A FLEXIBLE DSP-FPGA BASED PLATFORM FOR EXPERIMENTS WITH MODULAR MULTILEVEL CASCADE CONVERTERS

Lucas Koleff¹, Manoel Conde¹, Pedro Hayashi¹, Francesco Sacco¹, Kelly Enomoto¹, Eduardo Pellini², Wilson Komatsu¹, Lourenço Matakas Junior¹

¹Power Electronics Laboratory of the Escola Politécnica da Universidade de São Paulo, São Paulo – SP, Brazil
²Protection and Automation Laboratory of the Escola Politécnica da Universidade de São Paulo, São Paulo – SP, Brazil

e-mail: koleff.lucas@gmail.com, maconde@terra.com.br, phi_hayashi@outlook.com, francesco_sacco@hotmail.com, kellymingorancia@gmail.com, elpellini@usp.br, wilsonk@usp.br, matakas@pea.usp.br

Abstract – The growing use of Modular Multilevel Cascade Converter (MMCC) topologies in applications such as High Voltage Direct Current Transmission, Static Var Compensation, Alternating Current (AC) Drives and renewable energy systems, among others, turns their study relevant. However, as the number of Power Submodules in the MMCCs increases, the applied control strategies, data communication and digital hardware complexity increases as well. This paper proposes a flexible and scalable platform, that is adequate for implementing different MMCC topologies (single star, single delta or double star), communication schemes (star or ring) and control strategies (centralized or distributed). The Processing Board explores the availability of Field Programmable Gate Arrays, Digital Signal Processors and ARM microcontrollers. The Power and Measurement Submodules processing capability is provided by Complex Programmable Logic Devices and ARM microcontrollers. Description of the tasks implemented in the processing devices and experiments are presented for a test case, in which a double star topology with star communication and centralized processing strategy is tested using fixed switching frequency and equally displaced carriers.

Keywords – power electronics, modular multilevel cascade converters, control systems, FPGA, DSP.

I. INTRODUCTION

Modern applications in energy conversion systems often rely on Modular Multilevel Cascade Converters (MMCCs) [1, 2] for connection of wind power plants [3], high voltage direct current (HVDC) grids [2, 3], medium-voltage electric machine drives [2], static var generators [2], energy storage [2, 4], among others. The study of control systems, their implementation and their experimental validation for MMCCs is a relevant subject considering this scenario.

The motivation of this paper is to propose a flexible platform adequate for MMCCs experiments, which can include:

- Three topologies (single star, single delta and double star) [2];
- Two communication schemes (ring, star) [5];
- Two processing strategies (centralized and decentralized) [6].

Those possibilities are discussed in this work, and the implementation of a double star topology with star communication and centralized controller is shown in detail as a case study. In its current version, the platform is composed by 24 H-bridge Power Submodules (P-SM), 3 Measurement Submodules (M-SM) and a Processing Board (PB). The processing cores in the P-SM, M-SM and PB jointly use programmable logic devices and processors as a strategy to increase usage flexibility.

Regarding the converter topology, this paper uses Akagi’s [2] terminology, that proposes the MMCC family of converters by merging two well-known terms, the Cascade Multilevel Converters and Modular Multilevel Converters (MMCs).

The MMCCs consist in the series association of half-bridge or H-bridge converters [2]. They provide high Alternating Current (AC) or Direct Current (DC) voltage levels, high quality voltage and current AC waveforms, superior harmonic performance, higher efficiency, modularity and higher reliability in case of P-SM failure [7, 8]. Figure 1 shows the single star, single delta and double star topologies.

For the single star and single delta, the strings are composed by H-bridges without power supply (for reactive power generation [2] and active filters [9]) and with power supply (for photovoltaic generation [10], energy storage [4] and AC drives [11]). The double star topology is usually called MMC in the literature [1], can use either half or H-bridges for the P-SMs and is used for HVDC grids [3], machine drives [2], among other applications.
The increasing number of SMs in the MMCCs creates new challenges [7] due to the rising complexity of the controller hardware, of the communication structure and of the control algorithms.

Regarding the communication, [5] and [12] discuss many possibilities, including the star and ring schemes. For the star scheme [13], the main advantages of the dedicated channels between the Processing Unit (PU) and the SMs are:

- Lower requirements for the links’ speed;
- Lower delay in the data acquisition;
- Easy synchronization of the SMs, which is particularly useful for PWM strategies based on displaced carriers.

According to [14], the star scheme is not adequate for systems containing a large number of SMs because it becomes more complex and prone to complete failure in the case of PU malfunction. The ring architecture [5] simplifies the interconnection of a large number of SMs and the PUs but requires high speed communication links. Synchronization between units becomes more complex.

