FPGA-based reconfigurable control for fault-tolerant back-to-back converter without redundancy

Mahmoud Shahbazi, Philippe Poure, *Member, IEEE*, Shahrokh Saadate, and MohammadReza Zolghadri, *Member, IEEE*

***Abstract*— In this paper, an FPGA-based fault tolerant back-to-back converter without redundancy is studied. Before fault occurrence, the fault tolerant converter operates like a conventional back-to-back six-leg converter and after the fault it becomes a five-leg converter. Design, implementation and experimental verification of an FPGA-based reconfigurable control strategy for this converter are discussed. This reconfigurable control strategy allows the continuous operation of the converter with minimum affection from a fault in one of the semiconductor switches. A very fast detection scheme is used to detect and locate the fault. Implementation of the fault detection and of the fully digital control schemes on a single FPGA is realized, based on a suited methodology for rapid prototyping. FPGA in loop and also experimental tests are carried out and the results are presented. These results confirm the capability of the proposed reconfigurable control and fault tolerant structure.**

***Index Terms*— Fault tolerant converter, Five-leg converter, Hardware in the Loop (HIL), Fault detection, back-to-back converter, Field Programmable Gate Array (FPGA).**

I. Introduction

BACK-TO-BACK converters are widely used in many applications like Uninterruptable Power Supplies (UPS), electrical drives and Wind Energy Conversion Systems (WECS) with series full-rate converters or with Doubly-Fed Induction Generators (DFIG). However, this type of converter is sensitive to switch faults occurring in the power electronics. Failure in the power switches is critical: it decreases the system performance and can lead to hard failure. Therefore, once a fault occurs, the system operation has to be halted. This is not desirable in the systems where the operation of the drive is of essential importance and continuous operation of the system is mandatory. Examples of such systems are some military processes, most aerospace processes and automotive industries like steering, fuel pumps, and brake-by-wire [1-3]. Also in power generation units like Distributed Generation (DG) and renewable power units, halt of the system after a fault results in financial loss.

Therefore, to prevent unscheduled shutdown, real-time fault detection and converter reconfiguration schemes for back-to-back converter must be implemented.

In order to make a suitable reaction to a fault in one of the semiconductor devices, the first step consists of fast detection of the fault and its location. Several papers have discussed fault detection schemes for power electronic converters. Fault detection in a multilevel converter is studied in [4, 5]. A detection method for faults in IGBT switches based on gate signal monitoring is presented in [6]. An observer-based method is used in [7] to detect the fault in the sensors of a converter. Open-circuit faults in matrix converters are studied in [8]. Nonlinear observers are used in [9] to detect open-switch faults in induction motor drives. Another method for detection of open-switch faults in voltage source inverters feeding AC drives based on analyzing the load currents is presented in [10]. In our earlier contributions, very fast FPGA-based fault detection is presented that is based on a "time and voltage criterion" for a two-level converter; it can detect and locate a fault in a few tens of micro-seconds [11, 12]. Here, operation of the ensemble of the driver and switch is observed for converter fault detection. Therefore once a fault in the driver or in the switch is occurred, the "driver+switch" cannot perform the desired action, and the fault must be detected. In order to have very fast fault detection, the detection algorithm must be implemented on a very fast digital target. Thanks to its parallel architecture, FPGA can run the tasks very quickly. Therefore, FPGA appears as the most suitable choice for implementation of the real-time fault detection scheme [11]. Besides, the high performance of FPGA for many power electronic and drive applications has been proved [13-16]. More, by implementing both fault detection and converter control units on a single FPGA chip, cost will be decreased. Therefore, in this paper, a FPGA is used to perform both these tasks.

Manuscript received August 7, 2011. Accepted for publication Mars 10, 2012.

M. Shahbazi is with the Sharif University of Technology, Tehran, Iran and the Université de Lorraine, Nancy, France (email: Mahmoud.Shahbazi@green.uhp-nancy.fr).
M. R. Zolghadri is with the Sharif University of Technology, Tehran, Iran (email: zolghadr@sharif.edu)
P. Poure and S. Saadate are with the Université de Lorraine, Nancy, France (phone: +33-383684160; e-mail: Philippe.Poure@lien.uhp-nancy.fr; Shahrokh.Saadate@green.uhp-nancy.fr).

Once the fault and its location are detected, appropriate changes must take place in the hardware fault tolerant converter and also in the digital control part. Several reconfigurations for different fault tolerant converters are presented in [17-19]. In our earlier studies, three-phase inverter topologies with and without a redundant leg are studied [11, 12]. Although applicable for a back-to-back converter, these approaches should be used as two separate control and detection schemes. However, another more suited possibility is to have a generalized control and fault detection scheme for a fault tolerant six-leg back-to-back converter without redundancy. In such a topology and after a fault, the six-leg converter becomes a five-leg converter. The fault-tolerant back-to-back topology studied in this paper is based on this topology, and is referred to as the "6/5 leg converter" or the "fault tolerant converter" hereafter. The associated reconfigurable control for this type of fault tolerant converter has never been studied and experimentally validated before, especially with a single FPGA chip for control, fault detection and reconfiguration. Also for the verification purpose, Hardware in the Loop (HIL) methodology is used [20, 21]. This methodology seems very reasonable when testing systems operation under a fault, in order to minimize the risk of damaging the real setup.

In this paper, a more general fault tolerant converter without redundancy is proposed for the back-to back three phase two-level converters based on a 6/5 leg topology. Fault tolerant concept is explained briefly in section II. In section III, a reconfigurable control is proposed for the control in pre-fault and post-fault conditions. In order to perform fast fault detection for the studied 6/5 leg converter, the method developed in our earlier contributions is used [11] and generalized, as discussed in section IV. It is worth mentioning that short-circuit as well as open-switch faults in the switches are concerned by this study. Fault detection and control schemes are developed for this application for implementation on a single FPGA. The implementation procedure is based on an experimental platform for rapid prototyping which is explained in section V. Hardware in the Loop and experimental tests are carried out to validate the effectiveness of this system. Finally fully experimental results are provided in section VI.

II. FAULT TOLERANT SYSTEMS

In case of fault occurrence in sensors, actuators or other parts of a complex system, conventional systems may produce undesirable results. In critical applications where the need for safety, reliability and fault tolerance is high, it is necessary to anticipate the fault tolerance ability against unpredicted faults to increase the reliability and availability of the system while providing acceptable performances. These types of systems are known as the Fault Tolerant Systems (FTS). In the other words, an FTS can maintain the stability and acceptable performance after a fault occurrence. The performance of the system in normal and post-fault conditions may be different. In the first case, the emphasis is on the quality of the system performance, while in the second case, the most important issue is to achieve a stable and acceptable (probably degraded) performance.

An FTS may be based on redundancy or not. In an FTS with redundancy, a redundant part will replace the faulty part, after fault detection and upon reconfiguration. In this case, by correctly designing the FTS, the same operation capability can be obtained. However in some cases, some degree of performance degradation is accepted. In these cases, a suitable reconfiguration might be enough to assure the minimum required performance. In Fig. 1, the redundant part and its switching mechanism are shown with dashed bounds, because they might be missed in some FTS without redundancy.

FTSs are composed of different blocks. A Fault Detection and Isolation (FDI) unit is the most essential part. Its duty is to detect the fault, find its location and its magnitude. Once the fault is detected, a suitable response must be performed to compensate its effect. In many cases, FTSs use redundancy, but one can find the schemes that do not use redundant parts or subsystems. Fig. 1 shows general FTS architecture. Upon the fault detection, the Fault Tolerant Control (FTC) block makes the suitable changes in the references, in the controller and in the system and switches the redundant part in (in the case of an FTS with redundancy).

After fault detection and isolation, a reconfiguration is necessary. Reconfiguration is required in both hardware and software parts of the system. Reconfiguration in the software part consists of modifying the references and the controller characteristics. All required control signals are generated by the FTC block.

The overall performance of an FTS depends on different factors like speed and accuracy of the fault detection unit, method of using software/hardware redundancies and/or reconfiguration, control method and connection between these different parts.

