Article

A Capacitor Voltage Balancing Approach Based on Mapping Strategy for MMC Applications

Mattia Ricco [1] [ID], Laszlo Mathe [2], Manel Hammami [1],*[ID], Francesco Lo Franco [1], Claudio Rossi [1] and Remus Teodorescu [2][ID]

[1] Department of Electrical, Electronic, and Information Engineering, University of Bologna, 40136 Bologna, Italy; mattia.ricco@unibo.it (M.R.); francesco.lofranco91@gmail.com (F.L.F.); claudio.rossi@unibo.it (C.R.)

[2] Energy Technology Department, Aalborg University, 9100 Aalborg, Denmark; laszlo2mathe@gmail.com (L.M.); ret@et.aau.dk (R.T.)

* Correspondence: manel.hammami2@unibo.it; Tel.: +39-051-20-93589

check for updates

Abstract: This paper proposes a new strategy to achieve balanced capacitor voltages in modular multilevel converters. Among the possible solutions, centralized arm control approaches are often adopted. These methods require a balancing technique based on a sorted list of the sub-modules according to their capacitor voltages. In order to achieve the aforementioned sorted list, different algorithms have been proposed in literature, such as: Sorting algorithms, max/min approaches, etc. However, the sorting algorithms require a long execution time, while the max/min approaches affect the converter dynamic response during faults. To overcome these issues, a new mapping strategy providing a quasi-sorted list is proposed in this paper. The suggested method is compared in simulation with both the classical bubble sorting algorithm, and the max/min method during both normal and faulty conditions. Moreover, the three methods have been implemented in a Xilinx Zynq-7000 System-on-Chip (SoC) device, in order to analyze the corresponding execution time and the required computational effort. Hardware-in-the-loop results are presented for demonstrating the superior performance of the proposed balancing strategy.

Keywords: modular multilevel converters; capacitor voltage balancing; nearest level control; sorting strategy

1. Introduction

Several advantages, such as: Scalability, high modularity, high efficiency, low Total Harmonic Distortion (THD) and high reliability, are the benefits provided by Modular Multilevel Converters (MMCs). For these reasons, this topology is widely adopted in applications such as: High Voltage Direct Current (HVDC) [1], Static Compensators (STATCOM) [2] and in high-power motor drivers [3]. However, it presents several challenges that lead to a more complex control in comparison with the two-level converters. This paper mainly focuses on the balancing of the capacitor voltages among the different Sub-Modules (SMs). Mainly, two categories of Capacitor Voltage Balancing (CVB) control algorithms can be adopted, based on the considered modulation techniques. When the Phase-Shifted Carrier PWM (PSC-PWM) is used, individual balancing control (SM level balancing control) approach can be adopted [4]. On the other hand, when the Nearest Level Control (NLC) or Level Shifted Carrier PWM (LSC-PWM) is implemented, a centralized arm control is usually employed, together with the arm level balancing control algorithm [5–7]. In this paper, a solution for the latter approach is proposed. The arm control consists in selecting the proper SMs to be inserted/bypassed according to the capacitor

voltages and the arm current direction. This approach is usually based on a sorting algorithm that provides a sorted list of the SMs [8,9].

Among the sorting algorithms, the Bubble Sorting Algorithm (BSA) and the even/odd method are very popular. The main advantage is their easy implementation. However, the execution time and the computational efforts rapidly increase when the number of SMs grows; especially when the number of SMs is high (several hundreds). An alternative solution could be the implementation of these kinds of algorithms in a Field Programmable Gate Array (FPGA) in order to enhance their inherent parallelism [10]. Nevertheless, this solution leads to a significant increase of the required resources [11,12]. In order to overcome these limits, max/min approaches have been proposed in the literature [13,14]. These methods are based on the assumption that only one SM has to be inserted or bypassed in each sampling period, then the philosophy is to select the SM with the highest/lowest capacitor voltage, depending on the charging/discharging arm current. However, when faults appear in the system, a fast reaction from the control side is needed, resulting in a request for more SMs to be inserted or bypassed in a single sampling period. The drawback of max/min methods is that more sampling periods are required to insert or bypass the required number of SMs, affecting negatively the converter dynamic response. Other works propose new approaches for reducing either the switching frequency, or the complexity of the CVB algorithm. For example, in [15,16] group-sorting-based balancing approaches are investigated. These methods are based on grouping the SMs, and performing the sorting on the sum of the capacitor voltages of the SMs in the group by reducing the computational load of the sorting technique. Authors in [17], instead, propose a dual sorting mechanism with the same objective. However, these approaches still adopt the sorting algorithm in order to achieve the balancing, and they are mainly focused on the reduction of its computational effort. Then, these approaches could be still used by replacing the sorting algorithm with the proposed strategy. Different techniques have been also proposed for completely avoiding sorting algorithms. The authors in [18] propose a comparison logic for achieving a new modulation scheme, while in [19] a permutation sequence arrangement is adopted. However, in case of HVDC applications, where the number of SMs is large, their complexity increases.

In order to overcome these limits, a Capacitor Voltage Mapping Strategy (CVMS) is discussed in this paper. It avoids classical sorting algorithms by still providing a quasi-sorted list. The aim is to significantly reduce the execution time in comparison with the other sorting techniques. A previous work proposed the strategy [9], while in this paper the demonstration of the effectiveness of the CVMS in both simulations (normal and faulty conditions), and Hardware-In-the-Loop (HIL), is carried out along with the complete design and discussion of the strategy. Moreover, it is compared with both the BSA, which is the common way to implement the sorting technique, and also the max/min approach that is one of the fastest methods. Furthermore, the proposed strategy can handle faulty conditions without affecting the control dynamic, unlike the max/min techniques. These advantages are gained at the expense of a slight increase of the needed memory, which is not a problem in the modern microprocessors, DSPs or FPGA devices.

This paper is divided as follows: Firstly, the MMC topology along with its control hierarchy is presented in Section 2, with particular emphasis on the capacitor voltage balance control. In Section 3, the proposed mapping strategy is described. Section 4 aims to show the simulation results in both normal and faulty conditions. HIL results are achieved by using a Xilinx System-on-Chip (SoC) Zynq-7000 in Section 5. The real execution time for the bubble sorting algorithm, the max/min approach and the proposed strategy is also shown. Finally, the conclusions are drawn.

2. Problem Analysis

The proposed capacitor voltage mapping strategy can be adopted in any kind of MMC; i.e., the SMs can be either half-bridge or full-bridge [20], and as storage element capacitors, but battery cells can also be employed [21,22]. In the following, a three-phase grid-connected MMC, as shown in Figure 1, is considered as an example. The topology and the control are shown in the next paragraphs, and the description of the problem tackled in this paper is given.
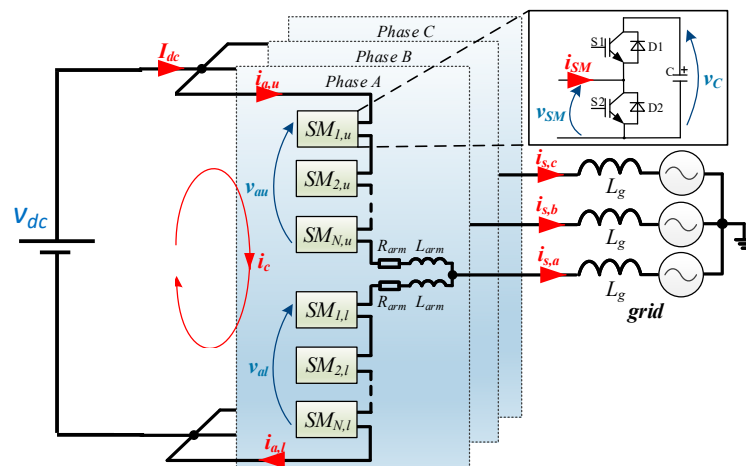
Figure 1. Schematic representation of a three phase grid connected Modular Multilevel Converter (MMC).