Besides the choice of communication schemes, processing can be realized either in a centralized or distributed way. In the centralized case, all the measurements must be sent to the PU. Some researchers propose control strategies where some tasks like, individual voltages balancing and PWM generation, can be executed locally in the SMs [6, 15]. By using decentralized processing in a ring communication scheme, it is possible to reduce the data throughput demand on the communication links.

A discussion on the processing cores in the PB, M-SMs and P-SMs follows. The implementation of control systems in power electronics usually relies on Digital Signal Processors (DSPs). However, due to restricted availability of resources, the number of SMs that can be controlled directly by a commercial DSP is limited [16]. The Field Programmable Gate Arrays (FPGAs) arise as an alternative due to their increased scalability. FPGAs feature a higher input-output (I/O) pin availability, can be completely reprogrammed according to the application demands and perform parallel processing with increased flexibility.

Therefore, by using FPGAs, it is possible to efficiently support the growth in the number of SMs in the MMCCs. Additionally, their parallel processing capability allows execution of control loops and tasks (such as data handling and signal processing) in adequate time intervals [16]. By design, the FPGAs present better performance when executing fixed-point calculations. In the case when double precision floating point calculations are done, the DSPs generally demonstrate good performance with reasonable costs. Some latest high-end FPGAs offer hardened floating-point capabilities [17] but are associated with higher costs. So, the processing core of the proposed PB features the DSP, FPGA and ARM microcontroller “triplet”. This allows the designer to freely test many possibilities of sharing the computational burden between the “triplet” components. The processing core of the M-SMs and P-SMs features the CPLD (Complex Programmable Logic Device) and ARM microcontroller “duo”. Again, the computations can be divided between those components. The PB can be the PU of a centralized control scheme or share the execution of the control algorithms with the PUs inside the P-SMs and M-SMs. The first (centralized) option is used in the case study presented in the article. The three modules are detailed in Section II.

Usually implementation details are not presented in papers due to the lack of space. The battery storage system based on a cascade H-bridge converter in [4] uses one FPGA for data acquisition, one FPGA for PWM and one DSP for centralized control. Fiber optic links are used for the data acquisition channels. The active filter based on a cascade H-bridge (19 level- three phase) in [9] uses a DSP (TMS320F2812). Nowadays real time simulation is being used for the evaluation of multilevel systems [18]. This strategy is adequate for the evaluation of the control loop performance but cannot address issues related to the communication and processing architectures.

This paper presents a flexible experimental platform that can be used for testing many combinations of communication and processing architectures for modular multilevel converters. The implementation of a star communication scheme with a centralized processing strategy is detailed in this work. The comparison of the combined FPGA-DSP system with the FPGA-only option is not analyzed in this work.

The scalability of the proposed system in changing the topology or the number of SMs with reduced reprogramming efforts is highlighted throughout the article. Experimental results obtained through a complete MMC prototype are presented to validate the proposed architecture.

The current article is an extension of the conference paper presented in [19], with more details on the circuit development, new references and results. Authorization for reusing content from that article has been granted by the IEEE.

This paper is organized as follows. Section II details the three platform parts (which are the PB, P-SMs and M-SMs). Section III details the FPGA synthesis for a test case consisting of a double star (MMC) topology with:

- Star communication scheme;
- Centralized processing strategy;
- Fixed frequency, equally displaced carriers;
- The DSP concentrating the execution of the control algorithms;
- The FPGA performing the data exchange with the DSP and the SMs, and generating the gating pulses.

Section IV details the execution of the DSP program, while Section V presents experimental results for the chosen test case. Finally, Section VI includes the conclusions.

II. SYSTEM ARCHITECTURE

In spite of the flexibility of the proposed flexibility, all signals and nomenclature are chosen for the particular test case defined in Section I. Whenever possible, the application in other topologies (single star, single delta), ring communication and decentralized control are discussed.

A. Overview of the Modular Multilevel Platform

In Figure 2 (see next page), the application example (test case) is presented, showing the interconnection of the PB with the SMs.

The case study uses 8 P-SMs per phase, resulting in a 9-level converter output voltage. Also, the phase-shifted
Both P-SMs and M-SMs feature a STM32F405 ARM microcontroller reserved for future usage as PU, eventually performing local processing tasks, allowing implementation of distributed control strategies.

The M-SMs can measure up to two currents and two voltages, as shown in Figure 4. Those measurements are presently used for the upper arm and lower currents and for the phase voltage and overall DC link voltage of the MMC. They use LEM LV-25NP and LEM LTS-6NP Hall-Effect sensors (which provide galvanic isolation) together with their respective amplifier circuits. Four A/D converters of the same model used in the P-SM are used to acquire the resulting signals. Two XC2C256 programmable logic devices generate the signals sd1...sd4 which are transmitted to the PB. Each one of those signals contains the serialized data words corresponding to the acquired current and voltage measurements.
6.146Mbit/s. The sync192k signal is the data word synchronization clock. Each from its falling and rising edges delimits one 16-bit data word in the sd lines. It has a frequency of 192 kHz. From the 16-bit data word, 14 bits represent one sample and the remaining bits indicate the status of the SM.

