control in multilevel power electronics converter,” Ph.D. dissertation, Norwegian University of Science and Technology, 2014.
- [41] C. L. Toh and L. E. Norum, “Synchronization mechanisms for internal monitoring and control in power electronics converter,” *Journal of Electrical Engineering*, 2014.
- [42] D. Cottet, W. van der Merwe, F. Agostini, G. Riedel, N. Oikonomou, A. Rueetschi, T. Geyer, T. Gradinger, R. Velthuis, B. Wunsch *et al.*, “Integration technologies for a fully modular and hot-swappable MV multi-level concept converter,” in *Proc. PCIM*. VDE, 2015, pp. 1–8.
- [43] P. Dan Burlacu, L. Mathe, M. Rejas, H. Pereira, A. Sangwongwanich, and R. Teodorescu, “Implementation of fault tolerant control for modular multilevel converter using EtherCAT communication,” in *IEEE Int. Conf. Ind. Technology (ICIT)*. IEEE, 2015, pp. 3064–3071.
- [44] S. Huang, R. Teodorescu, and L. Mathe, “Analysis of communication based distributed control of MMC for HVDC,” in *15th European Conference on Power Electronics and Applications (EPE)*. IEEE, 2013, pp. 1–10.
- [45] L. Mathe, P. D. Burlacu, and R. Teodorescu, “Control of a modular multilevel converter with reduced internal data exchange,” *IEEE Trans. Industrial Informatics*, vol. 13, no. 1, pp. 248–257, Feb. 2017.
- [46] I. Celanovic, “A distributed digital control architecture for power electronics systems,” Master’s thesis, 2000.
- [47] I. Milosavljevic, “Power electronics system communications,” Master’s thesis, Virginia Tech, 1999.
- [48] T. Laakkonen, “Distributed control architecture of power electronics building-block-based frequency converters,” Ph.D. dissertation, Lappeenranta University of Technology, 2010.
- [49] H. Tu and S. Lukic, “Comparative study of PESnet and SyCCo bus: Communication protocols for modular multilevel converter,” in *IEEE Energy Conversion Congress and Exposition (ECCE)*, Oct. 2017, pp. 1487–1492.

- [50] J. Gerdes, “Siemens debuts HVDC PLUS with San Francisco trans bay cable,” *Living Energy*, pp. 28–31, 2011.
- [51] GE Europe. (2018, Jul.) DolWin3 - the project that will deliver clean energy to over one million households. Youtube. Date last accessed Sep, 2018.
- [52] *IEC 60038:2009 IEC standard voltages*. IEC, 2009.
- [53] M. M. C. Merlin, T. C. Green, P. D. Mitcheson, D. R. Trainer, R. Critchley, W. Crookes, and F. Hassan, “The alternate arm converter: A new hybrid multi-level converter with DC-fault blocking capability,” *IEEE Trans. Power Del.*, vol. 29, no. 1, pp. 310–317, Feb. 2014.
- [54] J. Wang, R. Burgos, and D. Boroyevich, “A survey on the modular multilevel converters – modeling, modulation and controls,” in *IEEE Energy Conversion Congress and Exposition (ECCE)*. IEEE, 2013, pp. 3984–3991.
- [55] P. R. Remus Teodorescu, Marco Liserre, *Grid Converters for Photovoltaic and Wind Power Systems*. John Wiley and Sons Ltd, 2011.
- [56] P. Rodriguez, A. Luna, M. Ciobotaru, R. Teodorescu, and F. Blaabjerg, “Advanced grid synchronization system for power converters under unbalanced and distorted operating conditions,” in *Ann. Conf. IEEE Industrial Electronics Society (IECON)*, Nov. 2006, pp. 5173–5178.
- [57] J. Mei, K. Shen, B. Xiao, L. M. Tolbert, and J. Zheng, “A new selective loop bias mapping phase disposition PWM with dynamic voltage balance capability for modular multilevel converter,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 2, pp. 798–807, 2014.
- [58] G. S. Konstantinou and V. G. Agelidis, “Performance evaluation of half-bridge cascaded multilevel converters operated with multicarrier sinusoidal PWM techniques,” in *IEEE Conf. Industrial Electronics and Applications (ICIEA)*. IEEE, 2009, pp. 3399–3404.
- [59] K. Sharifabadi, L. Harnefors, H.-P. Nee, S. Norrga, and R. Teodorescu, *Design, Control, and Application of Modular Multilevel Converters for HVDC Transmission Systems*. John Wiley & Sons, 2016.
- [60] S. Rohner, S. Bernet, M. Hiller, and R. Sommer, “Modulation, losses, and semiconductor requirements of modular multilevel converters,” *IEEE Trans. Ind. Electron.*, vol. 8, no. 57, pp. 2633–2642, 2010.
- [61] W. Li, L.-A. Grégoire, and J. Bélanger, “A modular multilevel converter pulse generation and capacitor voltage balance method optimized for FPGA implementation,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 5, pp. 2859–2867, 2015.

- [62] A. Hassanpoor, L. Ängquist, S. Norrga, K. Ilves, and H. Nee, "Tolerance band modulation methods for modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 311–326, Jan. 2015.
- [63] Q. Tu and Z. Xu, "Impact of sampling frequency on harmonic distortion for modular multilevel converter," *IEEE Trans. Power Del.*, vol. 26, no. 1, pp. 298–306, 2011.
- [64] M. S. A. Dahidah and V. G. Agelidis, "Selective harmonic elimination PWM control for cascaded multilevel voltage source converters: A generalized formula," *IEEE Trans. Power Electron.*, vol. 23, no. 4, pp. 1620–1630, Jul. 2008.
- [65] K. Ilves, A. Antonopoulos, S. Norrga, and H. Nee, "A new modulation method for the modular multilevel converter allowing fundamental switching frequency," *IEEE Trans. Power Electron.*, vol. 27, no. 8, pp. 3482–3494, Aug. 2012.
- [66] Y. Li, E. A. Jones, and F. . Wang, "The impact of voltage-balancing control on switching frequency of the modular multilevel converter," *IEEE Trans. Power Electron.*, vol. 31, no. 4, pp. 2829–2839, Apr. 2016.
- [67] K. Ilves, L. Harnefors, S. Norrga, and H.-P. Nee, "Analysis and operation of modular multilevel converters with phase-shifted carrier PWM," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 268–283, 2015.
- [68] E. Solas, G. Abad, J. A. Barrena, S. Aurtenetxea, A. Cárcar, and L. Zajac, "Modular multilevel converter with different submodule concepts – Part I: Capacitor voltage balancing method," *IEEE Trans. Ind. Electron.*, vol. 60, no. 10, pp. 4525–4535, Oct. 2013.
- [69] K. Wang, Y. Li, Z. Zheng, and L. Xu, "Voltage balancing and fluctuation-suppression methods of floating capacitors in a new modular multilevel converter," *IEEE Trans. Ind. Electron.*, vol. 60, no. 5, pp. 1943–1954, May 2013.
- [70] M. Dommaschk, J. Dorn, I. Euler, J. Lang, Q. Tu, and K. Wrflinger, "Drive for a phase module branch of a multilevel converter," Patent WO2008/086 760A1, 2008.
- [71] J. Qin and M. Saeedifard, "Reduced switching-frequency voltage-balancing strategies for modular multilevel HVDC converters," *IEEE Trans. Power Del.*, vol. 28, no. 4, pp. 2403–2410, 2013.
- [72] G. P. Adam, O. Anaya-Lara, G. M. Burt, D. Telford, B. W. Williams, and J. R. McDonald, "Modular multilevel inverter: Pulse width modulation and capacitor balancing technique," *IET Power Electronics*, vol. 3, no. 5, pp. 702–715, Sep. 2010.
- [73] M. Ricco, L. Mathe, E. Monmasson, and R. Teodorescu, "FPGA-based implementation of MMC control based on sorting networks," *Energies*, vol. 11, no. 9, p. 2394, 2018.