2.1. MMC Topology

In the considered topology, each phase consists of an upper and a lower arm, which in turn is composed by a series connection of the SMs, an arm inductor L_{arm}, and an arm parasitic resistances R_{arm} [23]. The half-bridge configuration is considered for the SMs, which consists of two switches (typically insulated-gate bipolar transistors (IGBTs), or metal-oxide-semiconductor field-effect transistor (MOSFETs), with two antiparallel diodes and a capacitor C, as shown in Figure 1. The SM can take two operating states, which results in: Insert or bypass the capacitor in the arm circuit. The notation used in this paper, and the main equations of the converter, are introduced in [24], and shown in Figure 1.

2.2. MMC Control Levels Hierarchy

The overall block diagram of the typical MMC control scheme is depicted in Figure 2 [24]. The current references $i_{s,a}{}^*$, $i_{s,b}{}^*$ and $i_{s,c}{}^*$ are evaluated from the higher level control, after having measured the grid voltage v_{grid}, the DC voltage V_{dc}, and the three phase currents $i_{s,a}$, $i_{s,b}$ and $i_{s,c}$. These references are sent to the output current control block that provides the proper output reference voltage $v_{o_{p,m}}{}^*$ where p and m specify the phase and the arm (u = upper and l = lower), respectively. Below, in the hierarchy level, there is the arm balancing control. It is in charge of controlling the circulating current i_c and the energy between the arms [25]. The NLC is chosen as a modulation technique and placed at this stage. In each sampling period, it provides the insertion indices (which vary between 0–) $n_{p,u}$ and $n_{p,l}$ for the upper and lower arm, respectively:

$$n_{p,m} = round\left(\frac{v_{o_{p,m}}{}^*}{V_{dc}}\right) \tag{1}$$
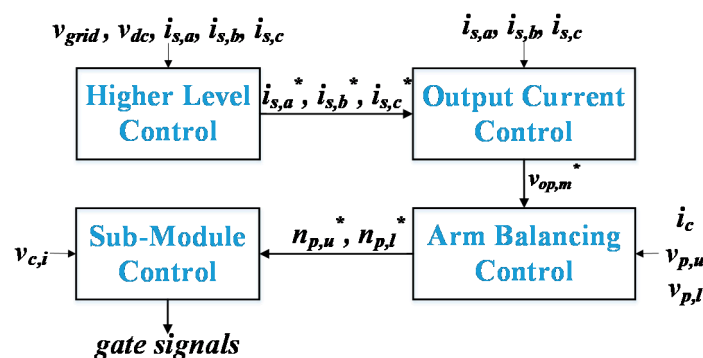

Figure 2. MMC control block diagram.

From the NLC point of view it is indifferent which particular SMs are inserted/bypassed, but it is important that the overall number of the inserted/bypassed SMs fulfills (1). This degree of freedom is used by the SM control for achieving the balance of the capacitor voltages and then generating the gate signals for the switches. This paper is mainly focused on the latter control, and it is deeply described in the following.

2.3. Capacitor Voltage Balancing Control

As already mentioned before, each capacitor from the SM can be inserted in the arm circuit, contributing to the output voltage, or it can be bypassed. In case the SM is in this bypassed state, its voltage does not change. On the other hand, if it is inserted, it can be charged or discharged according to the current direction. This leads to a variation on the capacitor voltage that it is superimposed to its inherent ripple. The CVB algorithm is in charge of limiting this variation by balancing the voltages. It mainly consists of two modules: The sorting algorithm and the selection method. Figure 3 shows the flowchart of the capacitor voltage balancing algorithm in the NLC. If $N_{p,m}[k-1] = N_{p,m}[k]$, no switching is required. Otherwise, the number of inserted SMs is evaluated, and the best suitable SMs for achieving the balancing are normally selected. However, this causes a high switching frequency. For this reason, an optimized switching frequency selection method can be adopted. It evaluates $\Delta N_{p,m}[k]$ as the difference between $N_{p,m}[k]$ and $N_{p,m}[k-1]$. This leads to a reduction of the switching frequency at the expense of a higher imbalance among the capacitor voltages. After having evaluated $\Delta N_{p,m}[k]$, the sorting algorithm is launched. Its aim is to provide a sorted list of the sub-modules according to their capacitor voltages. Among the different sorting algorithms, the bubble sorting algorithm is often used, due to its excellent scalability and faster response when the input data is quasi-sorted. Finally, four cases can be distinguished:

- $\Delta N_{p,m}[k] > 0$ and $i_{p,m} > 0$: The bypassed SMs with the lowest capacitor voltage should be inserted;
- $\Delta N_{p,m}[k] > 0$ and $i_{p,m} < 0$: The bypassed SMs with the highest capacitor voltage should be inserted;
- $\Delta N_{p,m}[k] < 0$ and $i_{p,m} > 0$: The inserted SMs with the highest capacitor voltage should be bypassed;
- $\Delta N_{p,m}[k] < 0$ and $i_{p,m} < 0$: The inserted SMs with the lowest capacitor voltage should be bypassed;

However, the sorting algorithm complexity and execution time both increase, along with the number of SMs. Then it becomes less attractive when the number of SMs in the MMC increases. This could either limit the maximum number of SMs in the arm, or reduce the levels of the converter output voltage. In order to overcome these limits, others' approaches have been proposed in literature [15,17–19].

Among all the possibilities, the max/min approach has been considered for comparison due to its very fast response and easy implementation. The aim is to show that it is possible to accomplish better timing performance by still achieving a sorted list. Usually in steady state operation, during one sampling period, only one SM is manipulated (inserted or bypassed), and thus there is no need to resort the whole list. Then it is possible to adopt the max/min function in order to find only the SM with either lowest or highest capacitor voltages [13]. However, during faults, when more SMs have to be inserted or bypassed, this technique gives some limitations. The control dynamic of the converter slows down, as it will be shown later through simulations.
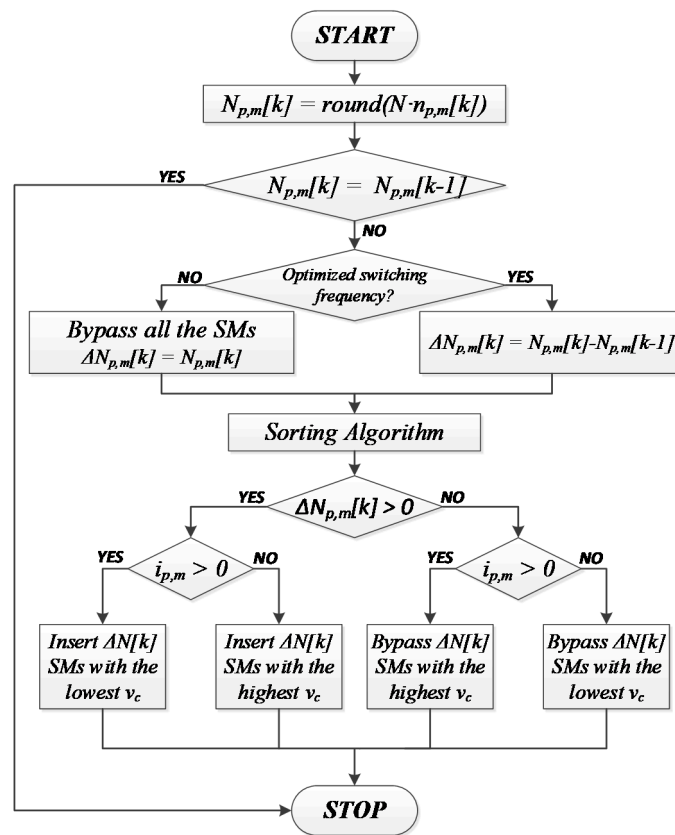
Figure 3. Flowchart of the basic capacitor voltage balancing algorithm in Nearest Level Control (NLC).