In the studied case, no redundant part (switch and driver) is used. Fault detection is implemented on a FPGA to assure very fast detection. Reconfiguration in the hardware (converter) is done using bidirectional switches (Triacs here), which make possible to change the structure from a six-leg converter to a five-leg one. Once a fault has been detected, the FTC switches and stays in the 5-leg mode until the maintenance. That is because it may be risky to go back to the normal state, since the system is clearly subjected to faults. This is especially true in safety critical systems. Software reconfiguration is applied in PWM signal generation unit, to generate the suitable gate signals based on the actual structure,

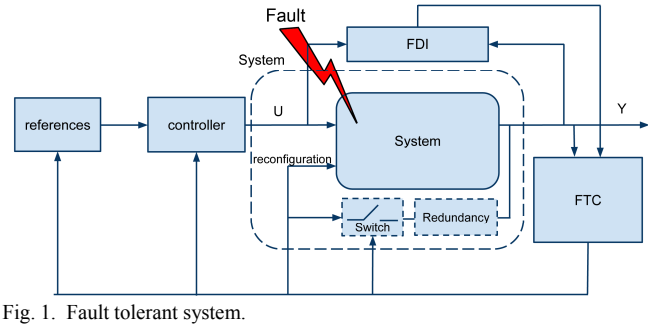

Fig. 1. Fault tolerant system.

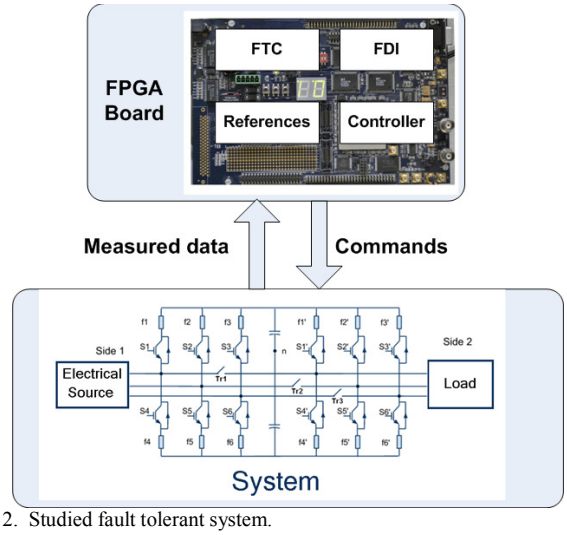

Fig. 2. Studied fault tolerant system.

as explained in section III. Fig. 2 shows the studied FTS. All the control, fault detection and isolation are implemented on a single FPGA. The system is composed of three-phase electrical sources and loads, with an AC/DC/AC fault tolerant converter between them. All different parts are explained further in detail.

III. FAULT TOLERANT SYSTEM

A. 6/5 leg converter topology

The studied converter is shows in Fig. 3 [19]. In normal operation (before the fault occurrence), the converter is a conventional back-to-back converter with three additional bidirectional switches (Fig. 3-a). These bidirectional switches are used for the converter reconfiguration after the fault isolation, i.e. before the fault occurrence these switches are all off. The converter configuration after a fault in one of the inverter legs is shown in Fig. 3-b. In this case, the fault has been occurred at either S_3' or S_6'.

Depending on the application, each side of the converter can be connected to a source, load or machine. For example, in drive application, one side might be connected to a three-phase source and the other will supply a three-phase machine. For a WECS with DFIG, side 2 will supply the rotor of the machine. In this paper, it is assumed that side 1 of the converter is connected to a three-phase balanced sinusoidal source and the second side of the converter is connected to a balanced three-phase load.

There are different options for the bidirectional switches used in this structure. The bidirectional switch can be a triac, an IGBT and a diode bridge or two anti-parallel IGBTs. In this study, triacs are used as bidirectional switches to minimize the additional cost.

It should be noted that the voltage producing capability in the five-leg converter is lower than the six-leg converter, meaning that with the same dc-link voltage, it can produce smaller three-phase voltages at its AC terminals, compared to the six-leg converter. Also the current rating in the shared leg of the converter is larger than in the other legs. Therefore,

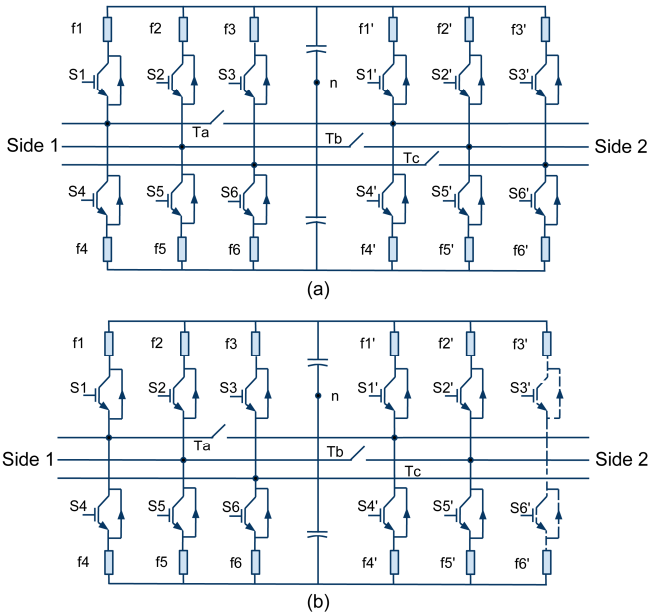

Fig. 3. 6/5 leg fault tolerant converter (a) pre-fault (b) post-fault (after reconfiguration).

when designing the fault tolerant converter, careful considerations must be taken into account to have the desired performance after reconfiguration, either by increasing the dc-link voltage or by changing the control strategy to decrease the required voltages [19,22], and also by considering the current rating of the shared leg. In this paper, it is assumed that by a proper design, the dc-link voltage is large enough to produce the desired voltages in five-leg converter mode, therefore the converter can perform in nominal capacity after the fault detection and reconfiguration.

It is also worth mentioning that in case where input and output frequencies are identical, the five leg converter can have dc-link voltage and switch currents close to the six-leg converter [19, 22]. Therefore in this case it may be possible to have almost the same power handling capability after the reconfiguration in the studied fault tolerant converter. Examples of such applications are some Uninterruptible Power Supplies (UPS). Other application may be the cases in which input and output frequencies can be forced to be equal after the fault.

B. Reconfigurable Control of the converter

Before the fault occurrence, the fault tolerant converter is performing as a conventional six-leg converter and any PWM method can be used for its control. After the fault detection and reconfiguration, one leg will be shared between the two sides of the converter and the faulty leg will be disconnected. So, the converter will become a five-leg structure. Among the several PWM approaches studied in the literature for this 5-leg post-fault topology, it seems that the suggested method in [23, 24] which uses all 32 possible voltage vectors, produces less voltage harmonics and is simpler and better suited for practical implementation. Therefore this approach is used in this paper. In this method, a so called “double zero-sequence injection

method" is performed. Voltage reference signals v_{xi}^{*} $(x\,\epsilon\{a,b,c\}, i\epsilon\{1,2\})$ for both sides of the converter are calculated using the appropriate methods depending on the targeted applications. Then, a Zero Sequence Signal (ZSS) is added to these values to form the modulation signals (1). In fact, ZSS does not change the output line-to-line and phase voltages, therefore it is used as a degree of freedom to reduce the current harmonics and improve the dc-bus utilization.

$$v_{xi}(t) = v_{xi}^{*}(t) + v_{zi}(t) \qquad (1)$$

where v_{zi} $(i\epsilon\{1,2\})$ is the ZSS for side "i".

Since there is 6 voltage references and only 5 legs, a reduction in the number of voltage references is required. Reduction of the number of voltage references can be made by using an inverse lookup table[25]. In [23], this is realized by adding another ZSS in accordance to the converter configuration in five-leg mode. The new set of voltage references assuming that two "c" legs of the two sides are connected in the five-leg mode are calculated as:

$$\begin{gathered} v_{A1} = v_{a1} + v_{c2} \;\; ; \;\; v_{B1} = v_{b1} + v_{c2} \\ v_{A2} = v_{a2} + v_{c1} \;\; ; \;\; v_{B2} = v_{b2} + v_{c1} \\ v_{C} = v_{c1} + v_{c2} \end{gathered} \qquad (2)$$

Since the same signal is added to all three reference values of the converter, the fundamental output voltage will not be affected. Fig. 4 shows the principle of this method. In Fig. 4(a) the ZSS for a conventional converter is shown. This ZSS injection will be repeated for both sets of three phase voltage references at the two sides of the converter, then by using (2) a new set of five voltage references are produced and sent to the PWM unit, as shown in Fig. 4-(b). Fig. 5 shows an example of five-leg voltage reference generation from two sets of three-phase voltage references.