For the star communication, each link between the PB and the SMs is composed by two Ethernet cables attached to the RJ-45 connectors conn1 and conn2 (see Figure 3 and Figure 4). The current PB supports 27 links, allowing usage of up to 24 P-SMs and 3 M-SMs in the star communication scheme, so the usage of this scheme is adequate in the test case.

The ring scheme is accomplished by connecting the output channels of conn1 to the input channels of conn2 of the neighboring SM, and so on. Reprogramming the CPLDs is necessary. The number of linked SMs is related to the data throughput requirements, which depends on the degree of control decentralization. This discussion is outside the scope of this paper.

C. Processing Board

As introduced in Subsection A, the PB features the transceivers required for communication with all SMs. Also, it features the FPGA (Xilinx Artix 200T) and the DSP (TMS320F28335), which are part of the processing core of the PB. The development of a combined FPGA-DSP control system benefits from the advantages of both devices. The communication between them is implemented using a high-speed parallel memory bus (XINTF) [20].

Additionally, an ARM (STM32F429) microcontroller is used to handle communication with the user and to govern the startup process of the MMC. An FPGA with integrated microcontroller, also known as a system-on-a-chip (SoC) solution [21], was not adopted in this project due to additional costs and programming efforts. In Figure 5, the block diagram of the PB is shown. For the communication links with all SMs, the I2S protocol [22] is used. It utilizes the clk6M146, sync192k and sd signals for transmitting/receiving the serialized data, guaranteeing exact synchronicity of the data words. Also, it has no data overhead such as data headers or handshaking.

For the communication between the FPGA and the DSP, the 32-bit wide XINTF external memory interface bus is clocked at 150MHz. Taking all wait states (lead, active and trail) into account, a measurable data exchange rate of 800 Mbit/s is achieved in this memory bus.

Fig. 5. Block diagram of the Processing Board.

III. FPGA SYNTHESIS

A. Integration

The block diagram of the FPGA synthesis for the test case described in Section I is shown in Figure 6. It is important to note that the DSP and FPGA programs are developed in a modular way. This ought to reduce reprogramming efforts if the number of SMs is changed or they need to be rearranged for other studies (e.g. single star or single delta topologies).

The Control Module is the main part of the synthesized FPGA circuit. It features interfaces for the PWM Modules, the Acquisition Modules and the DSP memory bus.

The PWM modules are responsible for generating the switching signals (pwm and enable) for the P-SMs, while the Acquisition Modules are responsible for receiving the serialized measurement data signal (sd) from the SMs, which are put in synchronization through the clk6M146 and sync192k signals. In the control cycle, the DSP is responsible for the calculation of the reference values for the modulation, by taking the measured variables into account.

The sync16 signal is used internally in the PB for synchronization of the sampling and control loop calculation with the PWM carriers. Its rising edges indicate that a carrier has reached its peak value and the control loop is updated with a new sampling event. The sync16 signal is also outputted to the DSP. It has a frequency of 16 kHz and determines the sampling frequency of the MMC control loop.

The dsp_finished and fpga_finished signals are needed in the process of memory bus access coordination. Both DSP and FPGA have access to the memory bus as they need to transfer data between themselves. However, they cannot perform read/write operations simultaneously as they share the bus. Therefore, the dsp_finished is set to high when the DSP needs to access the memory bus. During this period, the FPGA cannot access the bus. In the same way, the fpga_finished is set to high when the FPGA needs to access the bus, inhibiting the DSP from using it.

Fig. 6. Block diagram of the FPGA synthesis.
The Mixed-Mode Clock Manager (MMCM) is a feature of the Xilinx 7-Series FPGAs and is used to generate the System Base Clocks. Based on a reference clock provided by an external oscillator in the FPGA, the MMCM generates the 50MHz and 150MHz base clock signals using a phase-locked loop (PLL).

The System Base Clocks are used by the Clock Divider to generate the clk6M146 and sync192k signals. They are also used by the Control Module to generate the PWM base clock and the sync16 signals. All carrier signals used by the PWM Modules are generated by the Carrier Generator block, which is detailed in Subsection E.

By changing the number of PWM and Acquisition Modules in the FPGA, it is possible to adjust the number of SMs controlled by the PB, limited by the number of digital communication links of the PB.

The Control Module and its operation is described in detail in Subsections B and C, the Acquisition Modules in Subsection D and the PWM Modules in Subsection E.