3. Description of the Proposed Capacitor Voltage Mapping Strategy

As highlighted before, in order to achieve a balance between the capacitor voltages, a ranking is firstly needed. In this section the CVMS is deeply discussed. In comparison with [9] the choice of the voltage range is discussed, and the optimal values are found. Moreover, some improvements and a complexity evaluation are carried out.

3.1. Functional Principle

Normally the sorting methods, already proposed in literature, work with these fundamental steps:

1. Acquire the capacitor voltages;
2. Store the capacitor voltages and the SM positions in a memory block;
3. Perform the sorting.

On the contrary, the proposed CVMS acts between the first and the second step. The idea is to directly store the SM positions in the right order, avoiding further manipulations. The strategy consists in dividing the capacitor voltage operating range in different sub-ranges, and assigning each sub-range to a memory location.

Then, only the SM positions (named here P_i) are stored in the memory according to their corresponding sub-ranges. In this way a quasi-sorted list is achieved right after reading the analogue-to-digital conversion of the capacitor voltage.

In normal operation, the SM capacitor voltages are between a minimum and maximum value, $V_{c_{min}}$ and $V_{c_{max}}$, respectively. These two values can be evaluated when the MMC design is performed. The operation range of the capacitor voltage is divided into M sub-ranges with amplitudes equal to:

$$\Delta V = \frac{V_{c_{max}} - V_{c_{min}}}{M} \tag{2}$$

Each sub-range is mapped in a specific FIFO memory array as shown in Figure 4. Therefore, all the SMs with the capacitor voltage that falls in the sub-range 0 are mapped in the FIFO memory at address 0; all the SMs with the capacitor voltage in the sub-range 1 in the address 1; and so on. Then, the number of sub-ranges can be seen as an address ADDR of the FIFO memories.
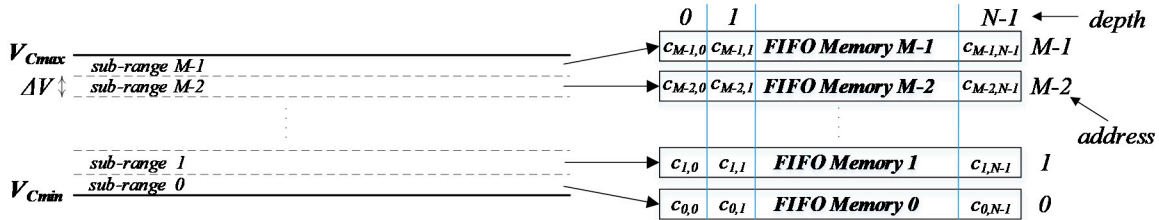

Figure 4. The operating voltage range of each capacitor is divided in sub-ranges with a resolution ΔV. Each sub-range is mapped in a FIFO memory, where only the Sub-Module (SM) positions are stored.

Firstly, a position number P_i is assigned for each SM based on its physical placement in the arm. Then, it is stored along with the SM status (inserted or bypassed) in the cell $c_{ADDRi,k}$, according to the address $ADDR_i$, and the status of the FIFO memory k, that takes into account how many positions are already stored in the FIFO memory i. The number of sub-ranges M defines how many FIFO memories are required. Their depth has been chosen equal to the number of SMs N in order to allow for storing all the positions in one FIFO memory when the capacitor voltages are close to each other. However, this FIFO depth can be optimized as described in [9]. The bit-length of each memory cell has to be larger or equal to the size of the SM position P_i plus the status bit.

The SMs are normally inserted or bypassed accordingly to the NLC, i.e., when the insertion indices $n_{p,u}$ and $n_{p,l}$, given in Equation (1) change. However, in order to avoid exceeding the capacitor voltage band during faults, some precautions should be considered. In addition to the normal insertion or bypass of the SMs according to the NLC, a swap is performed if an SM capacitor voltage reaches the last or the first voltage sub-range. The swap involves bypassing the SM that has reached the first or the last memory location, and inserting another SM accordingly to the mapping strategy.

3.1.1. Address Evaluation

In this paragraph the way to perform an efficient mapping of the SMs into the memory is presented. The memory address $ADDR_i$ is obtained by scaling the capacitor voltage $v_{c,i}$. The minimum capacitor voltage $V_{c,min}$ (which is an MMC design criteria) is subtracted from the measured voltage $v_{c,i}$. The result is firstly divided by the amplitude of the sub-ranges ΔV and then rounded in order to obtain an address, as shown in Figure 5. P_i is then stored in the memory according to the evaluated address.
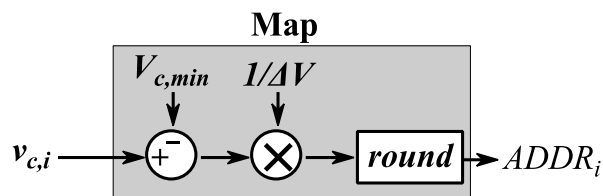

Figure 5. Evaluation of the memory address ADDRi. $V_{c,min}$ is subtracted to the measured capacitor voltage $v_{c,i}$ and the result is divided by the amplitude of the sub-ranges ΔV.

3.1.2. Reading Operation

The SMs to be inserted or bypassed are obtained by reading the FIFO memories from the top (bottom) if the SMs with the highest (lowest) capacitor voltages are needed, according to the flowchart in Figure 3. When a FIFO memory is empty, the memory located at the next address must be read.

3.1.3. V_{cmin} and V_{cmax} Evaluation

V_{cmin} and V_{cmax} can be chosen, either equal to the maximum voltage of the capacitors, or by evaluating the maximum capacitor voltage ripple. In the first case, the proper behavior is ensured for each condition; however, by considering the same amplitude of the sub-ranges, a higher amount of memory is required. On the other hand, the second case can reduce the hardware resources, but a more detailed study should be accomplished. Moreover, once having evaluated the maximum ripple, a safety tolerance must be added to ensure good behavior during fault conditions. From [24], the energy in the upper and lower arm can be evaluated:

$$W_u = \frac{W_\Sigma + W_\Delta}{2} \tag{3}$$

$$W_l = \frac{W_\Sigma - W_\Delta}{2} \tag{4}$$

where W_Σ and W_Δ are the per-phase energy and the imbalance energy, respectively. They are equal to:

$$W_\Sigma = W_{\Sigma 0} + \Delta W_\Sigma \tag{5}$$

$$W_\Delta = W_{\Delta 0} + \Delta W_\Delta \tag{6}$$

with:

$$W_{\Sigma 0} = \frac{C \cdot v_d^2}{N} \tag{7}$$

$$W_{\Delta 0} = 0 \tag{8}$$

$$\Delta W_\Sigma = -\frac{\hat{V}_s \cdot \hat{I}_s}{8 \cdot \omega_1} \sin(2\omega_1 t - \varphi) \tag{9}$$

$$\Delta W_\Delta = -\frac{v_d \cdot \hat{I}_s}{4 \cdot \omega_1} \sin(\omega_1 t - \varphi) + \frac{2 \cdot \hat{V}_s \cdot i_c}{2 \cdot \omega_1} \sin(\omega_1 t) \tag{10}$$

Being v_d the pole-to-pole dc bus, ω_1 the fundamental angular frequency, φ the phase angle, $\hat{V}_s$ and $\hat{I}_s$ the peak value of the fundamental output voltage and current, respectively.