C. *Voltage reference generation*

The control unit of the 6/5 leg converter produces the reference voltages and sends them to the PWM blocks. For each application case, depending on the source/load connected

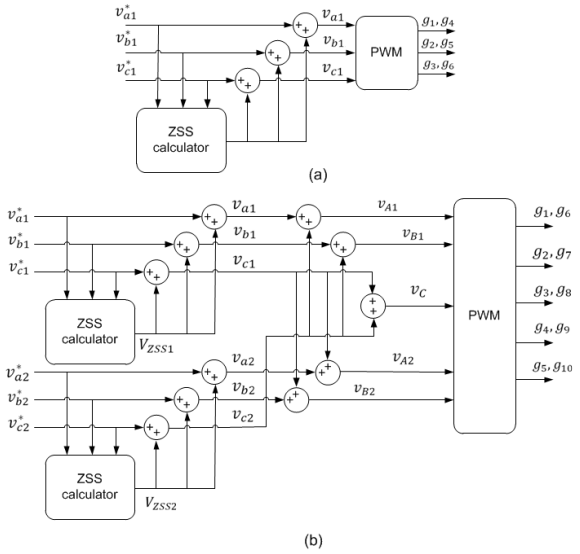

Fig. 4. (a) Principle of PWM module for one converter; (b) for 5-leg converter when "c" legs are shared.

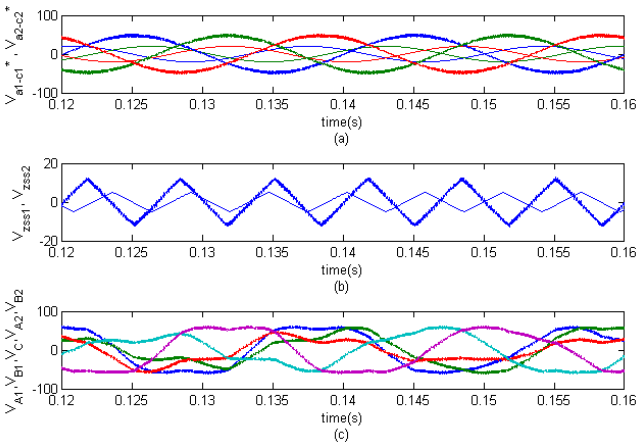

Fig. 5. Example of five-leg reference generation; (a): two sets of three-phase references; (b): two ZSSs; (c): resulted five-leg references.

at two sides of the converter, there are different methods for producing the reference voltages. For the studied application, the first side of the converter is connected to a three-phase balanced source. The objective is to have a regulated dc-link voltage with unity power factor at side 1. Side 2 of the converter is connected to a three-phase balanced RL load. It is required to have balanced sinusoidal voltages at this load side.

Therefore, the output reference voltages are sinusoidal balanced ones. The input reference voltages are calculated based on the well-known voltage-oriented control for classical three-phase rectifiers.

A general form of the reconfigured voltage reference generation and PWM can be explained as a software pseudo-code as follows. The required input data are six voltage references for the six legs of the converter, and the fault's location.

- Get six voltage references v_{lj} , $for\ all\ l,j\ where\ l\ \in \{a,b,c\}, j \in \{1,2\}$
- If there is no fault:
  - Use six voltage references v_{lj} in the PWM unit to compute the gate signals;
  - Put the gate signals for all triacs to zero;
- If a fault occurs:
  - Get the fault's location (leg $xi\ where\ x\ \epsilon\{a,b,c\}, i\epsilon\{1,2\}$) from FDI;
  - Specify the corresponding leg and Triac:
    - Corresponding Triac: T_x
    - Corresponding leg: leg $x(3-i)$
  - Make the required changes in the references and the controller, if necessary.

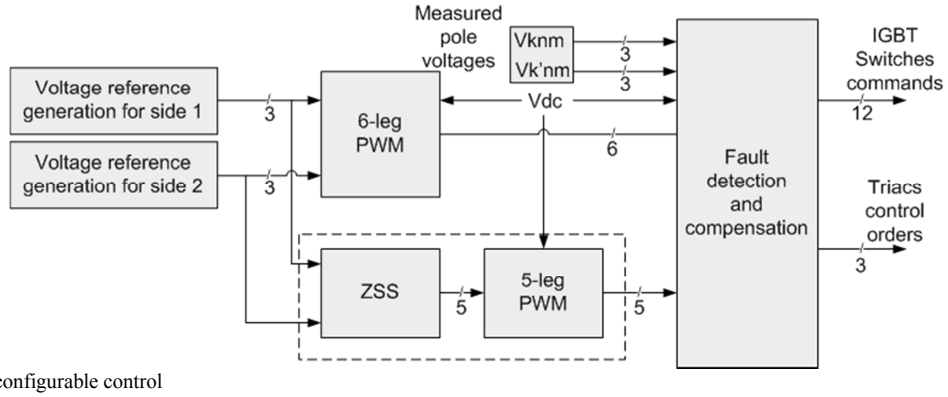

Fig.6. Proposed reconfigurable control

- For the faulty leg:
  - Cut the gate signals;
- Turn on the suited Triac;
- For the other legs (all legs $lj\ where\ l\ \epsilon\{a,b,c\}, j\epsilon\{1,2\}\ and\ lj \neq xi$)
  - $v_{lj\ new} = v_{lj} + v_{x(3-j)}$
- Use five new voltage references in the PWM unit to calculate the gate signals for the five healthy legs.

Required changes in the references might be necessary since the voltage production capability of the five-leg converter is lower than that of the six-leg converter. However, as mentioned before, in this paper it is assumed that by a proper design, the dc-link voltage is large enough to produce the desired voltages in five-leg converter mode.

D. The reconfigurable control

Fig. 6 shows the proposed reconfigurable control for the fault tolerant 6/5 leg converter. The fault detection and reconfiguration unit decides whether to use the commands from the "6-leg PWM" block, or to compute and use the appropriate 5-leg PWM signals based on the fault's location. It is important to detect the fault and change the control strategy as soon as possible, to avoid any discontinuity or undesirable transient in the inputs and outputs of the converter. Therefore, the proposed reconfigurable control of Fig. 6 is implemented on an FPGA, due to its parallel structure and high speed. In fact, system monitoring for fault detection must be performed in parallel with other system tasks. Therefore, more conventional software based processors like DSPs are not the most suited to perform very fast fault detection and control reconfiguration. This is mostly because the main performance limitation of such processors is their serialized treatment of the instructions to execute the code, also having to wait for Interrupt Service Routine (ISR) loops [14]. On the other hand, FPGA can execute all its tasks simultaneously by hardware implementation of the design. This characteristic can lead to drastic reduction in execution time [26, 27]. Moreover, especially in the studied fault tolerant application for power electronic converters, processing at logic level seems to be the best choice for real-time consideration. Programmable logic enables rapid control/corrective action, which is mandatory in fault tolerant applications. Therefore FPGA appears as an excellent choice for our application. On the other hand, it could be possible to use a software based processor to compute the switching orders, and use an FPGA for fault detection and compensation. However since the feasibility of FPGA for a variety of power electronics applications is already approved [13-16], in this paper a single FPGA is used for both purposes. In this manner, potential interface problems are avoided. In section V, the FPGA implementation is discussed. A methodology for rapid prototyping, developed in our laboratory, is used and expanded for fast implementation of the proposed reconfigurable control for the fault tolerant 6/5 leg converter.

IV. FAULT DETECTION AND RECONFIGURATION

Fast fault detection is essential for fault tolerant systems. Here the method detailed in our earlier contributions is used for detection of fault and its location [11]. Using this method, it is possible to detect the fault quickly and efficiently. In [11, 12] it is shown that the fault occurrence in each leg can be diagnosed by comparing the measured and estimated pole voltages. However, in reality, due to measurement and discretizing errors, and mainly because of non-ideal behaviors of switches and drivers (turn-off and turn-on propagation time and dead time generated by the drivers), the voltage error is not zero during normal operation. To avoid false fault detections, two adjustments are employed to compensate the effect of the measurement errors and delays. For this purpose, first the absolute value of the error between measured and estimated pole voltage is calculated. Then, this value is applied to a comparator with a threshold value 'h', to determine if the difference between the measured and estimated voltages is large enough to be considered as an error. Then, this signal is applied to an up-counter that computes the number of pulses while the output of the first comparator is high. In the other words, the output of the up-counter corresponds to the time during which v_{knm} (measured voltage) and v_{knes} (estimated voltage) are different. Consequently, the fault occurrence is detected using simultaneously a "time criterion" and a "voltage criterion". To

do this, the up-counter output is applied to a second comparator with a threshold value of 'N'. In this way, false fault detection due to semiconductor switching is avoided and fault can be detected very fast. Fig. 7 shows one leg of the converter, while there is an open-circuit fault in the upper switch. Note that in some cases, based on the direction of the current i_k, there might be a condition that D_k conducts instead of S_k; therefore in this case, the converter operates normally and the fault cannot be detected. For example, while $T_k = 1$ and $i_k < 0$, D_k is on. Here T_k is the command signal for the upper switch of the leg k. $T_k = 0$ indicates that the switch is commanded to be open, whilst $T_k = 1$ means that the switch is commanded to be closed. The switch commands in each leg are complementary.