- [74] P. M. Meshram and V. B. Borghate, "A simplified nearest level control (NLC) voltage balancing method for modular multilevel converter (MMC)," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 450–462, Jan. 2015.
- [75] Z. Li, P. Wang, H. Zhu, Z. Chu, and Y. Li, "An improved pulse width modulation method for chopper-cell-based modular multilevel converters," *IEEE Trans. on Power Electron.*, vol. 27, no. 8, pp. 3472–3481, Aug. 2012.
- [76] M. Vatani, B. Bahrani, M. Saeedifard, and M. Hovd, "Indirect finite control set model predictive control of modular multilevel converters," *IEEE Trans. Smart Grid*, vol. 6, no. 3, pp. 1520–1529, May 2015.
- [77] M. Saeedifard and R. Iravani, "Dynamic performance of a modular multilevel back-to-back HVDC system," *IEEE Trans. Power Del.*, vol. 25, no. 4, pp. 2903–2912, 2010.
- [78] R. Picas, J. Pou, S. Ceballos, V. Agelidis, and M. Saeedifard, "Minimization of the capacitor voltage fluctuations of a modular multilevel converter by circulating current control," in *Ann. Conf. IEEE Industrial Electronics Society (IECON)*. IEEE, 2012, pp. 4985–4991.
- [79] G. Konstantinou, M. Ciobotaru, and V. G. Agelidis, "Selective harmonic elimination pulse-width modulation of modular multilevel converters," *IET Power Electronics*, vol. 6, no. 1, pp. 96–107, 2013.
- [80] V. Sklyarov and I. Skliarova, "High-performance implementation of regular and easily scalable sorting networks on an FPGA," *Microprocessors and Microsystems*, vol. 38, no. 5, pp. 470–484, 2014.
- [81] H. Saad, X. Guillaud, J. Mahseredjian, S. Denneri, and S. Nguéfeu, "MMC capacitor voltage decoupling and balancing controls," *IEEE Trans. Power Del.*, vol. 30, no. 2, pp. 704–712, Apr. 2015.
- [82] F. Deng and Z. Chen, "A control method for voltage balancing in modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 29, no. 1, pp. 66–76, Jan. 2014.
- [83] D. Siemaszko, "Fast sorting method for balancing capacitor voltages in modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 463–470, 2015.
- [84] M. Hagiwara, R. Maeda, and H. Akagi, "Control and analysis of the modular multilevel cascade converter based on double-star chopper-cells (MMCC-DSCC)," *IEEE Trans. Power Electron.*, vol. 26, no. 6, pp. 1649–1658, Jun. 2011.
- [85] R. Teodorescu, E.-P. Eni, L. Mathe, and P. Rodriguez, "Modular multilevel converter control strategy with fault tolerance," in *Int. Conf. Renewable Energies and Power Quality (ICRE PQ)*, 2013.

- [86] S. I. Seleme, L.-A. Grégoire, M. Cousineau, and P. Ladoux, "Decentralized controller for modular multilevel converter," in *Proc. PCIM*, May 2016, pp. 1–8.
- [87] S. Fan, K. Zhang, J. Xiong, and Y. Xue, "An improved control system for modular multilevel converters with new modulation strategy and voltage balancing control," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 358–371, Jan. 2015.
- [88] K. Ilves, L. Harnefors, S. Norrga, and H. Nee, "Predictive sorting algorithm for modular multilevel converters minimizing the spread in the submodule capacitor voltages," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 440–449, Jan. 2015.
- [89] B. S. Riar, T. Geyer, and U. K. Madawala, "Model predictive direct current control of modular multilevel converters: Modeling, analysis, and experimental evaluation," *IEEE Trans. Power Electron.*, vol. 30, no. 1, pp. 431–439, Jan. 2015.
- [90] J. Qin and M. Saeedifard, "Predictive control of a modular multilevel converter for a back-to-back HVDC system," *IEEE Trans. Power Del.*, vol. 27, no. 3, pp. 1538–1547, Jul. 2012.
- [91] K. Ilves, A. Antonopoulos, S. Norrga, and H.-P. Nee, "Steady-state analysis of interaction between harmonic components of arm and line quantities of modular multilevel converters," *IEEE Trans. Power Electron.*, vol. 27, no. 1, pp. 57–68, 2012.
- [92] J. Pou, S. Ceballos, G. Konstantinou, V. G. Agelidis, R. Picas, and J. Zaragoza, "Circulating current injection methods based on instantaneous information for the modular multilevel converter," *IEEE Trans. Ind. Electron.*, vol. 62, no. 2, pp. 777–788, Feb. 2015.
- [93] X. Yang, J. Li, X. Wang, W. Fan, and T. Q. Zheng, "Circulating current model of modular multilevel converter," in *Asia-Pacific Power and Energy Engineering Conf (APPEC)*, Mar. 2011, pp. 1–6.
- [94] Q. Tu, Z. Xu, and L. Xu, "Reduced switching-frequency modulation and circulating current suppression for modular multilevel converters," *IEEE Trans. Power Del.*, vol. 26, no. 3, pp. 2009–2017, Jul. 2011.
- [95] D. Wu and L. Peng, "Eliminating the influence of capacitor voltage ripple on current control for grid-connected modular multilevel converter," in *Ann. IEEE Applied Power Electronics Conference and Exposition (APEC)*, Mar. 2015, pp. 2128–2132.
- [96] W. Song, Z. Yang, Y. Liu, A. Huang, and B. Chen, "A layered modular controller structure for multilevel converters," in *IEEE Power Electronics Specialists Conf (PESC)*, Jun. 2007, pp. 1448–1452.
- [97] Y. Park, H. Yoo, H. Lee, M. Jung, S. Lee, C. Lee, S. Lee, and J. Yoo, "A simple and reliable PWM synchronization amp; phase-shift method for cascaded h-bridge