Then, by replacing Equations (3) to (8) into W_u and W_l presented in Equations (11) and (12), the sum of the capacitor voltages for the upper arm and for the lower arm can be determined:

$$v^\Sigma_{c,u} = \sqrt{\frac{2 \cdot N}{C} W_u} \approx v_d + \frac{N}{2 \cdot C \cdot v_d}(\Delta W_\Sigma + \Delta W_\Delta) \tag{11}$$

$$v^\Sigma_{c,l} = \sqrt{\frac{2 \cdot N}{C} W_l} \approx v_d + \frac{N}{2 \cdot C \cdot v_d}(\Delta W_\Sigma - \Delta W_\Delta) \tag{12}$$

Finally, the capacitor voltage ripple can be expressed as:

$$\Delta v_{c,u} \approx \frac{1}{2 \cdot C \cdot v_d}(\Delta W_\Sigma + \Delta W_\Delta) \tag{13}$$

$$\Delta v_{c,l} \approx \frac{1}{2 \cdot C \cdot v_d}(\Delta W_\Sigma - \Delta W_\Delta) \tag{14}$$

Based on the evaluated ripple, the V_{max} and V_{min} can be chosen. In the following, a safety tolerance of 20% is also considered. It is worth noting that V_{max} and V_{min} are pre-evaluated and considered constant in this paper.

3.1.4. Accuracy and Complexity Evaluation

The achieved capacitor voltage balancing depends on the amplitude of the sub-ranges ΔV. It is then also correlated to the available memory. As already said, from the CVB point of view, the small deviations among the capacitor voltages are not of interest. It is rather crucial not to exceed the safety limit of the capacitor voltage. This concept is already adopted in literature in order to optimize the switching frequency in the NLC [8]. It allows to the capacitor voltages to freely vary inside a given band, performing extra switching only when the band is going to be exceeded. Therefore, a proper compromise can be guaranteed between the right CVB behavior and the memory usage. An M comprising between 8 and 64 can achieve this objective.

Concerning the complexity, the bubble sorting algorithm has the highest complexity equal to $O(N^2)$, where O defines an upper bound of the running time of the algorithm. On the other hand, the max/min approach has a complexity equal to $O(N)$. Regarding the CVMS, it is equal to $O(N)$ only when $M \ll N$. Indeed, if M is comparable with N, the time for reading the memory becomes comparable with the time for writing in the memory. This is due to the fact that it is easier to find empty FIFO memories, and it requires more often an update of the address. These results are summarized in Table 1.

Table 1. Complexity Evaluation.

Sorting Method	Complexity
Bubble Sorting Algorithm	$O(N^2)$
Max/Min Approach	$O(N)$
Proposed strategy (M $\ll$ N)	$O(N)$

4. Simulation Results

In this section the proposed CVMS is validated, and its advantages are highlighted compared to the popular bubble sorting algorithm and the fast max/min technique. In order to demonstrate the benefits of the proposed method, the MMC has been simulated in both normal and faulty conditions. The simulations have been performed in a PLECS® power electronic simulation environment. The MMC and grid parameters are given in Tables 2 and 3, respectively.

Table 2. MMC Parameters.

Quantity	Value
DC-link Voltage (V_{dc})	200 kV
SM Capacitor (C)	600 μF
Arm Inductance (L_{arm})	50 mH
Arm Resistance (R_{arm})	1.6 Ω
Number of SM (N)	16
Sampling frequency (f_s)	10 kHz

Table 3. Grid Parameters.

Quantity	Value
Grid frequency (f_{grid})	50 Hz
Grid Voltage (V_{grid})	121.2 kV
Grid Inductance (L_{grid})	3 mH
Grid Resistance (R_{grid})	0.1 Ω

All the three methods: Bubble sorting, max/min and the proposed sorting approach, have been implemented in PLECS®. Concerning the max/min approach, the one proposed in [13] has been implemented in this paper. Its tolerance band has been set to 2 kV. Concerning the proposed mapping

strategy, the adopted parameters are summarized in Table 4. At the beginning the system is in the steady state condition.

Table 4. Capacitor Voltage Mapping Strategy (CVMS) parameters.

Quantity	Value
Number of FIFO memories (M)	8-16-64
Nominal capacitor voltage ($V_{c,nom}$)	V_{dc}/N
Maximum capacitor voltage ($V_{c,max}$)	15 kV
Minimum capacitor voltage ($V_{c,min}$)	10 kV

The phase-to-ground fault is simulated at 0.7 s and removed at 0.8 s. When a fault occurs, the control requires to suddenly insert or bypass more SMs in the next sampling period. In this case, an algorithm that provides a fully sorted list (such as BSA) can enhance the dynamic performance of the converter. Indeed, it can allow the insertion (or the bypass) of the required SMs in only one sampling period. On the contrary, a max/min approach can insert only 1 SM, and then a number of sampling periods equal to the required SMs have to be intervened before the required control action is achieved. It might be executed in a repetitive manner during the same sampling period, but in this way the BSA is resulted.

4.1. Impact on Voltage Balance and Switching Frequency

Firstly, a comparison in terms of capacitor voltage balance and achieved switching frequency is made among the implemented methods before the fault occurs. Figure 6 shows the capacitor voltages obtained with: Bubble sorting algorithm, max/min approach, and capacitor voltage mapping strategy in the case of M = 8, 16, 64, with and without the optimization of the switching frequency. The bubble sorting algorithm is executed each time there is a change in the insertion index $n_{p,m}$, and the selection technique always selects the best suitable SM which leads to the best voltage balancing at the cost of a high switching frequency.

On the other hand, the max/min approach intrinsically optimizes the switching frequency; indeed, it just selects 1 SM to be inserted (or bypassed) each time there is a change in $n_{p,m}$. The proposed technique is executed with and without the switching frequency optimization (refers to Figure 3). From Figure 6c to Figure 6e the switching frequency optimization is off, and then the best SMs are inserted every time.

On the contrary, the last three plots of Figure 6 consider an optimized selection method that insert or bypass only the required SMs at the same time. The proposed strategy without the optimized selection technique tends to the balancing achieved with the BSA when the number of sub-ranges M increases. However, in this last case, the resulted average switching frequency ($f_{sw,av}$) is unacceptably high. Then, the switching frequency optimization can be taken into consideration. In this case, the achieved voltage balancing is degraded, but the capacitor voltages are kept between the acceptable boundaries (straight black lines in Figure 6). Moreover, by increasing M it is possible to improve the switching frequency at the cost of a small increase in the required memory of the controller.

