

FPGA-based Implementation of Modular Multilevel Converter Model for Real-time Simulation of Electromagnetic Transients

Mahmoud Matar, Dominic Paradis and Reza Iravani

Abstract-- This paper presents the real-time simulation of modular multilevel converter (MMC) systems on a hybrid simulation platform. An FPGA-based real-time simulator is used to augment the capabilities of a processor-based simulator to enable the real-time simulation of MMC-based systems. The FPGA-based simulator is used to simulate the MMC while the rest of the power system is simulated on a processor-based simulator from RTDS technologies. The FPGA-based simulator has a massively parallel, customized, hardware architecture that is tailored to the solution of the mathematical model of the modular multilevel converter.

The paper also presents the implementation of the sorting and submodule selection techniques required to maintain the capacitor voltages balanced.

Keywords: Modular multilevel converter, FPGA, real-time simulation, voltage balancing.

I. INTRODUCTION

THE modular multilevel converter (MMC) is an emerging high power switch-mode converter that offers many technical and economic advantages especially for high-voltage direct current (HVDC) applications. As compared to the other voltage sourced converter (VSC) topologies, the MMC (i) provides high quality, nearly ideal sinusoidal, AC-side voltages and thus significantly reduces the size of the required filters or even completely eliminates them, (ii) operates with a much lower switching frequency and thus reduces the switching losses, (iii) provides higher reliability due to the presence of redundant submodules, (iv) is easily scaled up to support higher voltage levels by increasing the number of submodules per arm, and (v) causes much lower high frequency noise [1]-[4].

However, the MMC imposes a new challenge for power systems simulation tools. Due to the presence of large number of switching elements, a couple of thousands per converter, not only does the system model become excessively large but there is also an additional excessive computational burden

imposed on the simulation tool due to the inherent switching nature of the MMC. As such, simulation of a MMC is computationally demanding and time consuming which hinders the simulation of systems with realistic size converters in real-time.

Due to their highly parallel structure, the presence of dedicated high-performance DSP blocks, their inherent capability to support customized hardware designs, and their versatile programmable interconnect structure, FPGAs are increasingly becoming a mainstream implementation platform for real-time power system simulators [5]-[9]

This paper extends the parallel implementation methodology of [7]-[9] and presents the design and implementation of a massively parallel customized hardware architecture tailored to the solution of the mathematical model of the MMC. The developed implementation methodology exploits all the levels of parallelism inherent to the MMC model.

In this work, the developed FPGA-based simulator for the MMC is used to augment the capabilities of a processor-based real-time simulator to enable it to simulate MMC-based systems in real-time. Only the MMC converter model is simulated on the FPGA whereas the rest of the power system model is simulated on a real-time simulator from RTDS technologies.

The developed real-time simulator is primarily intended for hardware-in-the-loop testing of MMC control platforms. The need for such testing arrangement stems from the fact that a control platform has to be tested prior to installation and commissioning. In addition, the complexity, size, and mixed-mode structure of a physical control platform do not readily lend it for accurate representation in an off-line simulation environment. As such, the ability to connect a physical controller to a real-time simulator is very useful as it can provide a very important insight into the operational limits of the converter controller.

The rest of this paper is organized as follows. Section II briefly presents the topology and basic operating principle of an MMC. Section III presents the architecture of the hybrid real-time simulator and the FPGA-based implementation of the MMC model. Section IV presents the capacitor voltage balancing technique, the modulation strategy, and their implementation aspects. Section V presents the real-time simulation of a 400 levels 3-phase MMC system. Conclusions are stated in section VI.

Mahmoud Matar, Dominic Paradis and Reza Iravani are with the Energy Systems group, Electrical and Computer Engineering Department, University of Toronto, Toronto, Ontario, Canada. (E-mail: mahmatar@ieee.org, dominic.paradis@mail.utoronto.ca, and iravani@ecf.utoronto.ca).

II. MMC TOPOLOGY AND BASIC PRINCIPLE OF OPERATION

Fig. 1 shows the basic structure of a three-phase MMC. Each phase has two arms; an upper arm and a lower arm, with N submodules per arm. The arm submodules are connected in series to generate a multilevel AC output voltage. The voltage and power levels of an MMC can be scaled up by including a larger number of submodules per arm [4].