- multilevel inverters based on a standard serial communication protocol,” in *IEEE IAS Ann. Meeting*, vol. 2, Oct. 2006, pp. 988–994.
- [98] A. Marquez, J. I. Leon, S. Vazquez, and L. G. Franquelo, “Communications scheme of a modular power conversion system,” in *IEEE Int. Conf. Ind. Technology (ICIT)*, Mar. 2015, pp. 3034–3039.
- [99] L.-A. Grégoire, W. Wang, S. I. Seleme, and M. Fadel, “High reliability observers for modular multilevel converter capacitor voltage evaluation,” in *Proc. IEEE 8th Int. Power Electronics and Motion Control Conf. (IPEMC-ECCE Asia)*, May 2016, pp. 2332–2336.
- [100] H. Flatt, J. Jasperneite, and F. Schewe, “An FPGA based cut-through switch optimized for one-step PTP and real-time Ethernet,” in *Proc. IEEE Int. Symp. Precision Clock Synchronization for Measurement Control and Communication (ISPCS)*, Sep. 2013, pp. 7–12.
- [101] J. Woods, “Cut-through considerations and impacts to industrial networks,” Presented in IEEE 802.1 WG meeting, 2017. [Online]. Available: <http://www.ieee802.org/1/files/public/docs2017/new-woods-cutthroughconsiderations-0518-v01.pdf>
- [102] S. Kilts, *Advanced FPGA design: Architecture, Implementation, and Optimization*. John Wiley & Sons, Inc., 2007.
- [103] J. Blattner and H. Weibel, “Study on propagation delay variation of 100BASE-Tx Ethernet PHY chips,” in *Conf. on IEEE-1588*, 2004.
- [104] C. Carstensen, R. Christen, H. Vollenweider, R. Stark, and J. Biela, “A converter control field bus protocol for power electronic systems with a synchronization accuracy of $\pm 5\text{ns}$,” in *European Conf. Power Electronics and Applications (EPE-ECCE)*. IEEE, 2015, pp. 1–10.
- [105] “IEEE standard for a precision clock synchronization protocol for networked measurement and control systems,” *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*, 2015.
- [106] P. L. G. Malapelle, G. Torri, R. Moruzzi, and A. Oliva, “A new, modular, programmable, high speed digital control for large drives,” in *Ann. Conf. IEEE Industrial Electronics Society (IECON)*, vol. 1, Sep. 1994, pp. 210–214 vol.1.
- [107] J. A. du Toit, A. D. le Roux, and J. H. R. Enslin, “An integrated controller module for distributed control of power electronics,” in *Ann. IEEE Applied Power Electronics Conference and Exposition (APEC)*, vol. 2, Feb. 1998, pp. 874–880 vol.2.
- [108] G. Francis, “A synchronous distributed digital control architecture for high power converters,” Master’s thesis, Virginia Tech, 2004.

- [109] D. Orfanus, R. Indergaard, G. Prytz, and T. Wien, "EtherCAT-based platform for distributed control in high-performance industrial applications," in *IEEE Int. Conf. Emerging Technologies & Factory Automation (ETFA)*, Sep. 2013, pp. 1–8.
- [110] Beckhoff Automation GmbH & Co. (2017, Feb.) EtherCAT slave controller - section I. Date last accessed Jun 22, 2017. [Online]. Available: https://download.beckhoff.com/download/document/io/ethercat-development-products/ethercat_esc_datasheet_sec2_registers_2i9.pdf
- [111] G. Cena, S. Scanzio, A. Valenzano, and C. Zunino, "Ethernet for control automation technology," in *Industrial Communication Technology Handbook*, 2nd ed., R. Zurawski, Ed. CRC Press, 2015, ch. 18.
- [112] T. Maruyama and T. Yamada, "Hardware acceleration architecture for EtherCAT master controller," in *IEEE Int. Workshop Factory Communication Systems (WFCS)*, May 2012, pp. 223–232.
- [113] Beckhoff Automation GmbH & Co. HW datasheet ET1100 - section III. Date last accessed Jun 22, 2017. [Online]. Available: https://download.beckhoff.com/download/document/io/ethercat-development-products/ethercat_et1100_datasheet_v2i0.pdf
- [114] A. Hillers, H. Tu, and J. Biela, "Central control and distributed protection of the DSBC and DSCC modular multilevel converters," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, Sep. 2016, pp. 1–7.
- [115] A. Antonino, S. Straullu, S. Abrate, A. Nespola, P. Savio, D. Zeolla, J. R. Molina, R. Gaudino, S. Loquai, and J. Vinogradov, "Real-time gigabit Ethernet bidirectional transmission over a single SI-POF up to 75 meters," in *IOptical Fiber Communication Conf. and Exposition and the National Fiber Optic Engineers Conf.*, Mar. 2011, pp. 1–3.
- [116] O. Ciordia, C. Esteban, C. Pardo, and R. P. de Aranda, "Commercial silicon for gigabit communication over SI-POF," in *Plastic Optic Fiber conference*, 2013, pp. 109–116.
- [117] J. Jasperneite, M. Schumacher, and K. Weber, "Limits of increasing the performance of industrial Ethernet protocols," in *IEEE Int. Conf. Emerging Technologies & Factory Automation (ETFA)*, Sep. 2007, pp. 17–24.
- [118] G. Prytz, "A performance analysis of EtherCAT and PROFINET IRT," in *IEEE Int. Conf. Emerging Technologies & Factory Automation (ETFA)*. IEEE, 2008, pp. 408–415.
- [119] Patrick O'Farrell, "Latency in factory automation," Texas Instrument, Tech. Rep., Oct. 2015, date last accessed Oct 30, 2018. [Online]. Available: <http://www.ti.com/lit/an/snla240/snla240.pdf>

- [120] Xilinx Inc., *7 Series FPGAs GTX/GTH Transceivers (UG476)*, Aug. 2018.
- [121] A. Athavale and C. Christensen, *High-Speed Serial I/O Made Simple*, 1st ed. Xilinx Inc., 2005.
- [122] C. E. Spurgeon and J. Zimmerman, *Ethernet: The Definitive Guide*, 2nd ed. O'Reilly Media Inc., Mar. 2014.
- [123] Texas Instrument. DP83867E/IS/CS robust, high immunity, small formfactor 10/100/1000 Ethernet physical layer transceiver. Date last accessed Feb 2, 2017. [Online]. Available: <http://www.ti.com/lit/ds/symlink/dp83867is.pdf>
- [124] K. Mittra. (2016, oct) Marvell PHYs for low-latency industrial Ethernet. Date last accessed Feb 2, 2017. [Online]. Available: <http://blogs.marvell.com/2016/10/marvell-phys-for-low-latency-industrial-ethernet/>
- [125] Microchip. Datasheet KSZ8091MLX - 10BASE-T/100BASE-TX physical layer transceiver. Date last accessed Mar 21, 2017. [Online]. Available: <http://ww1.microchip.com/downloads/en/DeviceDoc/00002297A.pdf>
- [126] VITESSE. Gigabit Ethernet PHY device latency. Date last accessed Feb 2, 2017. [Online]. Available: <https://ethernet.microsemi.com/products/download.php?fid=4307&number\=VSC8224>
- [127] S. Vitturi, L. Peretti, L. Seno, M. Zigliotto, and C. Zunino, "Real-time Ethernet networks for motion control," *Computer Standards & Interfaces*, vol. 33, no. 5, pp. 465–476, 2011.
- [128] A. Varga and OpenSim Ltd. OMNeT++ – simulation manual. Date last accessed Jan 3, 2017. [Online]. Available: <https://omnetpp.org/doc/omnetpp/manual/>
- [129] P. Palensky, A. A. V. D. Meer, C. D. Lopez, A. Joseph, and K. Pan, "Cosimulation of intelligent power systems: Fundamentals, software architecture, numerics, and coupling," *IEEE Ind. Electron. Mag*, vol. 11, no. 1, pp. 34–50, Mar. 2017.
- [130] Texas Instrument. (2012, Jul.) Ethernet media access controller (EMAC)/management data input/output (MDIO) (SPRUHH1). Date last accessed Jan 9, 2019. [Online]. Available: <http://www.ti.com/lit/ug/spruhh1/spruhh1.pdf>
- [131] "IEEE standard for ethernet," *IEEE Std 802.3-2015 (Revision of IEEE Std 802.3-2012)*, pp. 1–4017, Mar. 2016.
- [132] Xilinx Inc., *Zynq-7000 All Programmable SoC (Z-7007S, Z-7012S, Z-7014S, Z-7010, Z-7015, and Z-7020): DC and AC Switching Characteristics (DS187)*, Jun. 2017.
- [133] —, *Vivado AXI Reference (UG1037)*, Jul. 2017.
- [134] —, *Zynq-7000 All Programmable SoC: Technical Reference Manual (UG585)*, Dec. 2017.