### B. Control Module

The execution of the main steps of the MMC control cycle is orchestrated by the Control Module. Those steps are divided roughly in the following:

- Data acquisition step;
- Data processing step;
- PWM update step.

Both custom “Very High-Speed Integrated Circuit Hardware Description Language” (VSHIC-HDL or VHDL) and “Xilinx Intellectual Property” (IP) blocks are used in the design. It is conceived to be scalable, requiring minimum reprogramming efforts to be adapted for MMC architectures with a different number of levels. Its main functional blocks are presented in Figure 7.

The custom VHDL block named “Control Circuitry” coordinates the operation of the Control Module. It performs the following tasks in short:

- Synchronization of the FPGA and DSP activities along the control cycle of the MMC;
- Arbitration of the access to the memory blocks (DSP_Write and DSP_Read) inside the FPGA;
- Reading of new data coming from the Acquisition Modules and making it available in the DSP_Read memory for DSP access;
- Retransmission of the new modulation references written by the DSP into the DSP_Write memory, which are applied to the PWM Modules;
- Generation of synchronization and clock signals used in the control cycle and PWM carrier generation.

The DSP_Read memory block stores the measurement data provided by the Acquisition Modules. In this implementation, thirty-four voltage and current measurements are provided by the P-SMs and M-SMs and processed by the DSP. The DSP_Write memory block stores the modulation references signals provided by the DSP to be retransmitted to the PWM Modules. Currently, twenty-four modulation reference signals are used in this MMC implementation. Each of the DSP_Write and DSP_Read memory blocks has a data width of 32-bit wide and a data depth of 128 positions and are implemented using Xilinx IP. Since the system works with 16-bit data words, each memory position stores a pair of data words. The test case uses 17 positions of the DSP_Read and 12 positions of the DSP_Write blocks.

The memory bus address and control line multiplexers are custom VHDL blocks named Mux Addr & Ctrl. They control the data flow between the DSP and the memory blocks in the tristate data bus.

![Fig. 7. Block diagram of the Control Module.](image)

### C. Control Module Operation

The sequence of events in the main steps of the Control Module operation is shown in the flowchart of Figure 8 (see next page). The Control Module is interrelated with the execution of DSP program, which is described in detail in Section IV.

The sync16 signal is the basis for the synchronization of the control cycle, determining the effective sampling and control update rates. The control cycle begins at the “Wait for New Cycle Step”, in which the program waits for a rising edge of the sync16 signal.

Then, on the “Prepare for New Cycle Step”, the “Control Circuitry” checks if the dsp_finished signal is low (indicating that the DSP is idle). It also checks if all 34 Acquisition Modules have fresh data available, indicated by the READY lines of the READY & HOLD bus. If both conditions are fulfilled, the execution of the next step is initiated.

In the “Acquisition Step”, the “Control Circuitry” assumes the control of the memory blocks by setting the fpga_finished...
signal to high (indicating that it is accessing the memory blocks). Also, the HOLD lines of the READY & HOLD bus are set to high to instruct the Acquisition Modules to stop updating their outputs. In the continuation, it will sequentially read the data provided by all the Acquisition Modules and copy it into the DSP_Read memory block. The selection of the corresponding acquisition module pair being read is performed by an internal 17x32-bit multiplexer.

After all fresh 17 acquisition data words are successfully stored in the DSP_Read memory the “Acquisition Step” is finished. Then, the fpga_finished line is reset as well as the HOLD lines from the READY & HOLD bus.

The DSP detects the negative transition in the fpga_finished line. This initiates the “DSP Processing Step”, which is detailed in Figure 13. In short, the DSP takes control of the memory blocks by setting the dsp_finished line to high. Then, the DSP reads the acquisition data in the DSP_Read memory and performs the necessary calculations. Next, the DSP writes the new references for the PWM modulation into the DSP_Write memory. Once all write operations are finished, the DSP resets the dsp_finished line, releasing the access to the memory blocks and finishing the “DSP Processing Step”.

While the DSP is performing its tasks, the FPGA monitors the dsp_finished signal (generated by the DSP). The falling edge of the dsp_finished signal is sensed by the FPGA, which then goes into the next step.

In the “PWM Update Step”, the “Control Circuity” sets the fpga_finished line to high again, assuming control of the memory blocks. Then, it sequentially reads the 12 new modulation data pairs from the DSP_Write memory and loads the references into the shadow registers from each pair of PWM Modules. The selection and loading of the respective module pairs is performed in the Select & WE bus by setting the corresponding Select line to high and pulsing the WE (write enable) line.

After all the 12 data pairs containing new modulation references have been transmitted to the PWM Modules, the “PWM Update Step” is finished and the final step is initiated. In the “Finish Cycle Step”, the “Control Circuity” resets the fpga_finished line releasing control of the memory. Then, it goes directly into the “Wait for New Cycle Step”.