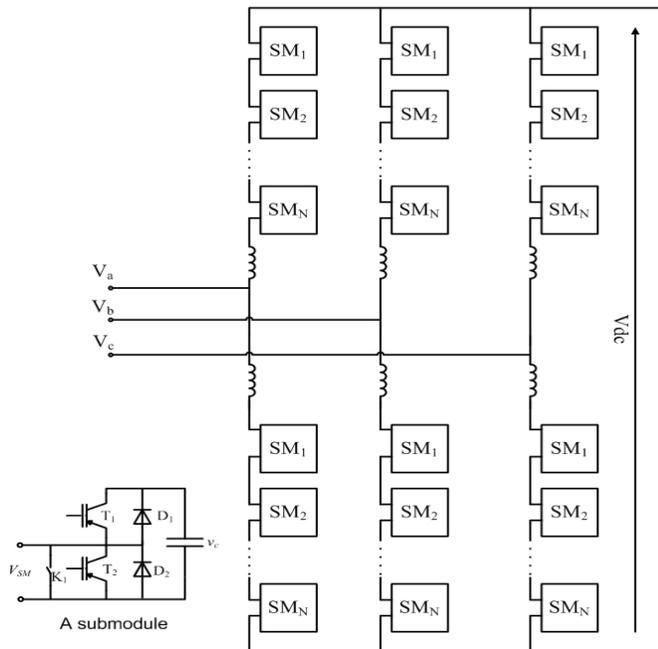


Fig. 1. Basic structure of an MMC.

The basic building block of an MMC is the half-bridge submodule. As shown in Fig. 1, each submodule has two IGBTs, two anti-parallel diodes, a bypass switch, and a capacitor. For in-service submodules, only one IGBT is ON at a given instant. When T_1 is ON, the output voltage V_{SM} is equal to the capacitor voltage V_c and when T_2 is ON the output voltage is zero. The bypass switch is used to (i) bypass redundant submodules during normal operation and (ii) bypass faulty submodules in the event of submodule failure [4].

The total voltage of an arm is the summation of the capacitor voltages of the ON submodules. As such, the output voltage is synthesized by turning the available submodules, of the upper and lower arms, ON/OFF in such a way to follow the desired reference voltage.

III. SIMULATOR ARCHITECTURE

Fig. 2 shows a high level functional block diagram of the real-time simulator. An FPGA-based real-time simulator is used to augment the simulation capabilities of a processor-based real-time simulator from RTDS technologies. The FPGA-based simulator is interfaced to the RTDS via a high throughput low latency fiber optical link. The RTDS utilizes multi-rate simulation techniques and provides the ability to include small time-step modules, with a time-step of 2.5 μ s,

within the standard 50 μ s time-step solution [10]. In this work, the FPGA-based simulator is interfaced to the RTDS small time-step module. In this hybrid simulator, the RTDS is used to simulate standard power system components, e.g., transformers and transmission lines, whereas the computational-demanding MMC model is simulated on the FPGA-based simulator.

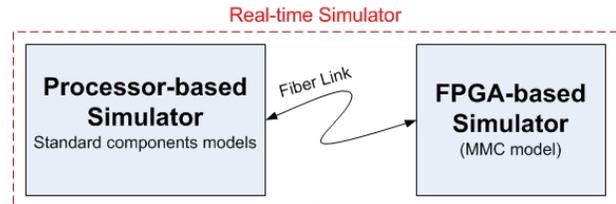


Fig. 2. Functional block diagram of the real-time simulator.

The keys for enabling the real-time simulation of a realistic size MMC are to:

- adopt a computationally-efficient MMC model, and
- parallelize the solution of the mathematical model of the MMC.

Due to the presence of hundreds of submodules per arm in a realistic MMC, e.g., the France-Spain MMC-HVDC [3] has 400 submodules per arm, the direct application of the standard modeling approach, where each switch is replaced by a small impedance during the ON state and a large impedance during the OFF state, generates an excessively large admittance matrix which is computationally demanding and time consuming. As such, the modeling approach adopted in this work is based on the concept of time-varying Thevenin equivalent [2]. Although the switches per submodule are still modeled based on the two-valued impedance approach, only the Thevenin equivalent per MMC arm is included in the admittance matrix.

In each simulation time-step, the Thevenin equivalent for each submodule is first evaluated based on the ON/OFF state of its switches and then the Thevenin equivalent per arm is constructed by computing the algebraic sum of the Thevenin equivalents for all the submodules per arm [2].