Fig. 8 shows the mentioned fault detection principle. The state-flow diagram of the fault detection scheme is presented in Fig. 9.

For a short-circuit fault of a switch, the very fast acting fuses isolate the faulty leg. Let's now consider two converter legs as shown in Fig. 10 and suppose that there is a short circuit fault in S_k. When $T_k = 1$, the leg k operates normally. However when the $T_k = 0$, the dc-link is short circuited. Fig. 10-b shows the short circuit path. The short circuit current is limited by only the short circuit resistance, which is a small value. That is why a suitable protection is needed in this case. In this case, if the fuse clearing time is smaller than the fault detection time, the faulty leg will be isolated by the fuses. If the fault is detected before the fuses act, the command of the faulty leg switches are set at zero and the corresponding bidirectional switch will be turned on. In this case, there is still at least one fuse in the short circuit path; therefore the short circuit will be cleared after fuse operation (Fig. 10-c).

Therefore with proper choice of fuse clearing time and the fault detection time, so that the fuse clearing time is shorter than the fault detection time, the short circuit fault will result in a similar situation as an open circuit fault and can be detected by the FDI unit. So, in the following, only open switch faults will be considered.

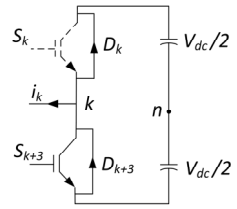

Fig. 7. One leg of converter during an open-circuit fault in the upper switch.

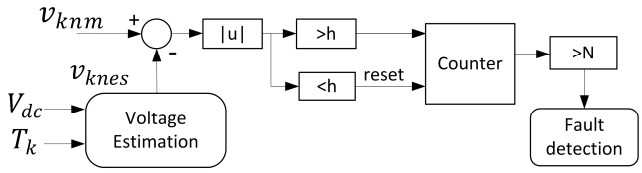

Fig. 8. Fault detection scheme.

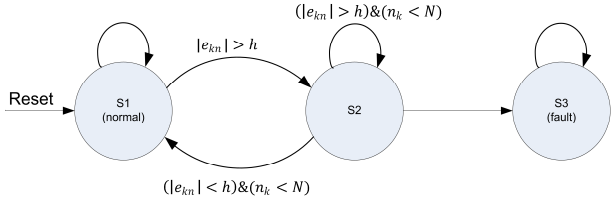

Fig. 9. State-flow diagram of the fault detection scheme.

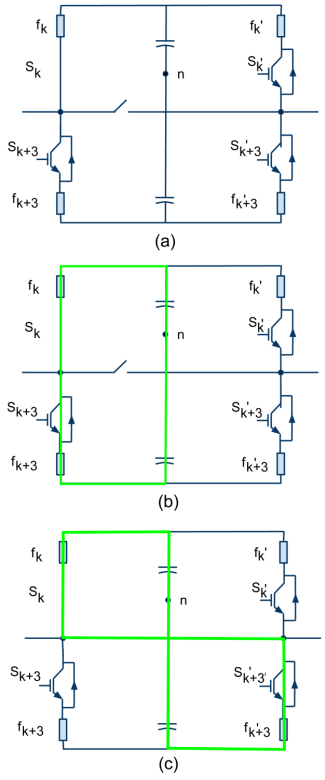

Fig. 10. Equivalent circuit for the leg "k" – (a) a short circuit fault in S_k. (b) short circuit path after the fault. (c) short circuit path when the fuse clearing time is larger than the fault detection time.

V. FPGA IMPLEMENTATION AND VALIDATION

A. Principle

As already mentioned before, all the control, fault detection and isolation parts have been implemented on one FPGA chip. However, designing and testing digital control systems for power electronics applications can be expensive and time consuming. More, traditional simulations cannot exactly reproduce the real condition, because they do not take into account some limitations of real controllers, like the limited resolution of registers or saturation of values in fixed point systems during the intermediate steps of calculations. Also the fully experimental tests may not be always possible or may be potential to damaging. One interesting solution to eliminate the risk of damaging the real plant while testing the digital controller in a realistic manner is Hardware-in-the-loop (HIL) analysis [20, 21]. It is especially interesting when examining fault tolerant systems, when faults are applied to the system and the ability of system in handling these faults may be unclear in practice.

In order to carry out the FPGA implementation for HIL experiments, a top-down design flow is used, as shown in Fig. 11. The flow we proposed consists of four parts: fundamental simulations, mixed simulation, HIL and finally, fully experimental test. This procedure is explained in detail in the following.

After HIL experiment, when the effectiveness of the FPGA implementation is experimentally approved, it is possible to move to the fully experimental tests.

B. Fault tolerant control implementation

The first step in the process of implementation of the control and detection units on a single FPGA chip is the fundamental simulation step. In this step, the studied system is modeled and simulated in the Matlab/Simulink environment, using Simulink and SimPowerSystem blocks. First continuous-time and then discrete-time simulations are performed.

In the mixed simulation stage, the control blocks are replaced with proper Dspbuilder blocks. Dspbuilder provides the possibility of visual programming of the control system in MATLAB-Simulink environment. Each Dspbuilder block represents a VHDL module in the Simulink environment. This set of blocks is very efficient for rapid prototyping of FPGA devices. However some of desired functions are not available in Dspbuilder library and must be constructed from basic blocks or imported using HDL programming. The power part (fault tolerant converter, sources and charges) is remained unchanged in this step, modeled by using SimPowerSystem blocks. Appropriate Dspbuilder input and output blocks are necessary to convert the Simulink signals to fixed point signals for Dspbuilder.

Finally in the third step, having validated Dspbuilder modeling by simulations, the blocks are translated into VHDL. This is done by using the signal compiler block which is available in the Dspbuilder library. After this step, a single HIL block replaces all of the Dspbuilder blocks. The VHDL design is compiled and downloaded to the FPGA via a Joint Test Action Group (JTAG) interface. Now the HIL block represents the FPGA in the Simulink environment. In each step of simulation, it gathers the inputs from system ($i_{a1}, i_{b1}, i_{c1}, V_{dc}$ …) and sends them to the programmed FPGA. Using these inputs and based on the control schemes of the system, the FPGA calculates the gate commands for all switches and sends them back to Simulink environment. At this point one cycle of the "FPGA in the loop" experiment is performed. Fig. 12 shows the "FPGA in loop prototyping".

In our experiment, a Stratix DSP S80 development board is used, which includes the Stratix EP1S80B956C6 FPGA chip. This chip contains 79,040 programmable logic elements. The development board has an on-board 80-MHz oscillator.