D. Acquisition Module

The block diagram of the Acquisition Module is shown in Figure 9. It features a 12S decoder, which takes the serial data sd, the sync192k synchronization and the clk6M146 signals as inputs. The decoder uses a shift-register to parallelize the incoming data signal and outputs the received 16-bit word.

When a complete data word is received, the DATA_GOOD pulse is generated indicating the completion of a Reception cycle. The Input Latch stores the 16-bit word and splits it into a 14-bit acquisition data word and 2 status bits. Also, the Median Filter and the Status Handler are triggered.

The Median Filter is developed in the C programming language and implemented using the “Xilinx High Level Synthesis” (HLS) development tool. The order of filter used in this implementation is 7, but it can be changed according to the application. It can also be completely bypassed if required.

The status bits are processed by the Status Handler to signal the occurrence of communication or internal SM errors. To prevent false error event triggering caused by signal noise, the error signals are filtered inside the Status Handler. A status bit x (x=0 or x=1) is confirmed by assuring that its value is kept constant for at least 3 consecutive samples.

When the Median Filter has finished its internal processing, it pulses the DONE line and makes the filtered 14-bit data word available for the OutputLatch. Finally, the Output Latch detects the pulse coming from the Median Filter and stores the filtered data together with the status/error signals in a 16-bit word, which is then available at the DATA_OUT output. The READY signal is set to high, as the processed data is available to the Control Module. The HOLD signal disables the update of the Output Latch.

The 12S Decoder, Input Latch, Status Handler and Output Latch are all custom VHDL blocks.
E. PWM Generation Module

In the present FPGA synthesis, the PWM generation is divided in carrier generation and modulation. In Figure 10, the block diagram of the PWM Module is presented.

On the left side, the Shadow Latch is seen. It takes the 16-bit reference data word provided by the Control Module as input, as well as the signals from the WE and Select bus. It stores the reference word when the respective Select line is high and the WE line is pulsed.

Then, when there is a positive transition in the sync16, the Internal Latch stores the output from the Shadow Latch itself. Consequently, the PWM references are calculated and stored, but only updated at the correct time instant. This is important for the correct synchronization of the PWM generation with the rest of the MMC control cycle.

Finally, the PWM logic compares the internal reference with the carrier input line COUNT_INx to generate the modulated signal.

It is important to notice that from the 16-bit received data word, only 11 bits are actually used for the PWM reference. From the remaining bits, two are reserved for control functions and the most significant bit (MSB) is used for the enable signal, which is responsible for activating or deactivating the GDs on the P-SMs.

Fig. 10. PWM Module block diagram.

Eight triangular wave generators are used in this implementation to generate the equally phase-shifted carrier signals COUNT1...COUNT8 based on the clk8M146 (PWM base clock) and synchronization sync16 signals provided by the Control Module. Each carrier corresponds to a P- SM on each phase. This block can be easily adapted to generate a different number of carriers, to change their relative phase shift and switching frequency. The Carrier Generator Block is depicted in Figure 11. By using the equally shifted carriers, it is possible to achieve harmonic cancellation in the MMC branches [23].

Fig. 11. Carrier Generator block diagram.

The waveforms of the generated carriers are shown in Figure 12. Each of the triangular waves has a frequency of 2 kHz, with the counters being updated in the range between 0 and 2048 at the frequency of 8.146 MHz. It is possible to note that the peaks of the carriers are synchronized with the sync16 signal as expected. All blocks involved in the PWM generation presented here are written in custom VHDL code.

Fig. 12. Waveforms of the generated carriers.

IV. DSP PROGRAM

A. Execution Steps

The DSP executes the calculations of the MMC control loop, using the measurement data received from the SMs to calculate the new modulation references. The sequencing of the DSP program execution steps is presented in Figure 13.

The execution of the DSP program is started when an interrupt service routine is triggered by a rising edge transition of the sync16 signal line. In this moment, as seen in Section III, the FPGA program is updating the internal latches from the PWM Modules and copying the acquisition data into the DSP_Read memory while keeping the fpga_finished line high. When finished, the FPGA lowers this line, signaling that the DSP should continue the execution of its program.

On the other hand, the DSP waits up to 50 internal clock cycles for the FPGA to lower the fpga_finished signal. If timeout occurs, an error is signaled as the FPGA has not been able to properly finish its tasks on time.

If the FPGA does not exceed this time interval, the DSP continues executing its routines and raises the dsp_finished to
gain access to the memory blocks.

Next, the DSP copies the acquisition data from the `DSP_Read` memory into its internal RAM. Then the preprocessing step is executed. The 32-bit words are split into two 16-bit words, converted to floating point values and the appropriate scaling and offset factors are applied.