For a given submodule, the solution is dependent solely on the external gating signals, the arm currents, and the submodule capacitor voltage from the previous time-step. As such, the solution of a given submodule is independent from the other submodules and thus the solution of all the MMC submodules can be conducted in parallel. Hence, the computational engine, implemented on the FPGA, has a number of parallel processing elements. Each processing element is responsible for solving the equations for a single submodule only. For the complete arm equivalent, a parallel adder tree is implemented to compute the algebraic sum of all the submodules per arm.

Fig. 3 depicts the adopted method for interfacing the solution of the MMC arms, conducted on the FPGA, with the solution of the rest of the power system, simulated on the RTDS. This method takes advantage of the presence of the

arm inductances to decouple the solution. On the RTDS side, a simplified equivalent representation of the MMC is connected directly to the power system being simulated. This equivalent has six externally controlled voltage sources and six resistors. On the FPGA side, the detailed solution of the MMC arms is performed. The FPGA receives the switching commands from the controller as well as six currents, corresponding to the MMC inductor currents, from the RTDS. Based on these currents and the switching commands, the FPGA solves the MMC and updates the 6 controlled voltage sources on the RTDS.

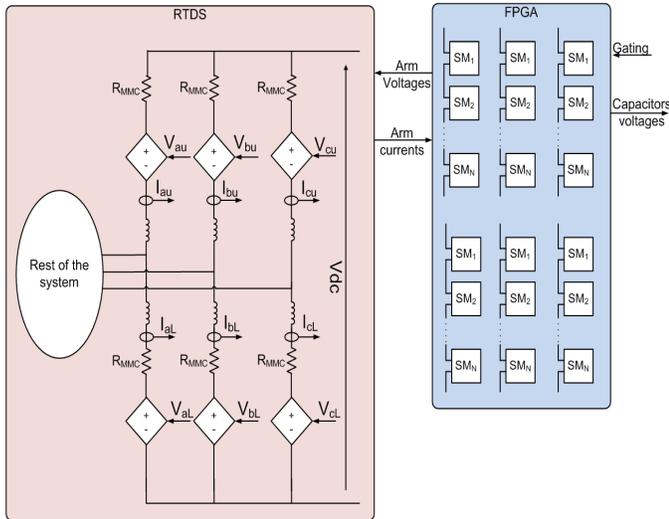


Fig. 3. Partitioning of the solution on the FPGA and the RTDS

IV. MODULATION AND CAPACITOR VOLTAGE BALANCING

For proper operation of the MMC, the capacitor voltages of all submodules have to be balanced and kept as close as possible to the nominal value. Failure to maintain the voltages balanced not only distorts the MMC output voltage but also can result in equipment damage if individual module voltages fluctuate outside of the rated values of the equipment.

For a given submodule, the change in its capacitor voltage is dependent on its ON/OFF state, as well as the magnitude and direction of the arm current. When the submodule is ON, the capacitor voltage increases (decreases) if the arm current is flowing into (out of) the submodule. On the other hand, if the submodule is in the OFF state, the capacitor voltage remains unchanged. This fact can be used to balance the submodules capacitor voltages. If a submodule voltage is lower (higher) than the average voltage per arm, it should be switched ON (OFF) when the arm current is such that it will charge the submodule capacitor, and should be switched OFF (ON) when the arm current reverses direction.

Based on the aforementioned discussion, the submodules have to be ranked based on their voltages. Ideally, if the current direction is such that it will cause the capacitors voltages to increase, then the required number of submodules should be turned ON starting from the submodule with the lowest voltage and up. If the current direction is reversed, the

ON submodules should be immediately replaced by the submodules with the highest voltages. This technique would provide very low harmonic distortion, and excellent voltage balancing, keeping the voltage evenly distributed between the submodules. However, this comes at the expense of very high switching losses. As such, other techniques that require a lower switching rate while maintaining acceptable capacitors voltage regulation must be used. The following subsections present the modulation technique, sorting and module selection algorithms adopted in this work.

A. Modulation Technique

Several switching techniques based on pulse width modulation (PWM) techniques, such as the phase-disposition PWM (PD-PWM) [11] and the phase-shift PWM (PS-SPWM) [12], have been proposed to be used with MMCs. These techniques use multiple triangular waveforms to control each submodule individually. Based on these techniques, it is possible to control the voltage of the submodules by adjusting the reference voltage used for individual comparison [13]. However, these techniques require individually controlling each submodule which becomes very cumbersome to implement when the MMC has large number of submodules per arm. Alternatively, the nearest level control (NLC) [2], which determines the number of submodules to be activated based on a staircase version of the reference voltage, has a much simpler implementation. However, the low switching rate associated with the NLC technique might cause large capacitor voltage fluctuations [4].