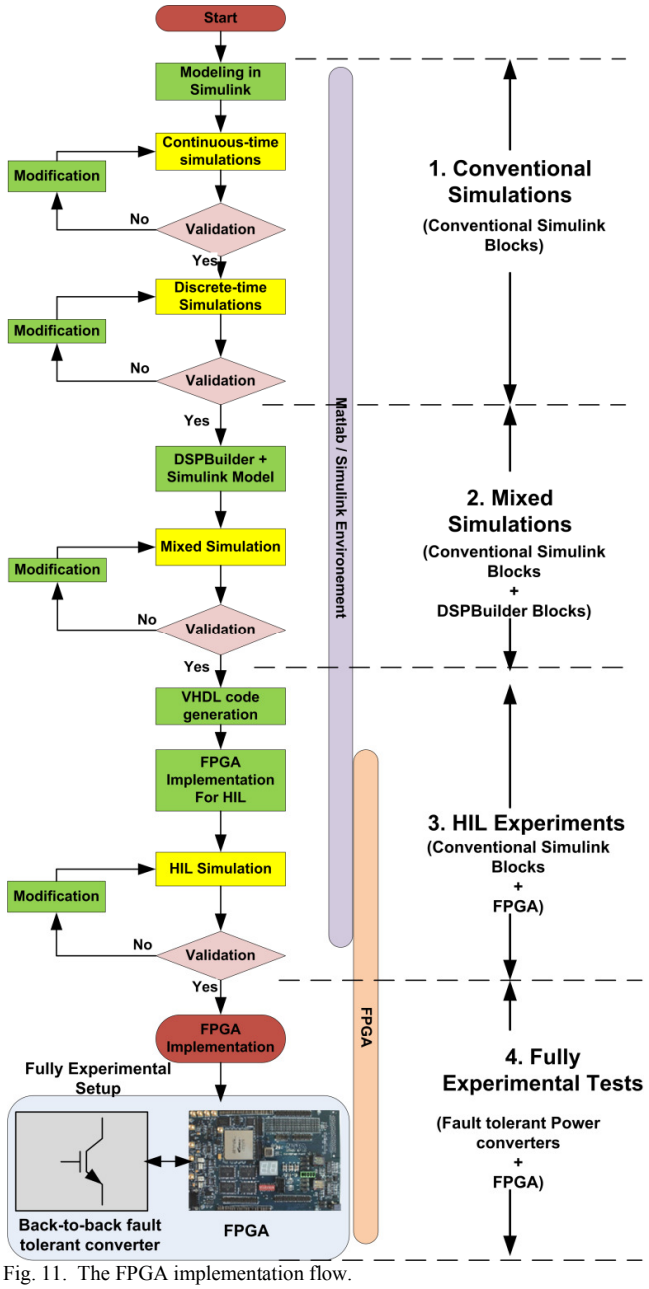

Fig. 11. The FPGA implementation flow.

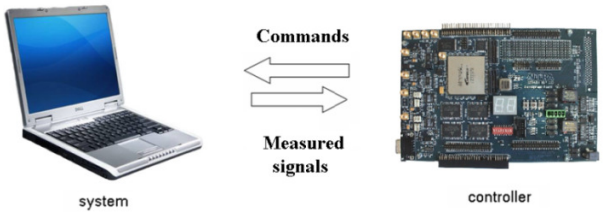

Fig. 12. "FPGA in loop" prototyping.

VI. VALIDATION AND RESULTS

"FPGA in the loop" and fully experimental tests are carried out to evaluate first the performances of the FPGA implementation of the reconfigurable fault tolerant control and then the fault tolerant converter topology. First, HIL results are shown and analyzed. Then the results of fully experimental tests are presented and discussed.

A. HIL results

For HIL experiment, the controller and detection schemes are implemented on the same FPGA chip and the power

system is modeled in MATLAB/Simulink environment using the SimPowerSystem toolbox. As stated earlier, three-phase voltages at the source side are controlled to provide the ability of the dc-link voltage regulation at unity input power factor. This is done using a well known source-voltage oriented control method for three phase controlled rectifiers [28]. Voltage references at the load-side are balanced and sinusoidal.

Two sets of voltage references obtained from these control schemes are sent to the appropriate six-leg or five-leg PWM blocks, based on the state of the converter. PWM block for a five leg converter has a preprocessing unit as stated in (1) and (2).

System parameters are reported in Table I. An open-circuit fault is introduced in one of the semiconductor switches in order to evaluate the response of the fault detection scheme and to validate the effectiveness of the fault tolerant converter.

TABLE I- SYSTEM PARAMETERS

Load impedance	$R_L = 2.75\,, L_L = 9mH$
Source filter impedance	$R_f = 0.4\Omega\,, L_f = 3mH$
DC-link capacitance	$C = 2200\ \mu F$
PWM carrier frequency	$f_c = 8kHz$
Detection Parameters	N=32 , h=10

First, an open switch fault is applied at time t = 0,5s in the lower switch of the leg '3'' in the inverter side (consisted of S_3' *and* S_6', in Fig. 3). The goal is to evaluate the proposed reconfigurable control and its implementation on the FPGA. Fig. 13 shows the fault occurrence and detection, and also the output of the counter in the corresponding detection unit (Fig. 8). A detection time of 32 μs is chosen to take into account all delays in the system. It is clear that the fault is detected very fast.

It is worth mentioning that due to the mentioned delays in drivers and switches, there are small spikes in the counter's output at the switching instants, as shown in Fig. 13. However by choosing a large enough value of N, false fault detection is avoided.

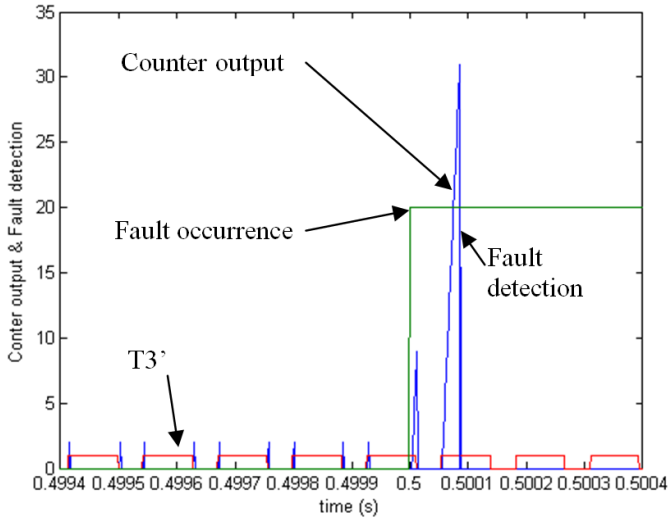

Fig. 13. Fault occurrence and detection. Fault is detected when the counter output reaches the value of 32.

Here, for a fault at t=0.5s, i_3' is negative at the fault occurrence moment; therefore, the fault can be detected instantaneously. Fig. 14 shows the dc-link voltage upon the fault occurrence. Load-side currents are shown in Fig. 15. Clearly, the detection and reconfiguration scheme is capable of maintaining the converter performance without being really affected by the fault.

However, if the fault occurs in S_6' while $i_3' > 0$, the diode D_6' conducts instead of S_6', so the converter acts normally and the fault cannot be detected. This situation is shown in Figs. 16-18. In this case, an open switch fault is applied in S_6' at t=0.486s, when $i_3' > 0$. Fig. 16 shows the fault occurrence and detection times, the output of the counter in the corresponding detection unit and i_3'. Clearly, the converter acts normally when $i_3' > 0$, and fault detection is possible when the current becomes negative. A zoomed view of Fig. 16 is repeated in Fig 17. Fig. 18 shows the load side currents for a fault in S_6' at t=0.486s. Dc-link voltage for this case is shown in Fig. 19. Obviously again in this case the fault detection and reconfiguration is successfully carried out. Evidently, even in the case of a fault occurrence, the load is still supplied with the desired currents.

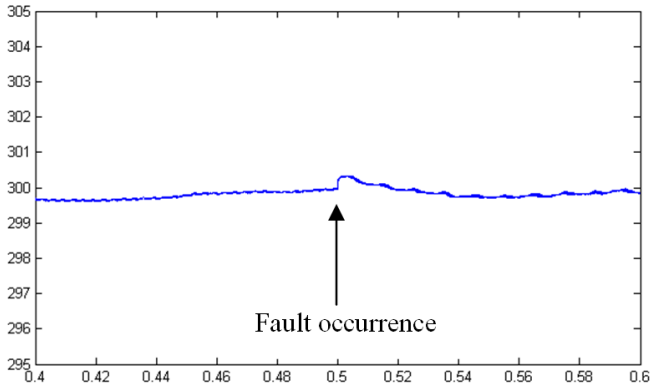

Fig. 14. Dc-link voltage (fault at 0.5s).

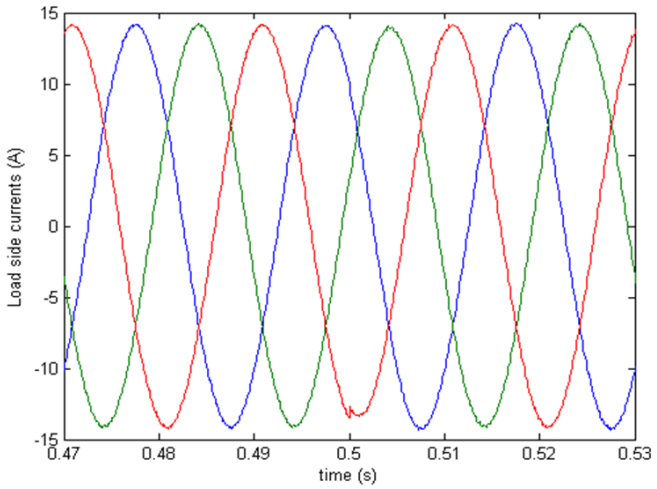

Fig. 15. Load currents (fault at 0.5s).