This enables the execution of the MMC control loop. The new PWM references are then calculated. In the Postprocessing step, the new references are converted to adequate 16-bit fixed-point values and concatenated into 32-bit words, which are then written in the `DSP_Write` memory.

Then, the DSP lowers the `DSP_finished` line, finishing the execution of its routines. Next, the FPGA is going to proceed with the conclusion of its own steps by copying the references from the `DSP_Write` memory into the corresponding shadow registers from the PWM Modules.

**B. Description of the MMC Control System**

The focus of this article is on the platform and DSP-FPGA programs that allow the execution of the control system and not on the design of the loops themselves. The control loop [24, 25] is designed for three-phase MMC converters and is based on the abc reference frame. An overview of the control loop (per phase) for the case study defined in Section I is provided in the block diagram of Figure 14.

In the block diagram, the capacitor voltages of the P-SMs are represented by \( V_{C1} \), the outputs of the individual voltage balancing loops by \( \Delta V_i \), and their PWM references by \( V_{SM} \). Note that the Greek letter \( \Delta \) in \( \Delta V_i \) does not mean a variation, only representing the name of the signal itself. The upper arm SMs are indexed as \( k=1...4 \) and the lower arm SMs as \( k=5...8 \).

The upper arm and lower arm currents of the MMC are indicated by \( i_a \) and \( i_b \), respectively. The upper arm total voltage is defined as \( V_{C1} \) and the lower arm total voltage as \( V_{Cn} \). Those are calculated by adding the DC bus voltages of the P-SMs in the upper and the lower arm, respectively. The total overall DC link voltage is \( V_{DC} \).

The measured circulating current is represented by \( i_z \), while its reference is composed by two parts: \( i_{z,ref}(\text{oscillating}) \) and \( i_{z,ref}(\text{average}) \). They are generated by the total arm capacitor voltage difference and sum loops, respectively. For tracking those references, the circulating current loop generates the voltage reference \( v_z \).

The reference signal for the output current \( i \) is given by \( i_{ref} \).

In order to track this current reference \( (i_{ref}) \), the output current loop is used. This loop generates the \( v_z \) voltage reference and uses the AC grid (or load) voltage \( v \) internally to perform feedforward.

The tuning of the control loop parameters is out of the scope of this paper. The interested reader can refer to [24, 25] for more details. The calculation of the reference signals is briefly examined next. In Figure 15, the calculation of the circulating current references by the voltage sum and difference loops is shown. A moving average filter MAF is used together with the proportional controllers \( P_z \) for the sum and \( P_a \) for the difference loops. For each phase, \( v_{PLL//} \) is an unitary amplitude signal, in phase with the corresponding positive sequence of the fundamental frequency components of the grid voltage.

![Fig. 15. Block diagram of the voltage sum and difference control.](image)

In Figure 16, the circulating and output current control loops are presented, and their respective PI controllers (\( P_z \) and \( P_i \)) can be seen. The measured current signals are compared against their respective references to perform tracking. The output of the controllers is divided by \( N \) to adjust them to the voltage level of the SMs.

![Fig. 16. Block diagram of the circulating and output current control.](image)

In Figure 17, the voltage balancing for the upper and lower arm SMs using the proportional controller \( P_b \) is shown. The individual voltages \( V_c \) are compared against the instantaneous averages of the upper arm \( V_{C1}/N \) and of the lower arm \( V_{Cn}/N \) SM voltages in order to perform balancing.

![Fig. 17. Block diagram of the voltage balancing control.](image)

Finally, in (1) the calculation of the \( k^{th} \) P-SM PWM reference is presented. For the upper arm P-SMs, \( \alpha=-1 \) and for the lower arm SMs, \( \alpha=1 \).

\[
v_{SMk}=v_z+\alpha v_S+\Delta v_X
\]
The control loop parameters used in the case study are shown in Table I. Those are calculated considering the MMC parameters summarized in Table II.

### V. EXPERIMENTAL RESULTS

#### A. Overview

For validation of the control system, the MMC prototype shown in Figure 18 is used. It is possible to see the 24 P-SMs, the 3 M-SMs and the PB. The Ethernet Cat-5e cables are used to carry the LVDS signals between the transceivers of the SMs and those of the PB.

![MCC prototype](image)

Fig. 18. MMC prototype.

The parameters relevant to this case study are presented in Table II, including the branch inductance $L$, the triangular carrier frequency $f_{tr}$, the load resistance $R_L$, the local DC capacitance $C_{cap}$, the local DC voltage $V_{cap}$, the sampling frequency $f_S$, the converter nominal output rms voltage $V_{cap}$, the converter nominal power rating $S_N$ and the number of P-SMs in the arms $N$. As seen in Figure 12, the 8 triangular carriers are equally shifted in the PWM modulation.