In this work, a modified version of the PD-SPWM is adopted. Instead of controlling each submodule individually, the result of the PD-SPWM algorithm is used to determine how many submodules per arm should be connected at any given instant. Submodules are only switched when the switching command changes. The submodule to be connected or bypassed is selected appropriately to balance the submodule voltages. To minimize the required number of triangular waveforms and the required resources, instead of using separate waveforms to generate gating signals for each submodule, the technique used here uses the voltage command and calculates the required number of submodules needed to synthesize it. The integer part of the result is used as is, and the fractional part is compared to a single triangular waveform. Fig. 4 shows the block diagram of the implementation. The modified PD-SPWM has a higher switching rate than the NLC, leading to a smoother output voltage and a higher flexibility to select the appropriate submodules and thus, improves voltage balancing.

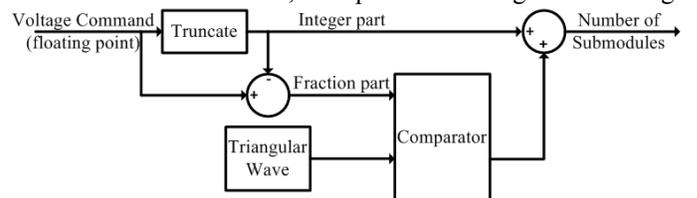


Fig. 4. Modified PD-SPWM algorithm

B. Sorting

As mentioned previously, the submodules have to be ranked based on their voltages. The submodule rank is obtained using sorting algorithms.

Several sorting techniques are available in the technical literature, among those the odd-even sorting technique is adopted in this work [14]. The odd-even sorting, which is a parallelized version of the well-known bubble sorting algorithm, is selected due to its simplicity, suitability for parallel FPGA implementation, and performance.

Similar to the bubble sorting, the main building block of the odd-even sorting is the compare and swap operation. A pair of consecutive values, in the list to be sorted, is compared. If the pair is in the correct order, no action is taken; if not, the elements are swapped. However, for an array of length N , unlike the bubble sorting which requires N^2 iterations to sort the array, the odd-even makes use of parallelization to complete the sorting in only N iterations.

In the odd-even sorting, a pair with indexes x and $x+1$ is considered even if x is an even number, odd otherwise. The odd-even sorting algorithm is separated into two sequential steps. In the first step, the compare and swap operation is performed on all the even pairs. Since the even pairs are all independent, the operation can be conducted in parallel. The second step mirrors the first, but this time comparing the odd pairs. Similarly, since the odd pairs are all independent, the compare and swap operation can be conducted in parallel. This sequence is repeated $N/2$ times to sort the whole array.

The odd-even sorting module is implemented on the FPGA based on utilizing two sets of processing elements. The first set has $N/2$ parallel processing elements responsible for comparing and swapping the even pairs. On the other hand, the second set has $(N/2)-1$ parallel processing elements responsible for comparing and swapping the odd pairs. The processing elements are running with a 100 MHz clock. When the sorting is completed, the output of the sorting module is sent to the selection module.

C. Selection

In order to select the right submodules to connect or bypass, it is not only necessary to know the ranking of all submodules, but it is also required to know, for each arm, which submodules:

- Have the highest voltage and are currently bypassed. These submodules will be the first to be connected if the current causes a voltage decrease.
- Have the lowest voltage and are currently bypassed. These submodules will be the first to be connected if the current causes a voltage increase.
- Have the highest voltage and are currently connected. These modules will be the first to be bypassed if the current causes a voltage increase.
- Have the lowest voltage and are currently connected. These modules will be the first to be bypassed if the current causes a voltage decrease.

The sorted array of submodule voltages is thus fed to this separate function which scans the array searching for these submodules. This only requires scanning once through the array. The selection algorithm finds the top 10 modules of each type. Thus, if more than one module switching is required before the next selection result is made available; the switching algorithm has enough modules to select from.

Finally the selected submodules are fed to a separate function, which compares the number of currently inserted submodules with the switching command. If they are different, the appropriate submodules are connected or bypassed. A functional block diagram of the modulation and capacitor voltage balancing is shown in Fig. 5.