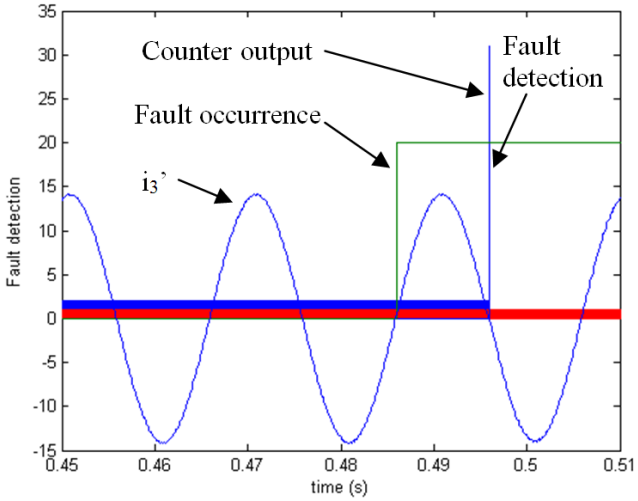

Fig. 16. Fault detection when $i_3' > 0$ (fault at t=0.486s).

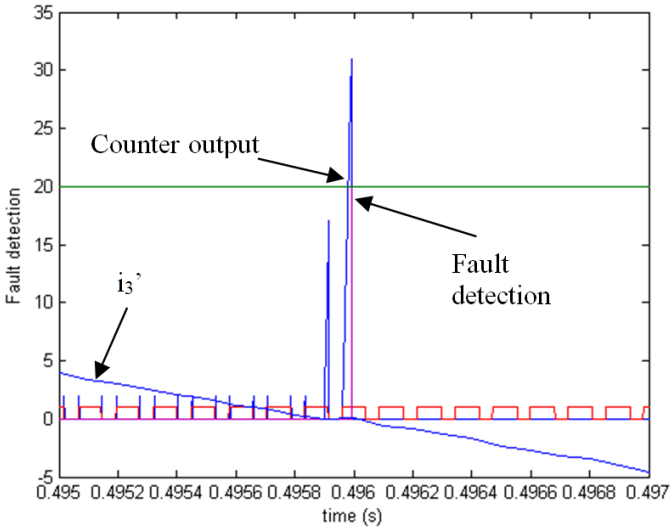

Fig. 17. Zoomed view of detection waveforms when i_3'>0 (fault at t=0.486s).

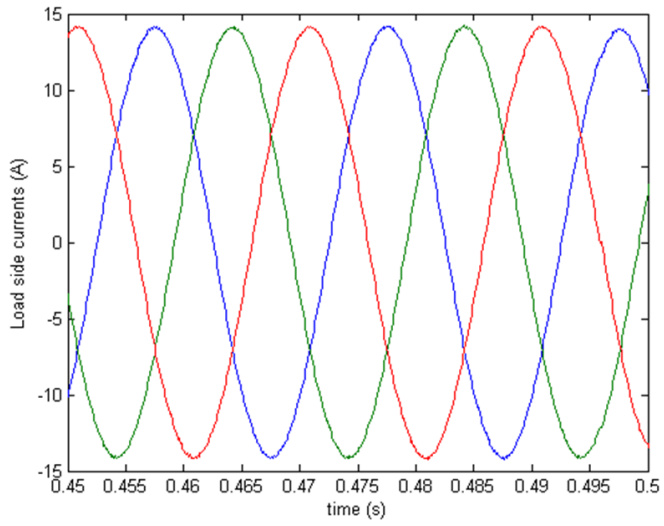

Fig. 18. Source currents for fault when i_3'>0 (fault at t=0.486s).

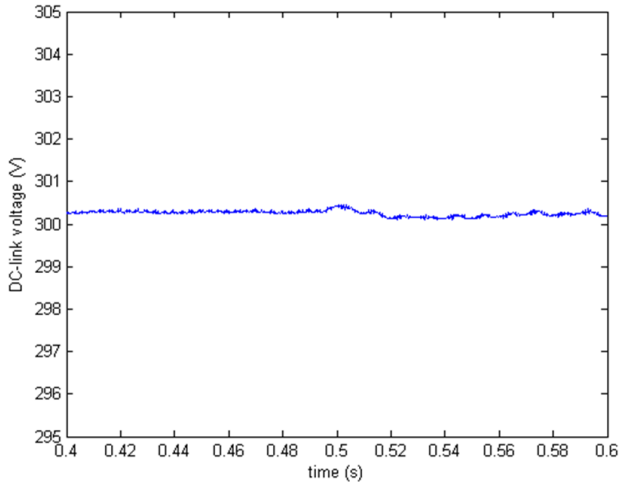

Fig. 19. V_{dc} for fault when i_3'>0 (fault at t=0.486s)

B. Experimental results

Now that the effectiveness of the fault detection and control schemes have been approved experimentally by HIL (on a FPGA device), it is possible to proceed to fully experimental validation. Experimental results are presented here. An experimental setup is built in our laboratory for this purpose, as shown in Fig. 20. Two three-phase IGBT converters and additional Triacs form the fault-tolerant converter. Parameters are the same as reported in Table I. A three-phase RL load is supplied by the studied fault-tolerant back-to-back converter. Control and fault detection is realized by using the FPGA. An open circuit fault is introduced in the lower switch of the leg "3′" at the inverter side (see Fig. 3). The open circuit fault is realized through the control signal of the switch, and no fuse component is used.

For IGBTs, SKM50GB123D of SEMIKRON are used. These IGBTs are commanded by SKHI22A drivers. When using commercial three-phase two-level converters, it is possible to easily assemble the proposed fault tolerant converter by using two three-phase converters and connecting the corresponding AC side outputs with three TRIACS, as shown in Fig. 3. For fault detection and reconfigurable control, it is necessary to have the voltage references which are produced by the controller, as well as the measured voltages, as shown in Fig. 6. Having this necessary information, it will be possible to generate the command signals for the IGBTs and TRIACs of the converter (Fig. 6).

A 2200 µF capacitance is placed at the dc-link. Side 1 of the fault tolerant converter is connected to a three-phase sinusoidal source via a three phase inductance of 3mH and resistance of 0.4 Ω. A three-phase RL load with a resistance of 2.75 Ω and an inductance of 9mH is connected to the load side of the converter. N=32 is chosen for fault detection, so the detection time is around 30 μs.

For voltage sensors, LEM CV3-1200 is used. AMP02E op-amp is used in order to amplify the output of voltage sensor, for analogue to digital conversion. The measured voltage is then digitalized using an ADS7810 Analog to Digital

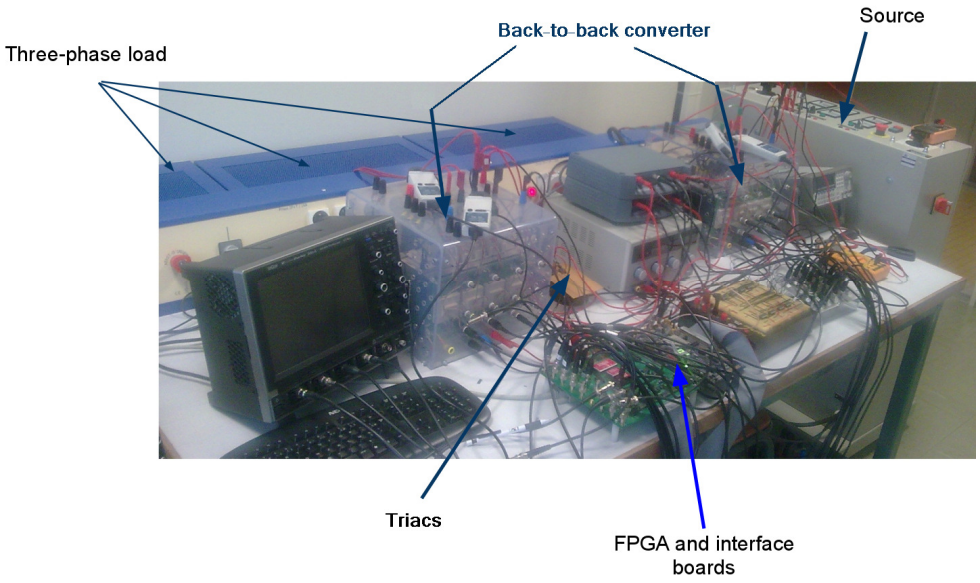

Fig. 20. Fully experimental setup.

Converter (ADC). The logic states at the output of the ADC are memorized during the conversion process using SN74HC174N D-type flip flops.