- [135] A. Dunkels, “Full TCP/IP for 8 bit architectures,” in *Proc. ACM/Usenix Int. Conf. Mobile Systems, Applications and Services (MobiSys)*, San Francisco, May 2003. [Online]. Available: <http://dunkels.com/adam/mobisys2003.pdf>
- [136] JBLopen. ARM Cortex-A interrupt latency. Date last accessed Mar 8, 2018. [Online]. Available: <https://www.jblopen.com/arm-cortex-a-interrupt-latency/>
- [137] Z. Palmor, “Time-delay compensation-smith predictor and its modifications,” *Control Systems Fundamentals*, 2000.
- [138] P. Cortes, J. Rodriguez, C. Silva, and A. Flores, “Delay compensation in model predictive current control of a three-phase inverter,” *IEEE Trans. Ind. Electron.*, vol. 59, no. 2, pp. 1323–1325, Feb. 2012.
- [139] K. J. Astrom and B. Wittenmark, *Computer-Controlled Systems Theory and Design*, 3rd ed. Dover Publications, 2011.
- [140] L. Mirkin and Z. J. Palmor, “Control issues in systems with loop delays,” in *Handbook of Networked and Embedded Control Systems*. Springer, Jan. 2005. [Online]. Available: http://dx.doi.org/10.1007/0-8176-4404-0_27
- [141] E. Garcia, P. J. Antsaklis, and L. A. Montestruque, *Model-based control of networked systems*. Springer, 2014.
- [142] T. P. Corrêa., E. J. Bueno, and F. J. Rodriguez, “Communication network latency compensation in a modular multilevel converter,” in *IEEE Energy Conversion Congress and Exposition (ECCE)*. IEEE, 2017.
- [143] L. Zhang, J. Qin, D. Shi, and Z. Wang, “Efficient modeling of hybrid MMCs for HVDC systems,” in *IEEE Energy Conversion Congress and Exposition (ECCE)*, 2017.
- [144] M. Moranchel, F. Huerta, I. Sanz, E. Bueno, and F. J. Rodríguez, “A comparison of modulation techniques for modular multilevel converters,” *Energies*, vol. 9, no. 12, p. 1091, 2016.

Appendix A

Node Carrier Board

A.1. Introduction

During the development of this work, we have tried several times to find a hardware platform that would enable us to implement the protocols proposed and check characteristics of the network.

The first board tested was the ZedBoard¹ together with the daughter board Ethernet FMC². The daughter board has four PHY ICs, the Marvel's 88E1510. Our measurements showed that this PHY has a large latency of 1.2 μ s, and it was impossible to get its datasheet to check if it had any register configuration could reduce the latency. Moreover, when we decided to pursue a platform that would allow us to assemble a network with more than just a couple of nodes, the ZedBoard + Ethernet FMC proved too expensive.

The second board employed was the Trenz Electronics' TE0729 board³ and TEB0729 carrier board, because the PHY employed, the Microship's KSZ8081, had lower delay and an open datasheet. We used this platform to measure the delays presented in Chapter 5 and to design and test the hardware accelerators proposed in this same chapter. Besides the fact that the KSZ8081 supports only 100BASE-TX/10BASE-T Ethernet, we had again problems with the cost as a limiting factor for assembling a large network.

Though we wanted to avoid the risk of a hardware design at an advanced stage of the work (the beginning of the third year), we weighted the benefits of a custom platform and decided to make the design. Fig. A.1 is a picture of the developed board together with the Zynq module. In the next section, we give some details of it.

¹<http://www.zedboard.org>

²<http://ethernetfmc.com/>

³<https://wiki.trenz-electronic.de/display/PD/TE0729+TRM>

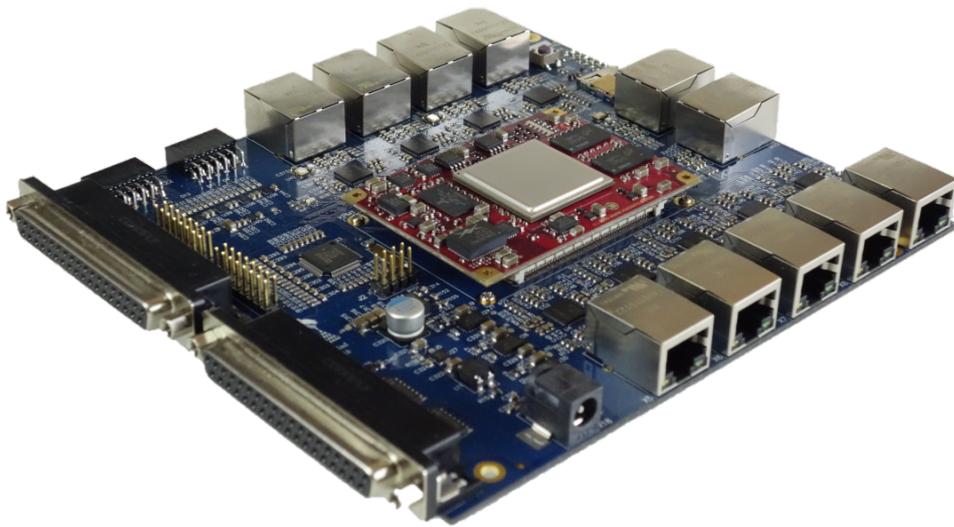


Figure A.1: Node Carrier board designed for the emulation of the communication network of an MMC.

A.2. Characteristics

The first point to consider was which device to use for the design. As we had been working with Xilinx' Zynq family, we decided to stay with it. To reduce the risk, the option was for developing a carrier board with the **PHYs** and other ICs that would use a commercial module or development board.

After a research of the modules available, our decision was in favor of Knowledge Resources' krm-3z7 modules⁴. The main reason was that this family has the highest number of pins from the Programmable Logic made available through the board-to-board connectors. This was important to allow the maximum number of **PHYs** in a single board.