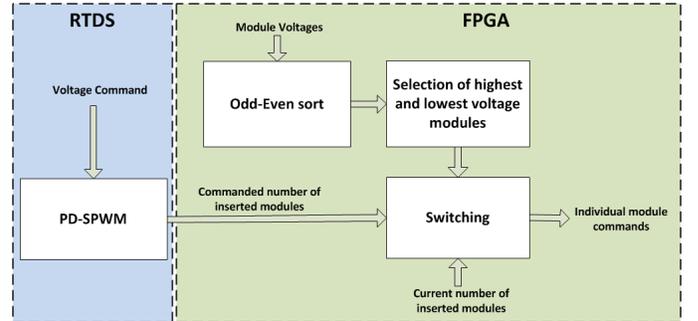


Fig. 5. Functional block diagram of the voltage balancing and module selection implementation

V. CASE STUDY

The main objective of this case study is to demonstrate the capability of the FPGA-based simulator to simulate realistic size MMCs. The system under consideration is based on the MMC-HVDC interconnect between France and Spain [3]. In this work, only one side of the interconnection is modeled and simulated. However, the simulation of full MMC-HVDC interconnects or more complex DC grids is feasible by adding additional FPGA boards. Fig. 6 shows the single line diagram of the system under consideration. The MMC has 400 submodules per arm. The system parameters are given in Table I.

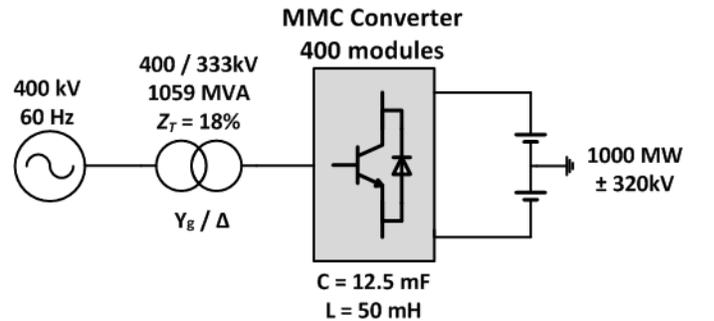


Fig. 6. One line diagram of the simulated system

The MMC was implemented on an Altera Stratix IV EP4SGX530 FPGA chip. The MMC utilizes around 77% of the total logic and 2% of the DSP blocks of the FPGA chip and approximately 1% of the total on chip memory. The rest

of the system is simulated on the RTDS simulator. Fig. 7 shows a photo of the FPGA-based simulator / RTDS setup.

TABLE I
SYSTEM PARAMETERS USED FOR SIMULATIONS

Parameter	Value
AC source voltage	400kV (L-L, rms)
AC Source frequency	60Hz
Transformer windings	Yg/ Δ
Transformer ratio	400 / 333 kV
Transformer rating	1059 MVA
Transformer impedance	18%
Number of modules per arm	400
Module capacitor value	12.5 mF
Capacitor voltage	1.6 kV
Arm inductor	50 mH
DC bus voltage	± 320 kV
DC bus rating	1000 MW

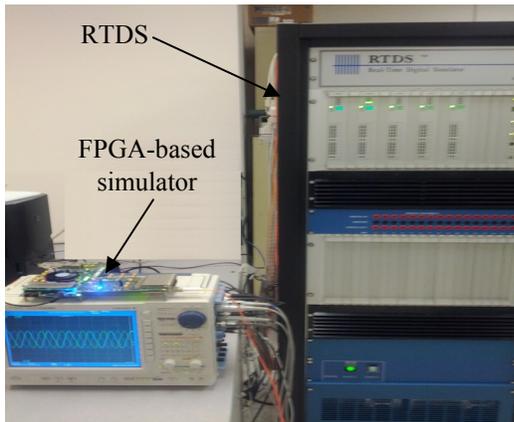


Fig. 7. Photo of the FPGA-based simulator / RTDS setup

A. AC-Side Current Controllers

Based on the abc/dq transformation and using the AC system voltage as the reference for the rotating dq frame, two identical PI controllers are designed. One controller is responsible for controlling the d component of the AC-side current and the other one is responsible for controlling the q component [15]. A basic circulating current controller was also implemented based on the technique presented in [15].

B. Test Scenarios

Several scenarios, that include reversal of the power flow and step changes in the current references, were simulated. The D/A converters of the RTDS were used to send the output waveforms to the scope.

The first scenario verifies the correct operation of the circulating current controller. With the I_d reference set to 0.9 pu and the I_q reference to 0 pu; the circulating current controller is turned on. Fig. 8 and 9 show an oscilloscope screen capture of the circulating currents, AC-side currents, upper and lower arm currents of phase A, and the capacitor voltage of one of the submodules.