### TABLE I

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$</td>
<td>5 mH</td>
</tr>
<tr>
<td>$C_{cap}$</td>
<td>1540 μF</td>
</tr>
<tr>
<td>$N$</td>
<td>4</td>
</tr>
<tr>
<td>$f_S$</td>
<td>16 kHz</td>
</tr>
<tr>
<td>$R_L$</td>
<td>33 Ω</td>
</tr>
<tr>
<td>$V_{cap}$</td>
<td>30 V</td>
</tr>
<tr>
<td>$f_{tr}$</td>
<td>2 kHz</td>
</tr>
<tr>
<td>$V_N$</td>
<td>127 V</td>
</tr>
<tr>
<td>$S_N$</td>
<td>2 kVA</td>
</tr>
</tbody>
</table>

#### B. Results for the FPGA Synthesis

The results for the programmable logic synthesis show that an expansion of the system in terms of number of SMs would be possible from the point of view of the FPGA resources. A summary of utilization of the Artix-7 200T resources is shown in Table III.

### TABLE II

Control Loop Tuning Parameters

<table>
<thead>
<tr>
<th>Block</th>
<th>Parameter Values</th>
<th>Block</th>
<th>Parameter Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_C$</td>
<td>$K_C = 1.13 \times 10^{-4}$</td>
<td>$P_A$</td>
<td>$K_A = 1.36 \times 10^{-3}$</td>
</tr>
<tr>
<td>$P_A$</td>
<td>$K_A = 1.36 \times 10^{-3}$</td>
<td>$P_S$</td>
<td>$K_S = 13.33$</td>
</tr>
<tr>
<td>$MAF$</td>
<td>267 samples (length) 1/60 ms (window)</td>
<td>$P_S$</td>
<td>$K_S = 1.5$</td>
</tr>
</tbody>
</table>

### TABLE III

Artix-7 200T FPGA Resource Utilization Summary

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPGA I/O Pin Usage</td>
<td>139/285 (48.77%)</td>
</tr>
<tr>
<td>FPGA Slice Register Usage</td>
<td>7737/269200 (2.87%)</td>
</tr>
<tr>
<td>FPGA Look Up</td>
<td>6139/134600</td>
</tr>
<tr>
<td>Table Usage</td>
<td>1/365</td>
</tr>
<tr>
<td>RAM Tile Usage</td>
<td>4 (0.6%)</td>
</tr>
</tbody>
</table>

The timing of all the control cycle steps is measured and the obtained values summarized in Table IV. According to the obtained results, the sequential processing in the DSP consumes most of the control cycle time. Also, there is spare processing time that can be used for executing more complex control algorithms, as 38% of the total time available in the control cycle is used. This also shows the viability to explore the execution (totally or partially) of the control loops in the FPGA, eventually parallelizing the calculations.

### TABLE IV

Measured Control Cycle Timing Summary

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquisition Step (FPGA)</td>
<td>504ns 0.81%</td>
</tr>
<tr>
<td>PWM Update Step (FPGA)</td>
<td>644ns 1.03%</td>
</tr>
<tr>
<td>DSP Processing Step</td>
<td>22750ns 36.4%</td>
</tr>
<tr>
<td>Total Time Available per Cycle</td>
<td>62500ns 100%</td>
</tr>
</tbody>
</table>

In Figure 19 and Figure 20, it is possible to see one of the P-SMs and the M-SMs used in the platform, respectively, with the main components being highlighted.

![Detail of the P-SM](image)

Fig. 19. Detail of the P-SM.

![Detail of the M-SM](image)

Fig. 20. Detail of the M-SM.

Next, in Figure 21, the PB is shown, and the main components are highlighted. The FPGA is positioned in the center of the board. This equalizes the propagation delays between the FPGA lines and the LVDS transceivers, which are distributed along the borders of the circuit board together.
with the corresponding RJ45 connectors. Additionally, the propagation delays (mainly in the XINTF bus) are minimized to guarantee maximum data throughput. Therefore, the DSP and ARM microcontroller are positioned as close as possible to the FPGA. Also, careful layout is performed to reduce the occurrence of crosstalk between the communication lines.

In order to verify the modulation, the DSP control loop is set to generate a sinusoidal reference waveform with a frequency of 60Hz. The amplitude of this voltage reference signal is 100% of the maximum modulation range.

The operation is tested in each phase individually. Figure 22 shows the upper (-vep) and lower (vep) arm output voltage measurements (showing 5 levels each) for phase B. The final MMC equivalent output voltage (which has 9 levels) is given by the addition of the arm voltages waveforms (vep-vep) using the mathematical function of the scope.

The mathematical functions from the scope measurements (showing 5 levels each) for phase B. The final MMC equivalent output voltage (which has 9 levels) is given by the addition of the arm voltages waveforms (vep-vep) using the mathematical function of the scope.