Fig. 21 shows the currents of the phase "3′" of the load (which was connected to the faulty leg) and of the phase "1" of the source as examples of the converter currents before and after the fault occurrence. Fault appears when $i_3' < 0$ and therefore the fault can be detected. Clearly, even in the presence of a fault, the load is still supplied with the desired currents and the source current is also unchanged. Fig. 22 shows a zoomed view of the fault detection process waveforms. The moments of fault occurrence, fault detection and the output of the up-counter unit (see Fig. 8) are shown in this figure. The fault is detected very fast. Note the low-amplitude changes in the counter's output that are due to delays in the control and measurement. Having defined a sufficiently large band (N) in the detection unit, these spikes are not considered as a fault, and therefore robustness of the fault detection method against false-alarms is experimentally achieved.

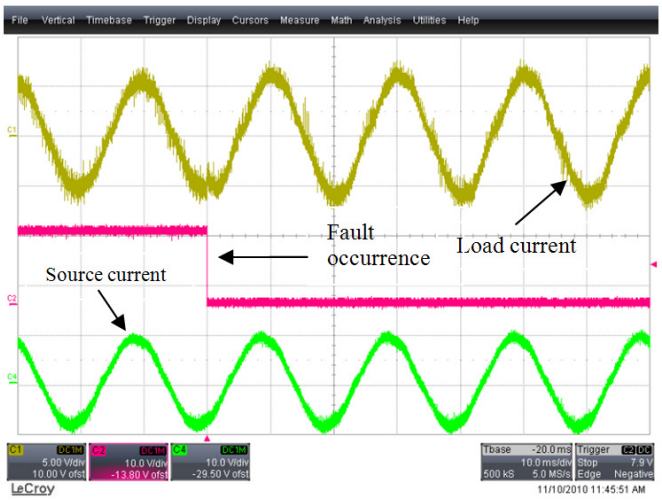

Fig. 21. Experimental results-from top to bottom: Load current (5A/div), Fault, source current(10A/div)- (x-axis: 10 ms/div).

In Fig. 23, the dc-link voltage is also shown, and it can be seen that the effect of the fault on the dc-link voltage is negligible.

If $i_3' > 0$ at the fault occurrence moment, the converter will continue to operate normally and therefore the fault cannot be detected. This is shown in Figs. 24 and 25. In this case, the

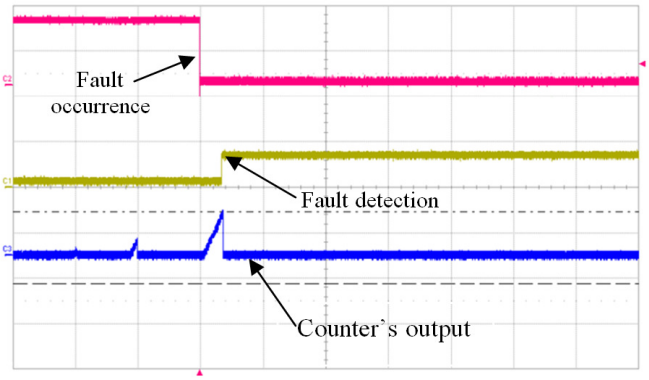

Fig. 22. From top to bottom: fault, fault detection and counter output- (x-axis: 100 μs/div).

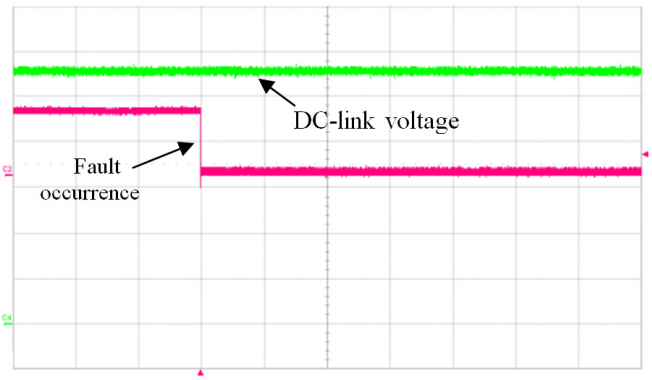

Fig. 23. From top to bottom: dc-link voltage (50V/div), fault- (x-axis: 100 μs/div).

fault will be detected upon the zero crossing of the defected leg's current, as shown in Fig. 24.

Experimental results are in accordance with the HIL ones. It is visible that in all studied cases, the fault tolerant converter can continue supplying the load and the proposed fault detection and reconfigurable control produces satisfactory results. These results show excellent operation of the proposed reconfigurable fault detection and control, as well as its FPGA implementation and the studied fault tolerant back-to-back converter.

VII. CONCLUSION

This paper presents and validates a high performance fault detection and reconfigurable control for a fault tolerant converter. A three-phase fault tolerant back-to-back converter with a very fast fault detection scheme and suited

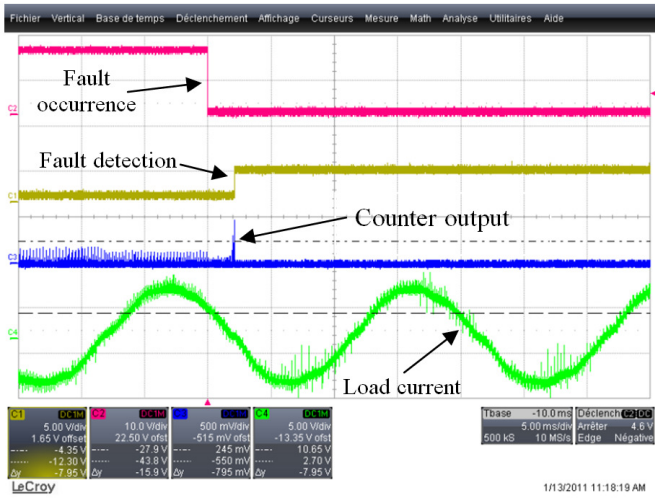

Fig. 24. Fault occurrence when i_3'>0; from top to bottom: Fault, Fault detection, counter's output, Load current (5A/div) - (x-axis: 5 ms/div).

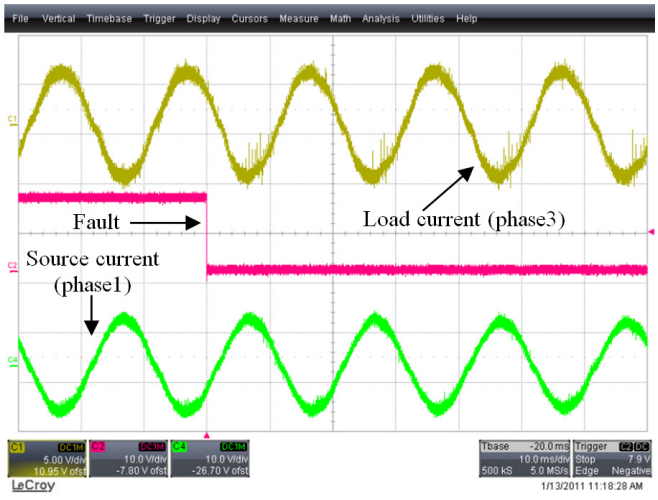

Fig. 25. Fault occurrence when i_3'>0; from top to bottom: Load current, Fault, Source current (5A/div) - (x-axis: 10 ms/div).

reconfigurable control is studied. A fully digital controller, fault detection scheme and a reconfigurable control strategy are proposed for this topology. The proposed reconfigurable control allows the continuity of service of the converter with minimum affection from the fault. More, in the proposed scheme, a single FPGA target is used for implementation of both reconfigurable control and fault detection units, therefore the total cost of the FTS is lowered. Rapid prototyping methodology is used to implement the detection and control units on the FPGA chip. This methodology is based on discrete simulation in Matlab/Simulink environment, using Dspbuilder blocks to replace the Simulink blocks, HIL experiments and finally experimental tests. Hardware in the Loop and fully experimental tests' results are presented to demonstrate the effectiveness of the proposed fully digital reconfigurable control and detection unit for the studied fault tolerant applications. Faults are detected in a few tens of microseconds and the transition during the system reconfiguration is soft. Examples of the possible applications of this structure are wind energy conversion systems with doubly-fed induction generators and applications with two three-phase inverters fed from a single dc-bus.

REFERENCES

[1] Man Ho Kim, Suk Lee, Kyung Chang Lee, "Kalman Predictive Redundancy System for Fault Tolerance of Safety-Critical Systems ," Industrial Informatics,IEEE Tranactions on, Vol. 6, No. 1, pp. 46-53, Feb 2010.