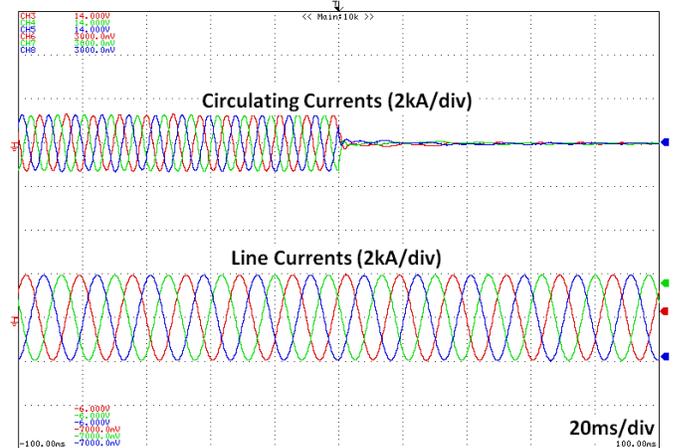


Fig. 8. Scope screen captures for the first scenario. From top to bottom: Circulating currents, and AC line currents.

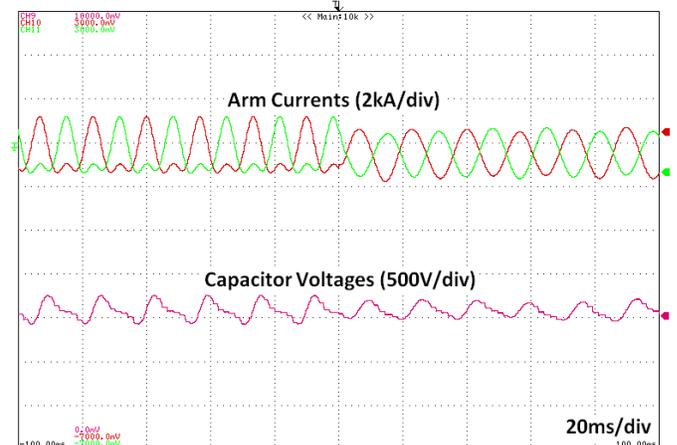


Fig. 9. Scope screen captures for the first scenario. From top to bottom: upper and lower arm currents of phase A of the converter, and capacitor voltage from the upper arm of phase A.

As shown in Fig. 8, upon the activation of the circulating current controller, the circulating currents drop down to almost 0 in few milliseconds. The figure also shows that the operation of the circulating current controllers had no impact on the AC-side currents; as the activation of the controller did not create any noticeable change of the AC line currents. The scope screen capture of the converter arms current, Fig. 9, clearly shows the effect of eliminating the circulating currents. The second harmonic is easily noticed in the current waveforms before the activation of the controller. On the other hand, this second harmonic component is eliminated after the activation of the circulating current controller. The effect of eliminating the circulating currents can also be seen in the submodule capacitor voltage where the voltage fluctuations are reduced after the control is activated.

The second scenario demonstrates the system response to a step change in the I_q current reference (representing a change in reactive power) from 0 to 0.5 pu while the I_d reference remains fixed at -0.9 pu, with the circulating current controller activated.

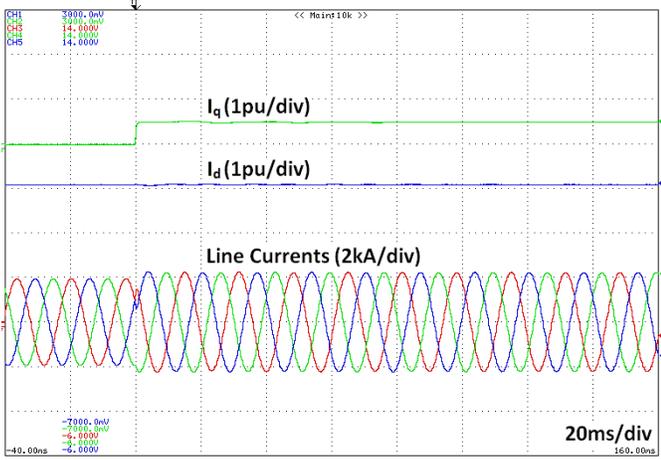


Fig. 10. Scope screen captures for the second scenario, i.e., step change in I_q reference from 0 to 0.5 p.u. From top to bottom: I_q , I_d , and AC line currents.