For verification of the switching harmonics cancellation, the spectra of the upper and lower arm voltages are calculated from the waveform data and shown in Figure 23. It is possible to see that, even with the carriers having a frequency of 2 kHz, the switching harmonics are centered at 8 kHz and 16 kHz, as expected. Those harmonics are multiples of order 4 and 8 from the original carrier frequency.

The same analysis is performed for the MMC equivalent output voltage, and the obtained spectrum is shown in Figure 24. The harmonics centered at 8 kHz are cancelled. As expected, only the components around 16 kHz remain. It is important to note that in Figure 23 and also in Figure 24, the fundamental frequency component is not fully represented due to the scaling of the vertical axis.

Next, in Figure 26, the same experiment is performed for an output current of 0.4A and a circulating current of 0.5A and in Figure 27 for an output current of 0.6A and a circulating current of 0.6A. The mathematical functions from the scope are used to obtain the load current from the subtraction of the arm currents. The arm currents are composed of a circulating current (constant) component and an output current (sinusoidal) component. As expected, the circulating current is not present in the load current [6, 25].
Also, the voltage balancing control is validated. The transient waveforms of the individual P-SMs capacitors voltages in one MMC branch during the start-up process are shown in Figure 28. A reference step of 7V is applied in this test. The obtained waveforms show that the capacitor voltages are being correctly balanced and reach steady state in 50ms.

VI. CONCLUSIONS

This paper described the development of a platform for MMCC (single star, single delta, double star) topologies with star or ring communication schemes and centralized or decentralized control strategies. The hardware comprises the PB, the P-SMs and the M-SMs. The PB integrates an FPGA, a DSP and an ARM microcontroller as possible PUs. The SMs offer a CPLD and an ARM microcontroller each.

A test case for the implementation of a double star (MMC) topology with star communication and centralized control is presented in detail to validate the designed platform. In this case, the DSP concentrates the floating-point calculation burden from the control loop. The FPGA is then concentrated in complex peripheral logic tasks. It performs circuit synchronization, high speed data transfer, memory blocks, multiple parallel measurement data acquisition and PWM modulation. This combination is proven to be an effective design, and the implemented circuits are tested for this case study. The DSP and FPGA programs are designed in a modular way. This can allow the reduction of the reprogramming efforts when using different numbers of SMs/levels or studying different topologies by reusing and reconfiguring the modules described in Section III.

Experimental results have validated the correct operation of the proposed platform. The obtained resources utilization levels and measured timing of the control cycle show that the PU used supports MMCC implementations with more complex control loops.

APPENDIX

A snippet of the output current control loop code in the DSP is shown in Figure 29 to give a sight of the implementation.

![Fig. 26. Closed-loop test waveform (arm currents and load current).](image1)

![Fig. 27. Closed-loop test waveform (arm currents and load current).](image2)

![Fig. 28. Transient waveforms of the capacitor voltages.](image3)

**ACKNOWLEDGEMENTS**

The authors would like to thank Texas Instruments, STMicroelectronics and TDK/EPCOS for their support. This work was funded by the FAPESP research grants 2016/16542-5 and 2016/01930-0 and by the CNPq grants 306970/2015-5 and 311789/2014-5. This study was financed in part by CAPES - Finance Code 001.

REFERENCES


**BIOGRAPHIES**

**Lucas Koleff**, received his B.Sc. degree from the University of São Paulo (USP) and his D.D. M.Sc. degree from the Technical University of Darmstadt in 2015. He is currently working towards his Ph.D. degree at the USP.

**Manoel Conde de Almeida**, received his B.S. degree from University of Campinas in 1986 and his MBA degree from the USP in 2002. He is currently working towards his M.Sc. degree at the USP.

**Pedro Henrique Itio Hayashi**, received his B.Sc. and M.Sc. degrees from the USP in 2015 and 2019, respectively. He is currently looking forward to starting his Ph.D. course.

**Francesco Sacco**, received his B.Sc. degree from Pontifical Catholic University of São Paulo in 2008. He is currently working towards his M.Sc. degree at the USP.

**Kelly Caroline Mingorancia de Carvalho Enomoto**, received her B.Sc., M.Sc. and Ph.D. degrees from the USP in 2012, 2015 and 2019, respectively.

**Eduardo Lorenzetti Pellini**, received his B.Sc., M.Sc., and Ph.D. degrees from the USP in 2000, 2005 and 2010, respectively. He is currently an Assistant Professor at the USP.

**Wilson Komatsu**, received his B.S., M.S., and Ph.D. degrees from the USP in 1986, 1992 and 2000, respectively. He is currently an Associate Professor at the USP.

**Lourenço Matakas Junior**, received his B.S., M.S., and Ph.D. degrees from the USP in 1984, 1989 and 1998, respectively. He is currently an Associate Professor at the USP.