Finally, it can be concluded that the proposed strategy well balances the capacitor voltages in normal conditions like the other two methods. Moreover, the achieved switching frequency is close to the one obtained with the max/min approach, but a complete sorted list is performed. The number of sub-ranges M can be chosen according to the application under study. In the following, the optimized switching frequency method and M = 8 are considered.
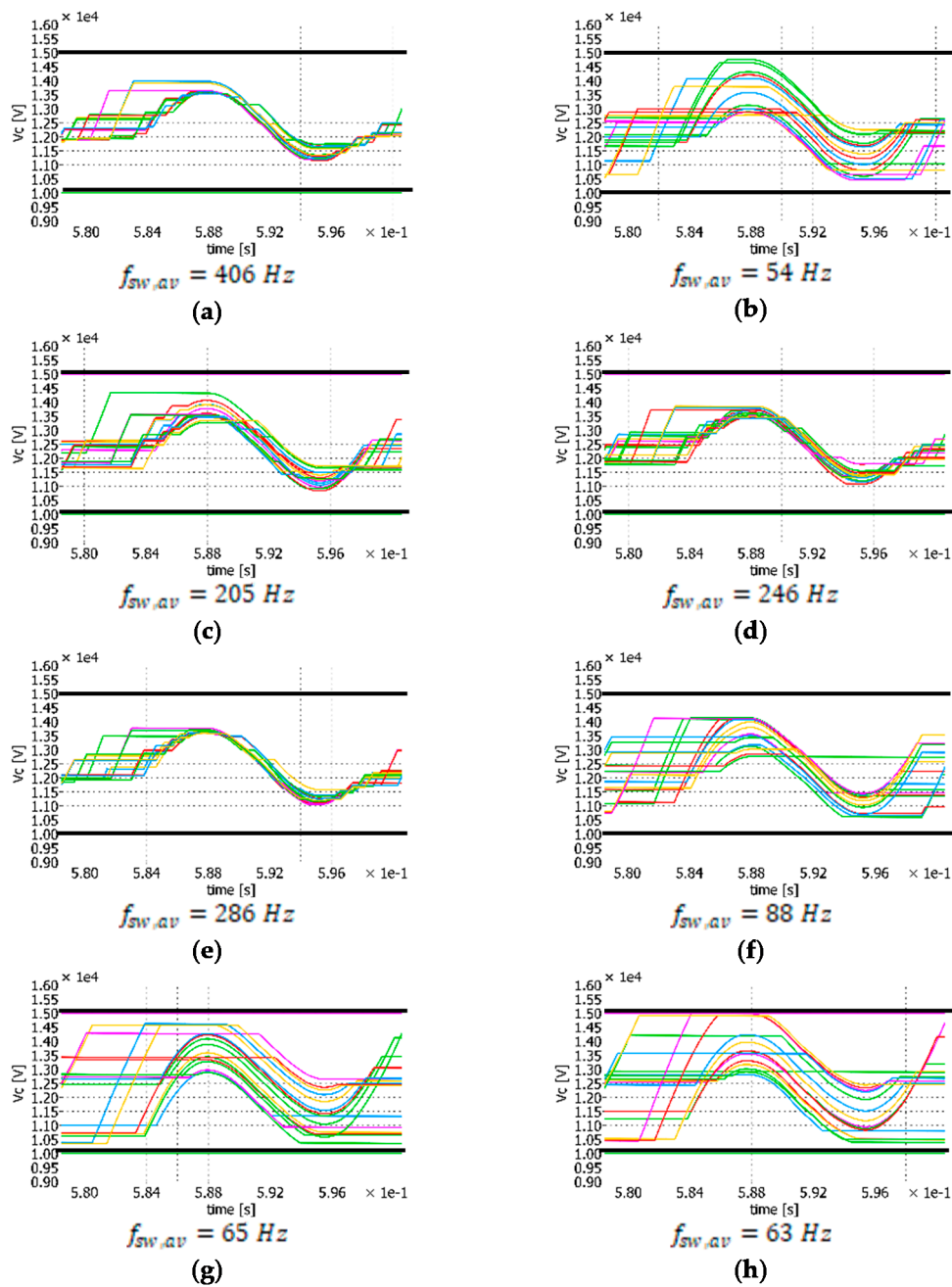
Figure 6. Achieved capacitor voltage waveforms during simulations applying different sorting methods: (**a**) Bubble Sorting Algorithm; (**b**) max/min approach; (**c**) Capacitor Voltage Mapping Strategy (CVMS) with M = 8 without switching frequency optimization (wosfo); (**d**) CVMS with M = 16 wosfo; (**e**) CVMS with M = 64 wosfo; (**f**) CVMS with M = 8 with switching frequency optimization (wsfo); (**g**) CVMS with M = 16 wsfo; (**h**) CVMS with M = 64 wsfo.

4.2. Performance during Phase-to-Ground Fault

Figure 7 shows the output currents, the capacitor voltages and the required number of switches for the upper arm of phase *a* in the case of the bubble sorting algorithm, the max/min approach and the proposed mapping strategy before, during and after the phase-to-ground fault. The fault occurs at 0.7 s, and it is removed at 0.8 s. At the moment of the fault, 3 SMs have to be bypassed for this example.
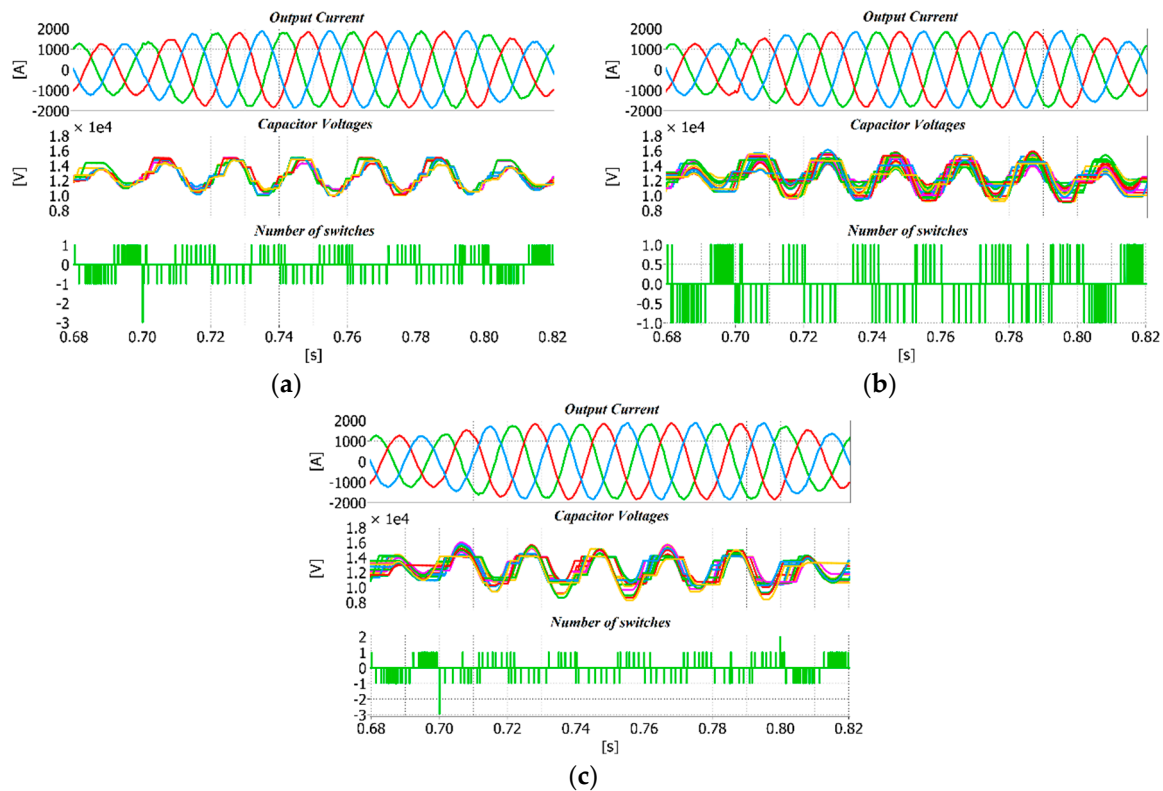
Figure 7. Output currents, capacitor voltages and number of switches per each T_s in simulation during phase to ground fault: (**a**) Bubble Sorting Algorithm; (**b**) max/min approach and (**c**) proposed mapping strategy M = 8 with the switching frequency optimization.

Figure 8 shows a zoom of the output current at the time of the fault. It is worth noting that the output current achieved with the max/min approach has an overshoot caused by the slower dynamic response of the controller.
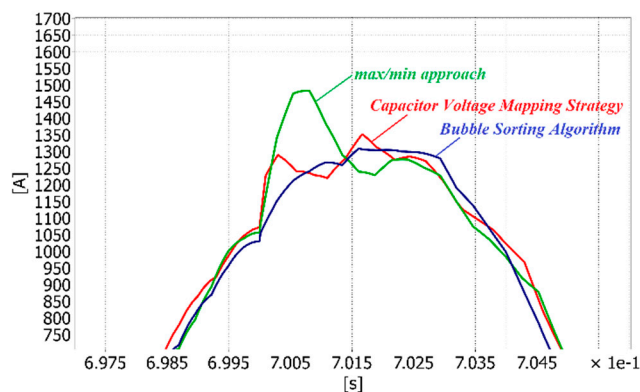

Figure 8. Zoom of the output currents of the upper arm in phase *a* and during phase to ground fault. Blue line: Bubble sorting algorithm. Green line: Max/min approach. Red line: Proposed mapping strategy.