The second point was the choice of the **PHY** IC. We opted for the TI's DP83867I⁵, a Gigabit Ethernet IC designed for the industrial environment and optimized for low latency. The documentation was available and our previous experience with TI support was positive. The version chosen was the smaller package with RGMII interface, again to maximize the number of Ethernet ports in one board, though compromising a bit the **PHY** latency.

At the time of the design, we had already in mind the construction of the ring topology, where each two ports represent one network node. Besides the communication part, we included a 16-bit Analog Digital Converter with eight channels and drivers for Pulse Width Modulation (PWM) outputs. We connect the analog in and **PWM** out channel to two connectors that are compatible with the OPAL-RT Hardware-in-the-Loop platform. As each Node Carrier board can sample eight analog channels and the OPAL-RT device typically outputs 16 channels, we added two pin headers that allow the higher channels

⁴<http://www.knowres.ch/overview-of-products/#krm-3z7-module-family>

⁵<http://www.ti.com/product/DP83867IR>

to be re-routed to a second Node Carrier board (via X16 and X17).

A.3. Schematics

In the next pages we include the circuit diagrams of the Emulation Board. The cover page contains a representation of the board in a block diagram.

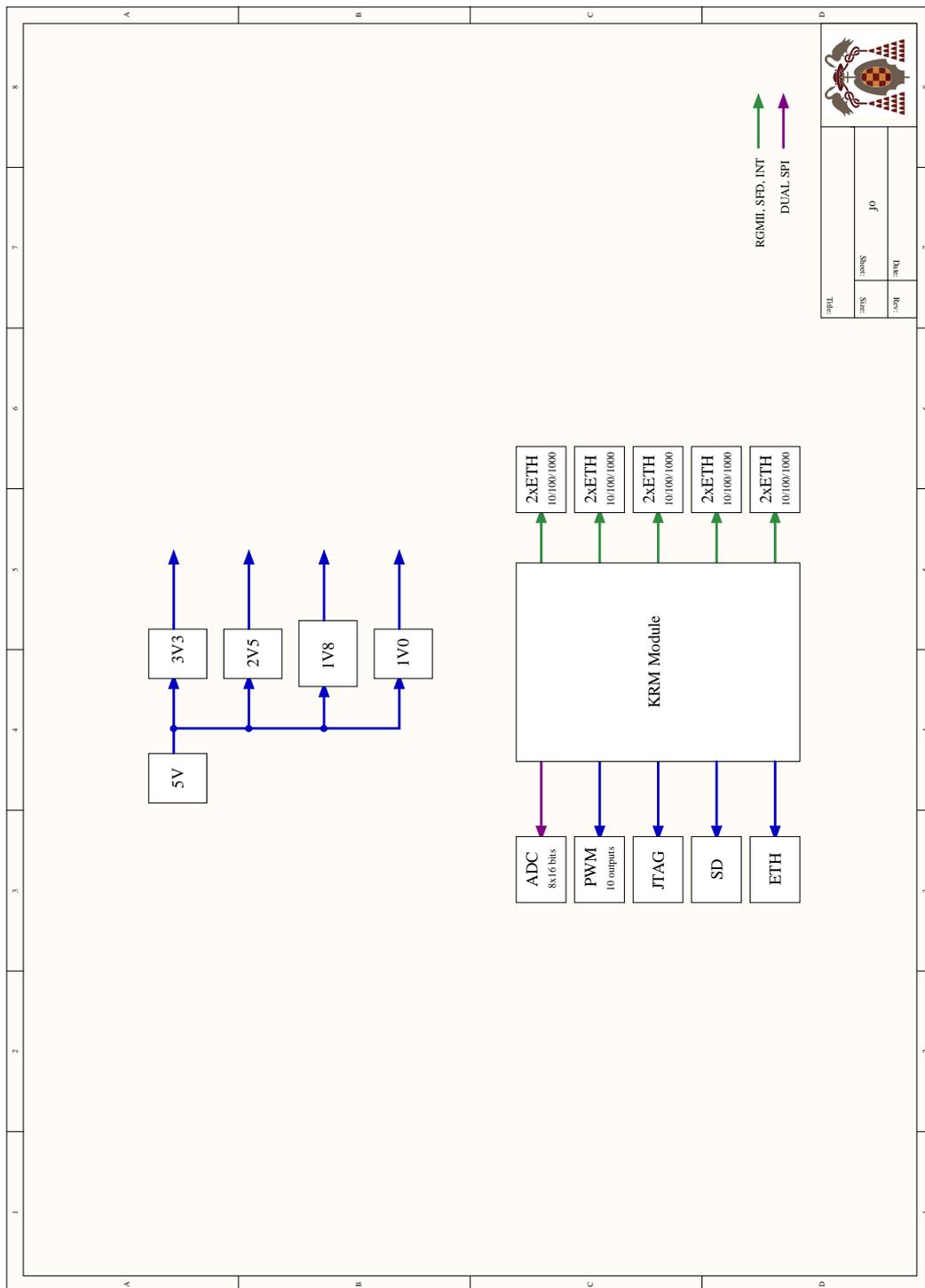


Figure A.2: Cover sheet

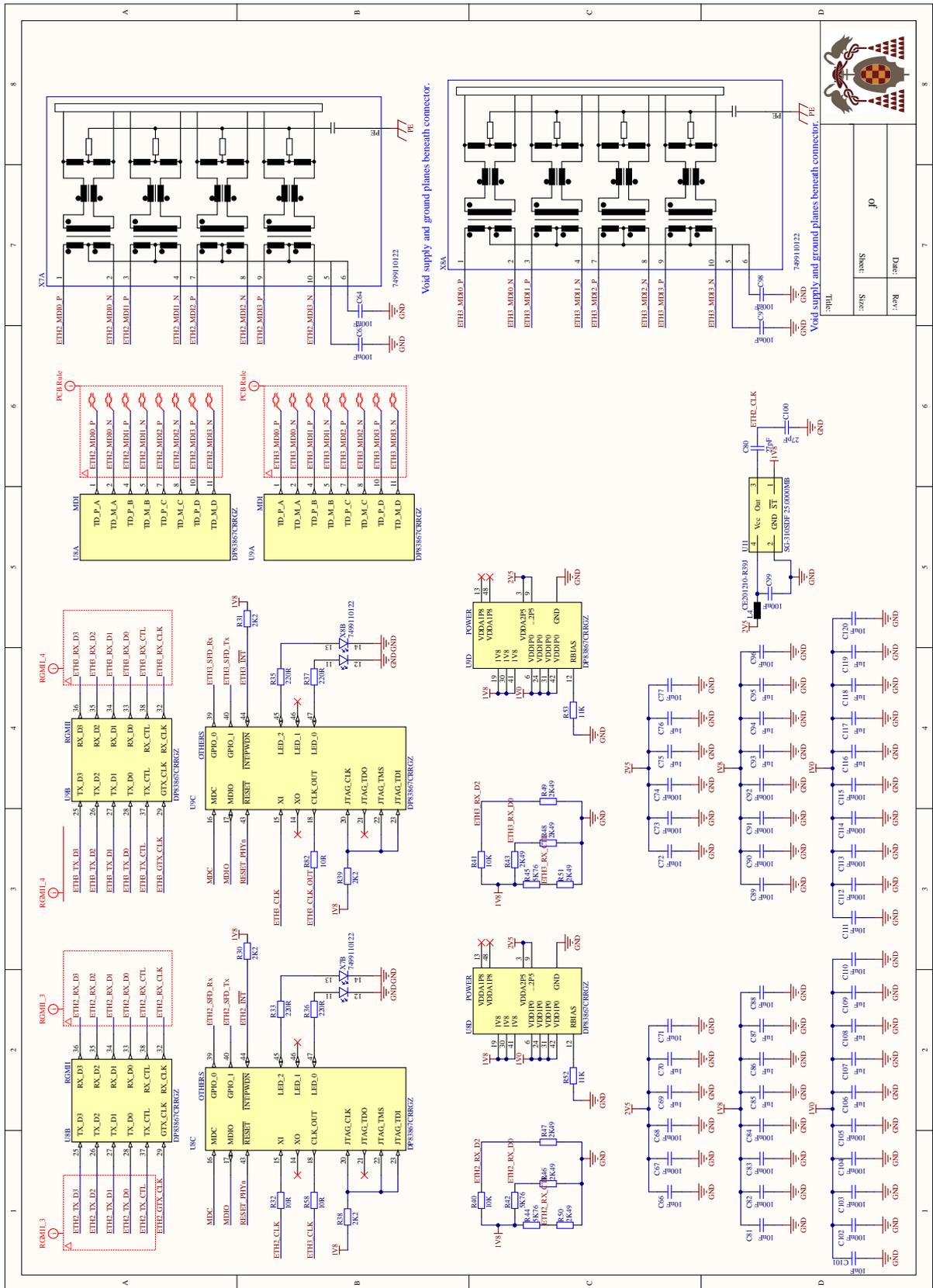


Figure A.6: Sheet 5

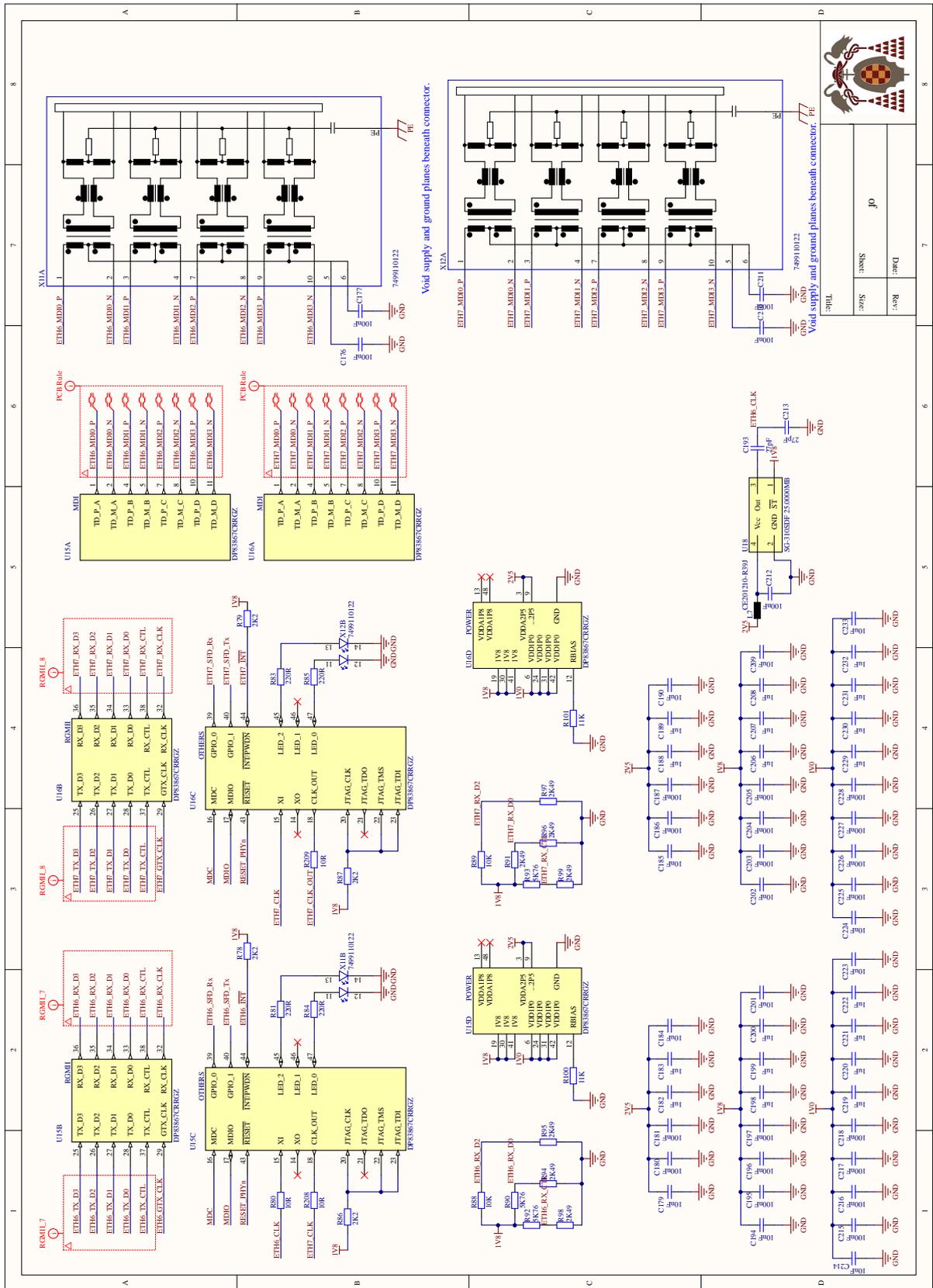


Figure A.8: Sheet 7

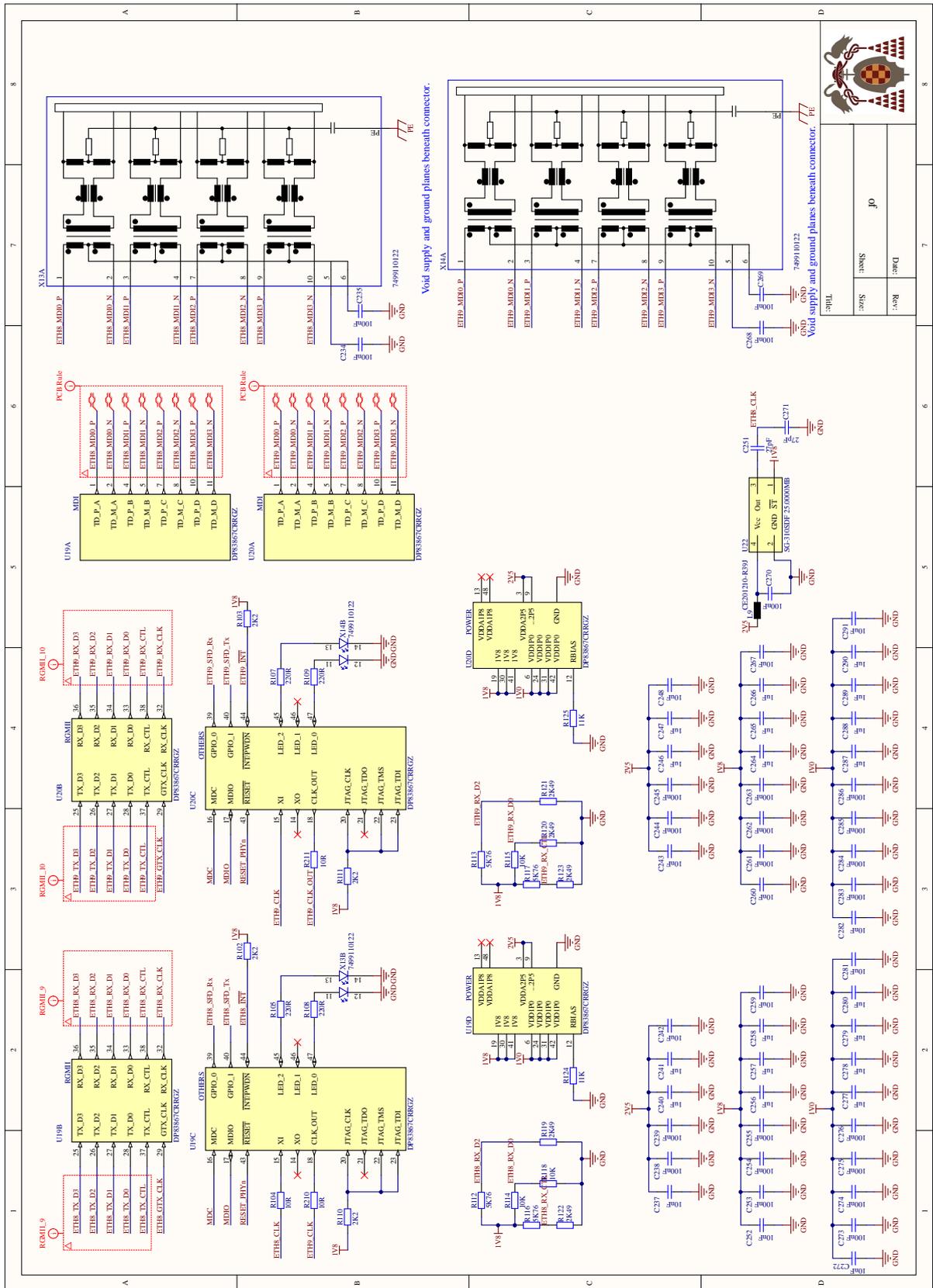


Figure A.9: Sheet 8

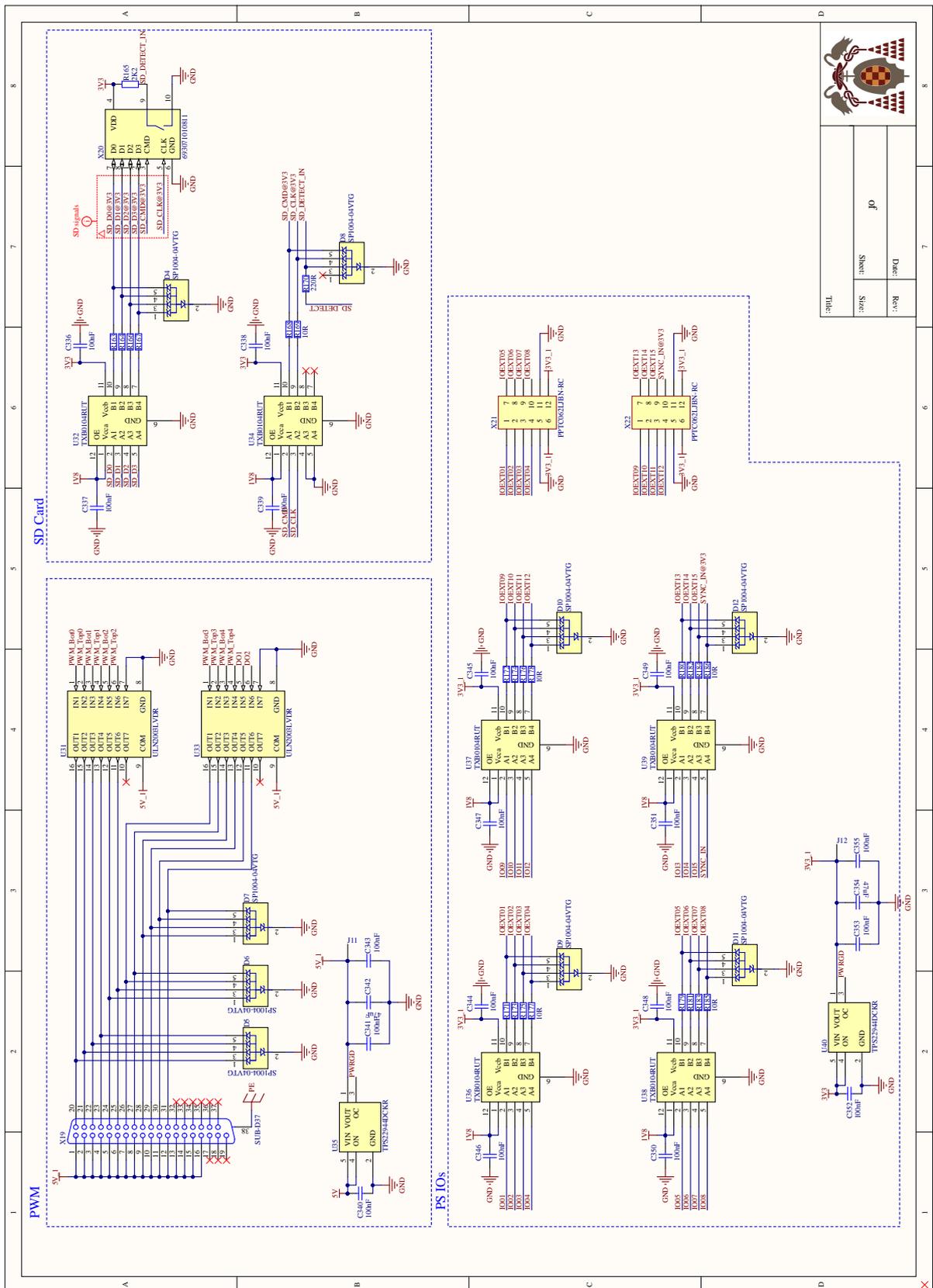


Figure A.10: Sheet 9

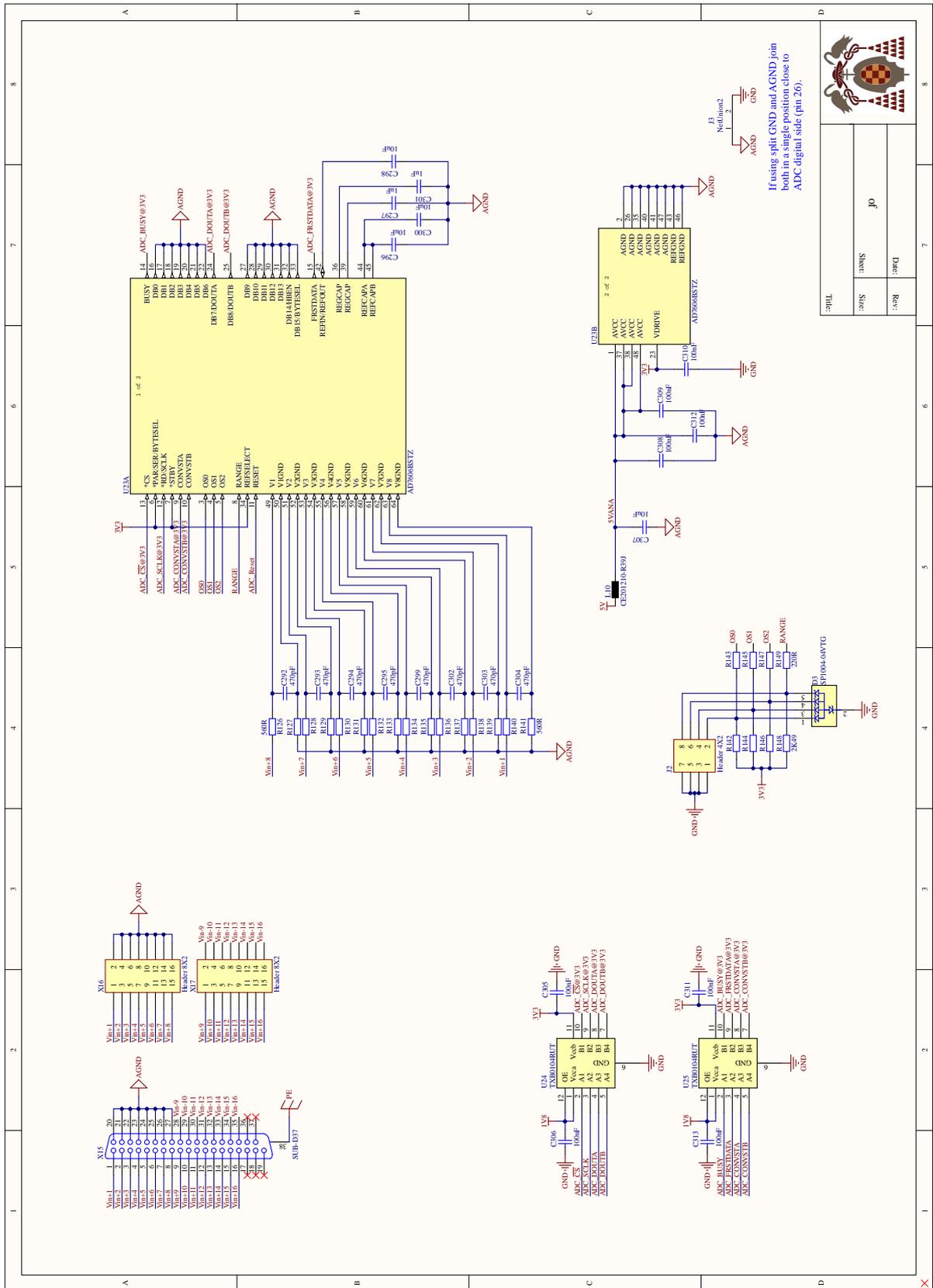


Figure A.11: Sheet 10

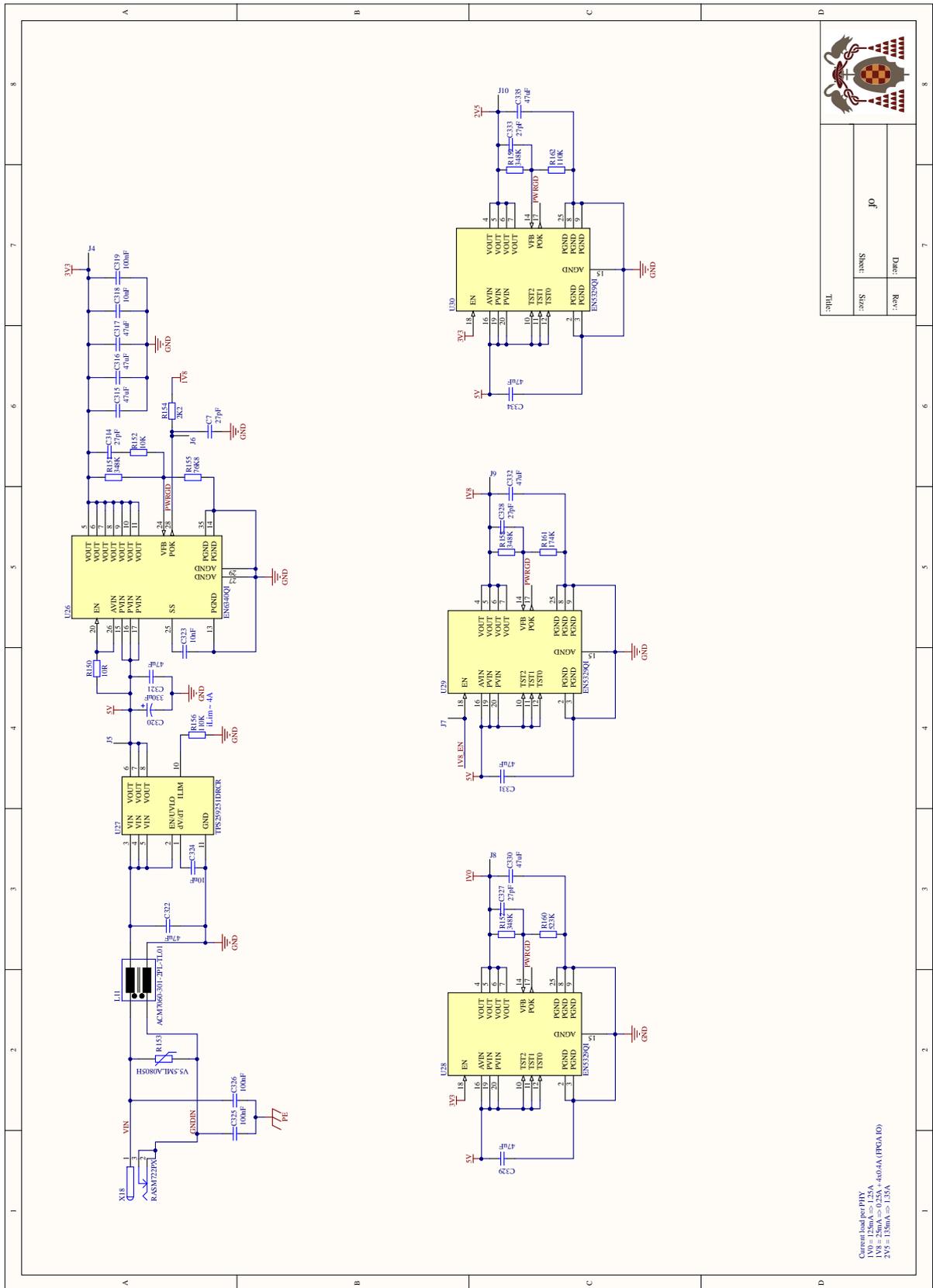


Figure A.12: Sheet 11