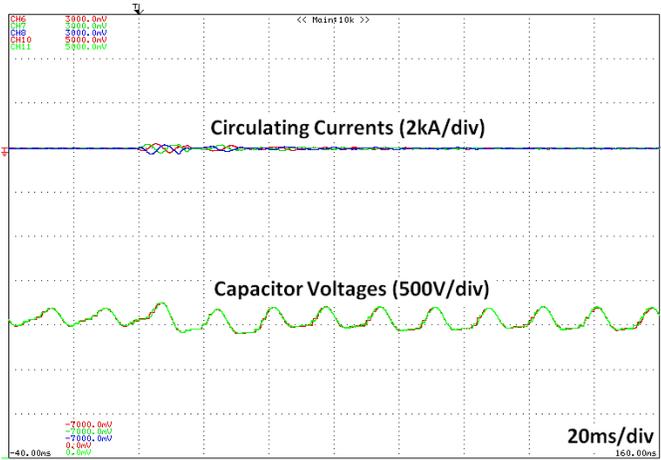


Fig. 11. Scope screen captures for the second scenario, i.e., step change in I_q reference from 0 to 0.5 p.u. From top to bottom: circulating currents, capacitor voltages from the upper arm of phase A.

Fig. 10 and 11 show oscilloscope screen captures of the circulating currents, AC-side currents, upper and lower arm currents of phase A, and the capacitor voltage of one of the submodules. The figures show that the control is stable and has excellent tracking capabilities for step changes in reference currents. In addition, the capacitor voltages remain balanced proving the effectiveness of the voltage balancing approach adopted in this work.

The last test scenario simulates a large reversal in the power flow through the converter, with the reference current command (I_d) changed from -1 pu (inverter) to 1 pu (rectifier) and the reactive reference (I_q) set to 0. For this case as well, the circulating current controller is active. The results are shown in Fig. 12 and 13.

Fig. 12 and 13 show that in this case as well, the control is stable and provides a good response to the step change. Given the significantly large change in currents, circulating currents are noticeable for about 60ms after the change but do settle back to zero. In addition, the capacitor voltages remain

balanced. As can be noticed in Fig. 13, the slopes of the capacitor voltage change after the reference change, reflecting the reversal of the current flow through the converter.

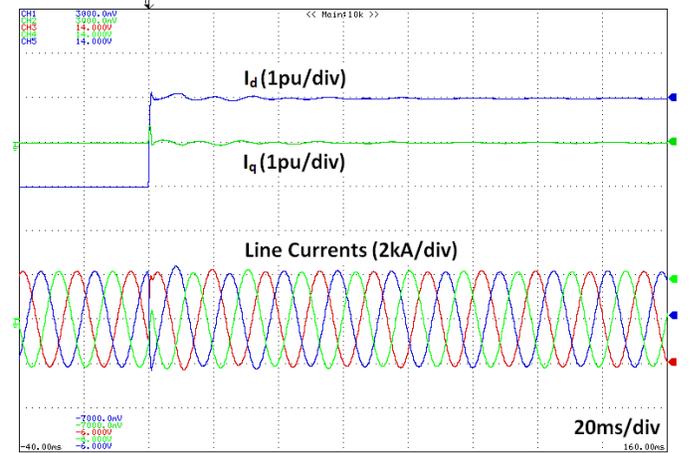


Fig. 12. Scope screen captures for the second scenario, i.e., step change in I_d reference from -1 to 1 p.u. From top to bottom: circulating currents, capacitor voltages from the upper arm of phase A.

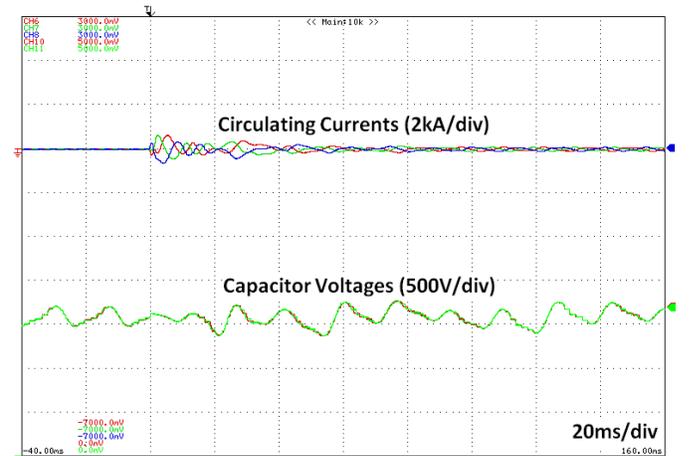


Fig. 13. Scope screen captures for the second scenario, i.e., step change in I_d reference from -1 to 1 p.u. From top to bottom: circulating currents, capacitor voltages from the upper arm of phase A.