4.3. Performance during Line-to-Line Fault

A line-to-line fault between phase a and phase b has been also simulated in a PLECS® environment. The fault occurs at 0.9 s and is removed at 1.0 s. The output currents, capacitor voltages and the number of required switches are shown in Figure 9 for both max/min approach, Figure 9a, and the proposed capacitor voltage mapping strategy, Figure 9b.
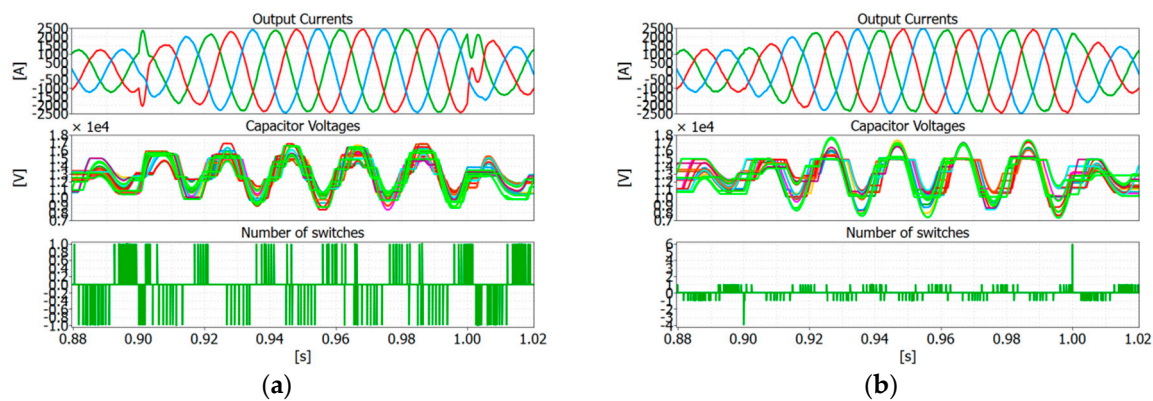
Figure 9. Output currents, capacitor voltages and number of switches per each Ts in simulation during phase to phase fault: (**a**) Max/min approach; (**b**) proposed mapping strategy M = 8 with the switching frequency optimization.

Like in the case of phase to ground fault, the max/min approach slows down the dynamic response of the controller by causing an overshoot of the output current, as it can be seen in Figure 9a. On the other hand, the proposed technique is able to bypass 4 SMs at the same time when the fault occurs by then providing a faster dynamic response and avoiding overshoots on the output currents.

The good behavior of the proposed mapping strategy has been demonstrated with a different number of sub-ranges M. Moreover, the phase-to-ground fault and the phase-to-phase fault have been considered in order to show the limits of the max/min approach. The latter allows a very low execution time, and an easy implementation at the expense of a slower response of the controller during fault conditions. The proposed technique, instead, does not affect the control dynamic, still keeping the execution time low, as shown in the next section.

5. HIL Results

Due to the fact that the realization of large MMC (consisting of hundreds of modules) systems are very expensive, a common approach to test the behavior of the control system is the use of the HIL approach. In this section HIL results have been provided in order to evaluate the implementation requirements, show the proper behavior of the proposed strategy. and compare it with the bubble sorting algorithm and the max/min approach in terms of the achieved execution time. For the HIL system the Digilent Zedboard mounting a Xilinx SoC Zynq-7000 device (named in the following simply Zynq) has been adopted. The adopted Zynq consists of two embedded ARM Cortex processors, 85,000 logic cells, 4.9 Mb block RAM and 220 DSP slices. It is as shown in Figure 10. The MMC control is implemented in the first processor, along with the three sorting methods. The emulated plant is instantiated in the second processor. The proposed CVMS technique is also implemented in the programmable logic in order to enhance its inherent parallelism, and then reduce the execution time. The processor system and the programmable logic communicate through the AXI bus. A single-phase MMC model, with $N = 64$ and based on [26], is implemented in the second processor.

The case of $N = 64$ is realistic in view of an industrial application. The SM capacitors are equal to 2.4 mF and the output energy of 500 kW is dissipated on a resistive load. The remaining MMC parameters are equal to the ones presented in Table 2.

It is worth it to note here that the implementation of a three-phase system requires a huge amount of hardware resources for the real-time emulation of the plant, without adding anything to the performance analysis intended into this paper. Indeed, the considered single-phase controller can be seen as the phase controller of a three-phase system. In order to enhance the timing performance of the proposed CVMS, it has been also implemented in the Programmable Logic (PL) part of the SoC. The hardware architecture is depicted in Figure 11.
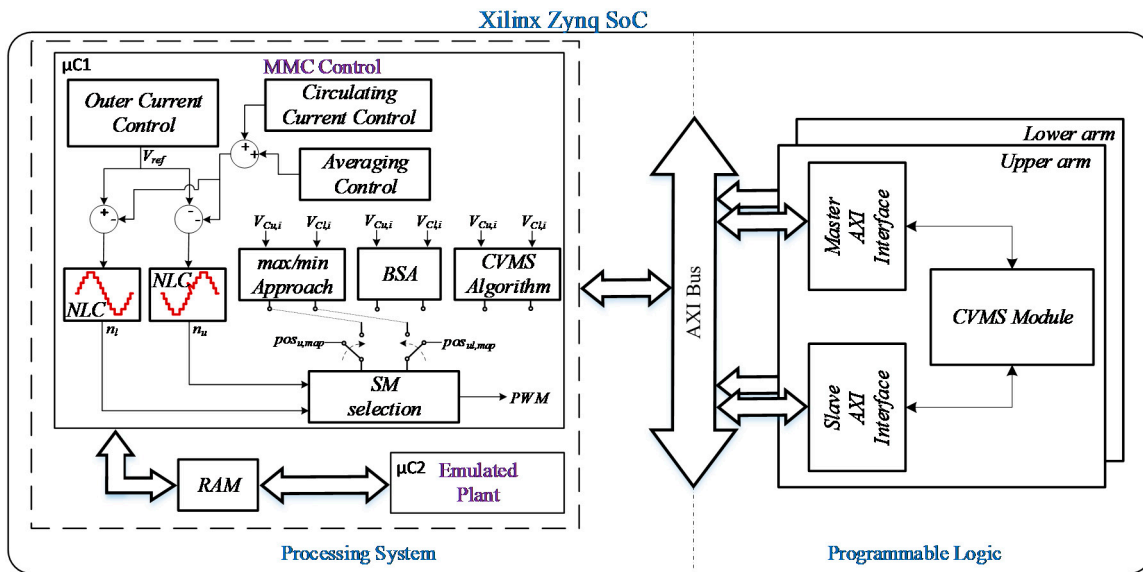
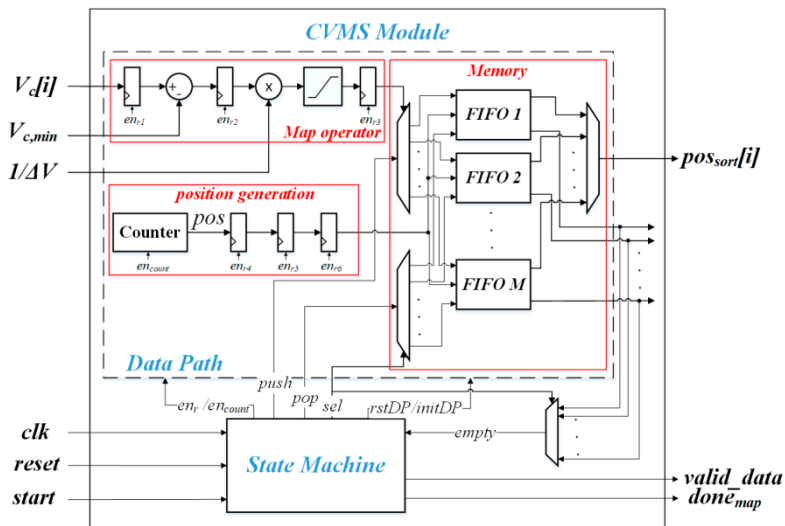**Figure 10.** Hardware architecture of the developed system.

Figure 11. Hardware architecture of the proposed mapping strategy composed by the data path and the state machine. The data path consists of the map operator, the memory block and the position generator.

It consists of a data path for treating the data, and a state machine for the synchronization. In the data path, the map operator, the position generator and the allocated memory, can be easily noted. The quasi-sorted list is sent to the controller through the AXI bus. The chosen AXI mode is the burst configuration, in order to decrease the latency due to the communication.

It is worth noting that the main controller can start its action as soon as it receives the first data, because the sent list is already sorted. Considering the BSA and the max/min approach, the controller should wait all the voltages in order to start the algorithms in the PS.

Figure 12 shows the flow chart of the state machine that controls the data path in Figure 11. When the capacitor voltages are measured, the operation for writing the SM positions in the memory firstly starts. Then, the position pos is set to 0, and the map operator evaluates the address add corresponding to the first capacitor voltage. The position pos is stored in the FIFO memory add, and the value of pos is then increased. If it is still less than the number of SMs, the map operator is executed again. If pos is equal to N it means that all the capacitor voltages have been measured, and the SM positions have all been stored in the memory, and the writing operation is ended. At this point the

reading operation starts, and the order direction is firstly checked. If a descending list is needed, the address add is set to M, and the FIFO memory add is read until it is empty. When this happens, the value of add is decreased and then the next FIFO memory can be read. This operation lasts until the value of add is equal to 0. On the other hand, if an ascending list is required, add is set to 0, and each time the FIFO memory add is empty, its value is increased. In this case the operation ends when add is equal to *M*.
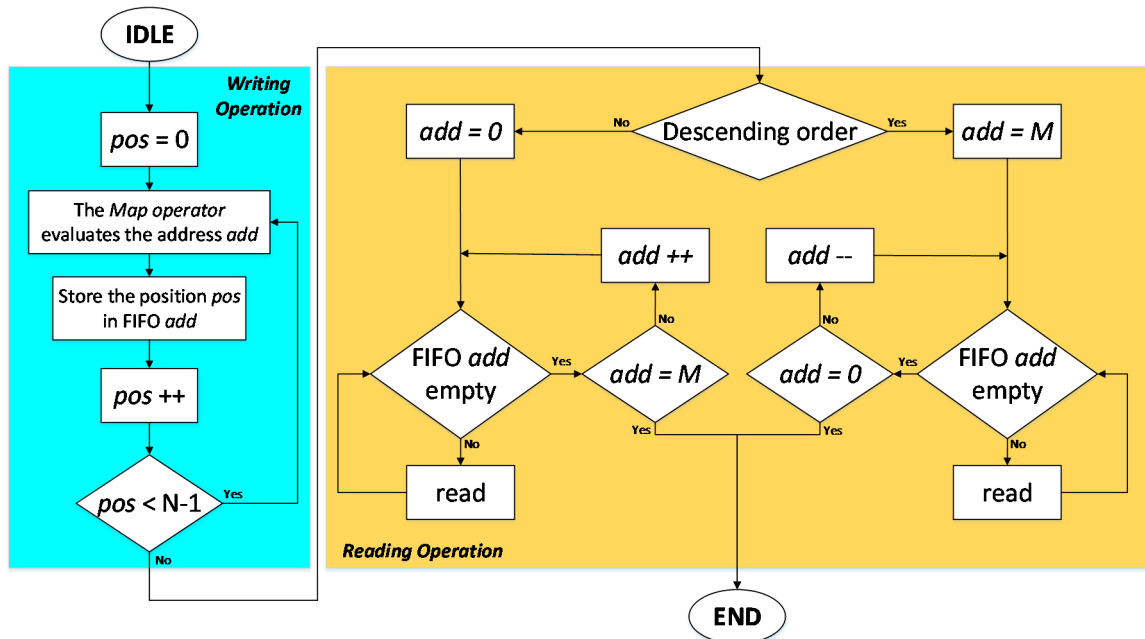

Figure 12. Flow chart that describes the state machine in Figure 10.

Figure 13 shows the capacitor voltages and the execution time for the bubble sorting algorithm, the max/min approach, the CVMS implemented in the processor system and in the programmable logic. These results have been achieved by changing in real-time the sorting technique by using the push buttons provided on the Zynq development board.
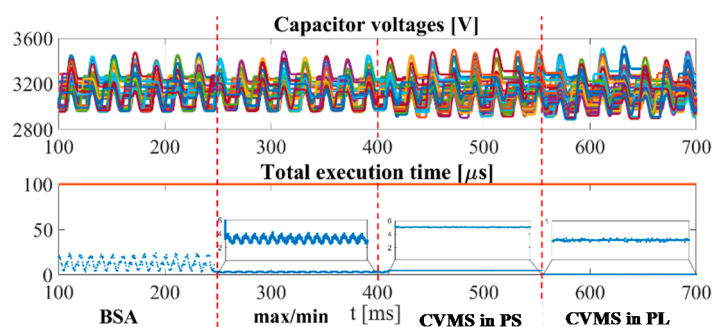

Figure 13. Capacitor voltages of the upper arm and execution time for the bubble sorting algorithm, the max/min approach, the CVMS implemented in the processor system and in the programmable logic (Hardware-In-the-Loop (HIL) results).

A serial communication protocol has been employed to read the internal signals and display them on the PC. The achieved execution times are summarized in Table 5. It is noteworthy that the execution time of the CVMS in PS is comparable with the one obtained with the max/min approach, but it provides a quasi-sorted list useful for enhancing the controller performance, as shown in the previous section. Besides, if the PL solution is adopted, thanks to the exploitation of the inherent

algorithm parallelism, its execution time can be reduced, further achieving the best timing performance, and still keeping the controller dynamic. The required resources for the proposed mapping strategy implemented in the PL part are summarized in Table 6. It is worth it to note that the required memory is less than the 3% of the available memory on the Zynq platform. Then, it is possible to conclude that a faster capacitor voltage balancing algorithm is achieved by preserving the controller dynamic during fault, and that without significantly increasing the required resources.

Table 5. Execution time when $N = 64$.

Sorting Method	Maximum Execution Time T_{ex}
Bubble Sorting algorithm	24.0 µs
Max/min approach	4.3 µs
CVMS in PS	5.0 µs
CVMS in PL	0.7 µs

Table 6. Required Resources for the proposed mapping strategy implemented in the Programmable Logic (PL).

Resource	Required (Available)	Percentage (%)
Look Up Table (LUT)	518 (53,200)	0.97
Flip-Flop (FF)	589 (106,400)	0.55
BRAM	4 (140)	2.86
DSP	1 (220)	0.45

6. Conclusions

This paper presented a capacitor voltage balancing strategy that provides a quasi-sorted list of the SMs according to their capacitor voltages. It acts just after the analogue-to-digital conversion of the capacitor voltages. The idea behind the method is based on dividing the capacitor voltage operating range into different sub-ranges, and assigning each sub-range to a memory location. Then, the SM positions are stored in the memory according to their corresponding sub-ranges by directly leading a quasi-sorted list. This allows a strong reduction of the execution time in comparison with the other sorting algorithms. Moreover, unlike max/min approaches, the proposed method still provides a sorted list by avoiding a slower dynamic performance. These benefits come at the expense of a slight increase in memory requirements. The proposed technique has been compared with the bubble sorting algorithm and a max/min approach in both normal and fault conditions by demonstrating its effectiveness. HIL results prove that the execution time of the mapping strategy is very close to the one achieved with the max/min approach, and five times lower than the one obtained with the bubble sorting algorithm. It has been also shown that by adopting a full hardware implementation, the timing performance significantly increases by also leading a solution that is five times faster than the max/min method.