VI. CONCLUSIONS

This paper presents a high performance FPGA-based simulator that is used to augment the capabilities of processor-based real-time simulators to enable them to simulate modular multilevel converters that have large number of submodules. The core of the FPGA-based simulator is the massively parallel computational engine that is designed to exploit all possible levels of parallelism.

The performance of the FPGA-based simulator has been verified based on the simulation of a three-phase 400 level MMC system.

VII. REFERENCES

- [1] A. Lesnicar and R. Marquardt, "An Innovative Modular Multilevel Converter Topology Suitable for a Wide Power Range," IEEE Power Tech. Conference, Bologna, June 2003.

- [2] U. Gnanarathna, A. Gole, and R. Jayasinghe, "Efficient Modeling of Modular Multilevel HVDC Converters (MMC) on Electromagnetic Transient Simulation Programs," *IEEE Trans. on Power Delivery*, Vol. 26, No. 1, pp 316-324, Jan 2011.
- [3] J. Peralta, H. Saad, S. Denneriere, J. Mahseredjian, and S. Nguefeu, "Detailed and Averaged Models for a 401-Level MMC-HVDC System," *IEEE Trans. on Power Delivery*, Vol.27, No. 3, pp1501-1508, Jul. 2012.
- [4] X. Jianzhong, Z. Chengyong, L. Wenjing, and G. Chunyi, "Accelerated Model of Modular Multilevel Converters in PSCAD/EMTDC," *IEEE Trans. Power Delivery*, Vol.28, No. 1, pp 129-136, Jan 2013.
- [5] T. Bachir, J. David, C. Dufour, and J. Bélanger, "Effective FPGA-based electric motor modeling with floating-point cores", *IECON 2010*, Glendale, USA, Nov 2010.
- [6] Y. Chen and V. Dinavahi, "FPGA-based Real-time EMTF", *IEEE Trans. on Power Delivery*, Vol. 24, No. 2, pp 892-902, April 2009.
- [7] M. Matar, and R. Iravani, "FPGA Implementation of the Power Electronic Converter Model for Real-Time Simulation of Electromagnetic Transients," *IEEE Tran. on Power Delivery.*, vol. 25, no. 2, pp. 852 – 860, April 2010.
- [8] M. Matar, and R. Iravani, "The Reconfigurable-Hardware Real-Time and Faster-than-Real-Time Simulator for the Analysis of Electromagnetic Transients in Power Systems," *IEEE early access, Trans. on Power Delivery.*
- [9] M. Matar, R. Iravani, "Massively Parallel Implementation of AC Machine Models for FPGA-Based Real-Time Simulation of Electromagnetic Transients", *IEEE Trans. Power Delivery*, Vol. 26, No. 2, pp 830-840, April 2011.
- [10] T. Maguire and J. Giesbrecht, "Small Time-step (<2us) VSC Model for the Real Time Digital Simulator," in *IPST'05*, Montreal, Canada, June 2005.
- [11] M. Saedifard, R. Iravani, "Dynamic Performance of a Modular Multilevel Back-to-Back HVDC System", *IEEE Trans. Power Delivery*, vol. 25, No.4, Oct 2010.
- [12] H. Mohammadi P., M. Tavakoli Bina, "A Transformerless Medium-Voltage STATCOM Topology Based on Extended Modular Multilevel Converters", *IEEE Trans. Power Electronics*, vol. 26, No.5, May 2011.
- [13] L. Xiaoqian, S. Qiang, L. Jianguo, L. Wanhua, "Capacitor voltage balancing control based on CPS-PWM of Module Multilevel Converter" *Energy Conversion Congress and Exposition (ECCE)*, Phoenix, AZ, USA, Sept. 2011.
- [14] M. Lanzagorta, S. Bique, R. Rosenberg, "Introduction to Reconfigurable Supercomputing", Morgan & Claypool, 2010
- [15] Q. Tu, Z. Xu, J. Zhang, "Circulating current suppressing controller in modular multilevel converter". *IECON 2010*, Glendale, AZ, USA, Nov. 2010