Author Contributions: Conceptualization, M.R. and L.M.; Formal analysis, M.R.; Methodology, M.R., M.H. and R.T.; Validation, M.R.; Visualization, F.L.F.; Writing – original draft, M.R. and L.M.; Writing – review & editing, M.H., F.L.F., C.R. and R.T.

References

1. Lyu, J.; Cai, X.; Molinas, M. Frequency Domain Stability Analysis of MMC-Based HVDC for Wind Farm Integration. *IEEE J. Emerg. Sel. Top. Power Electron.* **2016**, *4*, 141–151. [CrossRef]
2. Bhesaniya, M.M.; Shukla, A. Current Source Modular Multilevel Converter: Detailed Analysis and STATCOM Application. *IEEE Trans. Power Deliv.* **2016**, *31*, 323–333. [CrossRef]

3. Jung, J.-J.; Lee, H.-J.; Sul, S.-K. Control Strategy for Improved Dynamic Performance of Variable-Speed Drives with Modular Multilevel Converter. *IEEE J. Emerg. Sel. Top. Power Electron.* **2015**, *3*, 371–380. [CrossRef]

4. Mathe, L.; Burlacu, P.D.; Teodorescu, R. Control of a Modular Multilevel Converter with Reduced Internal Data Exchange. *IEEE Trans. Ind. Inform.* **2017**, *13*, 248–257. [CrossRef]

5. Lesnicar, A.; Marquardt, R. An innovative modular multilevel converter topology suitable for a wide power range. In Proceedings of the 2003 IEEE Bologna Power Tech Conference Proceedings, Bologna, Italy, 23–26 June 2003; pp. 272–277. [CrossRef]

6. Dekka, A.; Wu, B.; Fuentes, R.L.; Perez, M.; Zargari, N.R. Evolution of Topologies, Modeling, Control Schemes, and Applications of Modular Multilevel Converters. *IEEE J. Emerg. Sel. Top. Power Electron.* **2017**, *5*, 1631–1656. [CrossRef]

7. Wang, Y.; Hu, C.; Ding, R.; Xu, L.; Fu, C.; Yang, E. A Nearest Level PWM Method for the MMC in DC Distribution Grids. *IEEE Trans. Power Electron.* **2018**, *33*, 9209–9218. [CrossRef]

8. Hassanpoor, A.; Angquist, L.; Norrga, S.; Ilves, K.; Nee, H.-P. Tolerance Band Modulation Methods for Modular Multilevel Converters. *IEEE Trans. Power Electron.* **2015**, *30*, 311–326. [CrossRef]

9. Ricco, M.; Mathe, L.; Teodorescu, R. New MMC capacitor voltage balancing using sorting-less strategy in nearest level control. In Proceedings of the 2016 IEEE Energy Conversion Congress and Exposition (ECCE), Milwaukee, IL, USA, 18–22 September 2016; pp. 1–8. [CrossRef]

10. Monmasson, E.; Idkhajine, L.; Cirstea, M.N.; Bahri, I.; Tisan, A.; Naouar, M.W. FPGAs in Industrial Control Applications. *IEEE Trans. Ind. Inform.* **2011**, *7*, 224–243. [CrossRef]

11. Ricco, M.; Mathe, L.; Teodorescu, R. FPGA-based implementation of sorting networks in MMC applications. In Proceedings of the 18th European Conference on Power Electronics and Applications (EPE'16 ECCE Europe), Karlsruhe, Germany, 6 September 2016; pp. 1–10. [CrossRef]

12. Matar, M.; Paradis, D.; Iravani, R. FPGA-based Implementation of Modular Multilevel Converter Model for Real-time Simulation of Electromagnetic Transients. In Proceedings of the International Conference Power Systems Transients (IPST), Vancouver, BC, Canada, 18–20 July 2013.

13. Saad, H.; Guillaud, X.; Mahseredjian, J.; Dennetière, S.; Nguefeu, S. MMC Capacitor Voltage Decoupling and Balancing Controls. *IEEE Trans. Power Deliv.* **2015**, *30*, 704–712. [CrossRef]

14. Siemaszko, D. Fast Sorting Method for Balancing Capacitor Voltages in Modular Multilevel Converters. *IEEE Trans. Power Electron.* **2015**, *30*, 463–470. [CrossRef]

15. Liu, P.; Wang, Y.; Cong, W.; Lei, W. Grouping-Sorting-Optimized Model Predictive Control for Modular Multilevel Converter with Reduced Computational Load. *IEEE Trans. Power Electron.* **2016**, *31*, 1896–1907. [CrossRef]

16. Rashwan, A.; Sayed, M.A.; Mobarak, Y.A.; Shabib, G.; Senjyu, T. Predictive Controller Based on Switching State Grouping for a Modular Multilevel Converter With Reduced Computational Time. *IEEE Trans. Power Deliv.* **2017**, *32*, 2189–2198. [CrossRef]

17. Chen, G.; Peng, H.; Zeng, R.; Hu, Y.; Ni, K. A Fundamental Frequency Sorting Algorithm for Capacitor Voltage Balance of Modular Multilevel Converter with Low-Frequency Carrier Phase Shift Modulation. *IEEE J. Emerg. Sel. Top. Power Electron.* **2018**, *6*, 1595–1604. [CrossRef]

18. Dekka, A.; Wu, B.; Zargari, N.R. A Novel Modulation Scheme and Voltage Balancing Algorithm for Modular Multilevel Converter. *IEEE Trans. Ind. Appl.* **2016**, *52*, 432–443. [CrossRef]

19. Ghazanfari, A.; Mohamed, Y.A.-R.I. A Hierarchical Permutation Cyclic Coding Strategy for Sensorless Capacitor Voltage Balancing in Modular Multilevel Converters. *IEEE J. Emerg. Sel. Top. Power Electron.* **2016**, *4*, 576–588. [CrossRef]

20. Debnath, S.; Qin, J.; Bahrani, B.; Saeedifard, M.; Barbosa, P. Operation, Control, and Applications of the Modular Multilevel Converter: A Review. *IEEE Trans. Power Electron.* **2015**, *30*, 37–53. [CrossRef]

21. Vasiladiotis, M.; Rufer, A. Analysis and Control of Modular Multilevel Converters with Integrated Battery Energy Storage. *IEEE Trans. Power Electron.* **2015**, *30*, 163–175. [CrossRef]

22. Goetz, S.M.; Li, Z.; Liang, X.; Zhang, C.; Lukic, S.M.; Peterchev, A.V. Control of Modular Multilevel Converter With Parallel Connectivity—Application to Battery Systems. *IEEE Trans. Power Electron.* **2017**, *32*, 8381–8392. [CrossRef]

23. Perez, M.A.; Bernet, S.; Rodriguez, J.; Kouro, S.; Lizana, R. Circuit Topologies, Modeling, Control Schemes, and Applications of Modular Multilevel Converters. *IEEE Trans. Power Electron.* **2015**, *30*, 4–17. [CrossRef]

24. Sharifabadi, K.; Harnefors, L.; Nee, H.-P.; Norrga, S.; Teodorescu, R. *Design, Control and Application of Modular Multilevel Converters for HVDC Transmission Systems*; Wiley: Hoboken, NJ, USA, 2016.

25. Casadei, G.; Teodorescu, R.; Vlad, C.; Zarri, L. Analysis of dynamic behavior of Modular Multilevel Converters: Modeling and control. In Proceedings of the 16th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, 12–14 October 2012; pp. 1–6.

26. Ricco, M.; Gheorghe, M.; Mathe, L.; Teodorescu, R. System-on-chip implementation of embedded real-time simulator for modular multilevel converters. In Proceedings of the 2017 IEEE Energy Conversion Congress and Exposition (ECCE), Cincinnati, OH, USA, 1–5 October 2017; pp. 1500–1505. [CrossRef]