[2] C.H. Lo, E.H.K. Fung, Y.K. Wong, "Intelligent Automatic Fault Detection for Actuator Failures in Aircraft," Industrial Informatics, IEEE Tranactions on, Vol. 5, No. 1, pp. , Feb 2009.

[3] B.A. Welchko, T.A. Lipo, T.M. Jahns and S.E. Schulz, "Fault tolerant three-phase AC motor drive topologies: A comparison of features, cost, and limitations," Power Electronics, IEEE Transactions on, Vol. 19, No.4, pp. 1108-1116, Jul 2004.

[4] A. Yazdani, H. Sepahvand M. Crow and M. Ferdowsi, "Fault Detection and Mitigation in Multilevel Converter STATCOMs," Industrial Electronics, IEEE Transactions on, Vol. 58, No. 4, pp. 1307-1315, 2011.

[5] P.Lezana, R.Aguilera, and J.Rodriguez, "Fault detection on multicell converter based on output voltage frequency analysis", Industrial Electronics, IEEE Transactions on, Vol.56, No.6, pp. 2275-2283, 2009.

[6] M.A. Rodríguez-Blanco, A. Claudio-Sánchez, D. Theilliol, L.G. Vela-Valdés, P. Sibaja-Terán, L. Hernández-González, J. Aguayo-Alquicira, "A Failure-Detection Strategy for IGBT Based on Gate-Voltage Behavior Applied to a Motor Drive System", Industrial Electronics, IEEE Transactions on, Vol. 58 , No. 5, pp. 1625 – 1633, 2011.

[7] T.A.Najafabadi, F.R. Salmasi, P. Jabehdar-Maralani, "Detection and Isolation of Speed-, DC-Link Voltage-, and Current-Sensor Faults Based on an Adaptive Observer in Induction-Motor Drives", Industrial Electronics, IEEE Transactions on, Vol. 58 , No. 5,pp. 1662 – 1672, 2011.

[8] S. Cruz, M. Ferreira, A. Mendes, and A. J. M. Cardoso, "Analysis and Diagnosis of Open-Circuit Faults in Matrix Converters," Industrial Electronics, IEEE Transactions on, Vol. 58 No.5 pp. 1648 – 1661, May 2011.

[9] D.U. Campos-Delgado and D.R.Espinoza-Trejo, " An Observer-Based Diagnosis Scheme for Single and Simultaneous Open-Switch Faults in Induction Motor Drives" Industrial Electronics, IEEE Transactions on, Vol. 58 No.5 pp. 671-679, 2011.

[10] W.Sleszynski, J.Nieznanski, and A.Cichowski, "Open-transistor fault diagnostics in voltage-source inverters by analyzing the load currents", Industrial Electronics, IEEE Transactions on, Vol.56, No.11, pp. 4681-4688, 2009.

[11] S. Karimi, A.Gaillard, P.Poure and S.Saadate, "FPGA-Based Real-Time Power Converter Failure Diagnosis for Wind Energy Conversion Systems," Industrial Electronics, IEEE Transactions on, Vol. 55, No. 12, pp. 4299-4308, Dec 2008.

[12] S. Karimi, P.Poure and S.Saadate, "FPGA-based fully digital fast power switch fault detection and compensation for three-phase shunt active filters," Electric Power Systems Research, Vol. 78, No. 11, pp. 1933-1940, Nov 2008.
[13] E. Monmasson, L. Idkhajine, M.W. Naouar,"FPGA-based Controllers" Industrial Electronics Magazine, IEEE, Vol. 5, No. 1, pp. 14-26, Mar 2011.
[14] E. Monmasson, L. Idkhajine, M. N. Cirstea, I. Bahri, A. Tisan, and M. W. Naouar, "FPGAs in Industrial Control Applications" Industrial Informatics, IEEE Transactions on, No. 99, pp. 1-1, 2011.
[15] M. Hamouda, H. Blanchette, K. Al-Haddad, and F. Fnaiech, "An efficient DSP-FPGA based real time implementation method of SVM algorithms for an Indirect Matrix Converter," Industrial Electronics, IEEE Transactions on, in press.
[16] F.Taeed, Z.Salam, Md. S.Ayob, "FPGA Implementation of a Single Input Fuzzy Logic Controller for Boost Converter with the Absence of an External Analog to Digital Converter," Industrial Electronics, IEEE Transactions on; in press.
[17] D.U. Campos-Delgado, D.R. Espinoza-Trejo, and E. Palacios, "Fault-tolerant control in variable speed drives: a survey," IET Electric Power Applications, Vol. 2, No.2, pp. 121-134, Mar 2008.
[18] W.Sae-Kok, D.M. Grant and B.W. Williams, "System reconfiguration under open-switch faults in a doubly fed induction machine," IET Renewable Power Generation, Vol. 4,No.5, pp. 458-470, Sep 2010.
[19] C. B. Jacobina, R.L. de Araujo Ribeiro, A.M.N. Lima and E.R.C. da Silva, "Fault-tolerant reversible AC motor drive system," Industry Applications, IEEE Transactions on, Vol. 39, pp. 1077-1084, Jul-Aug 2003.
[20] S. Karimi, P.Poure and S.Saadate, "An HIL-Based Reconfigurable Platform for Design, Implementation, and Verification of Electrical System Digital Controllers," Industrial Electronics, IEEE Transactions on, Vol. 57, No. 4, pp. 1226-1236, Apr 2010.
[21] L. A. Gregoire, K. Al-Haddad, and G. Nanjundaiah, "Hardware-in-the-Loop (HIL) to reduce the development cost of power electronic converters," IICPE2010 India, pp. 1-6, 2011.
[22] M.Shahbazi, P.Poure, S.Saadate, M.R. Zolghadri, "Five-leg converter topology for wind energy conversion system with doubly fed induction generator," Renewable Energy, Vol. 36, No. 11, pp. 3187-3194 November 2011.
[23] M. Jones, et al., "Five-leg inverter PWM technique for reduced switch count two-motor constant power applications," IET Electric Power Applications, Vol. 2, No.5, pp. 275-287, Sep 2008.
[24] M. Jones, D. Dujic and E. Levi, "A performance comparison of PWM techniques for five-leg VSIs supplying two-motor drives," IECON 2008, pp. 508-513.
[25] A. Bouscayrol, B. Francois, P. Delarue and J. Niiranen, "Control implementation of a five-leg AC-AC converter to supply a three-phase induction machine," Power Electronics, IEEE Transactions on, Vol. 20, pp. 107-115, Jan 2005.
[26] Z. Shu, Y.Guo and J. Lian, "Steady-state and dynamic study of active power filter with efficient FPGA-based control algorithm," Industrial Electronics, IEEE Transactions on, Vol. 55, No. 4, pp. 527-1536, 2008.
[27] A. de Castro, P.Zumel, O.Garcia, T.Riesgo, J.Uceda,"Concurrent and simple digital controller of an AC/DC converter with power factor correction based on an FPGA," Power Electronics, IEEE Transactions on, Vol.18, No.1, pp. 334-343, 2003.
[28] Mariusz Malinowski , "Sensorless Control Strategies for Three - Phase PWM Rectifiers" , PhD thesis, Warsaw University of Technology, 2001

Mahmoud Shahbazi was born in Mahallat, Iran in 1983. He received the B.S. degree in electrical engineering from the Isfahan University of Technology, Isfahan, Iran, in 2005 and the M.S. degree in electrical engineering from Amirkabir University of Technology, Tehran, Iran, in 2007. He is currently working toward his Ph.D. degree in collaboration between Université de Lorraine, Nancy, France, and Sharif University of Technology, Tehran, Iran.
His research interests are wind energy conversion systems, Power electronic converters and fault tolerant converters.

Philippe POURE was born in 1968. He received the Engineer Degree and Ph.D. Degree in Electrical Engineering from INPL-ENSEM-GREEN, France, in 1991 and 1995 respectively. From 1995 to 2004, he was an Associate Professor and worked at the University Louis Pasteur of Strasbourg, France, in the field of mixed-signal System-On-Chip for control and measurement in Electrical Engineering. Since September 2004, he joined the Université de Lorraine, Nancy - France and works on fault tolerant power systems and FPGA based real time applications.

Shahrokh SAADATE, born in Tehran/IRAN on May 6th, 1958, received his diplôme d'ingenieur (1982), DEA (1982), thèse de doctorat (1986) and Habilitation à diriger les recherches (1995) from ENSEM, INPL, GREEN laboratory Nancy / France. Currently, he is the head of GREEN Laboratory at the University of Lorraine in Nancy France. His main research domain is power systems reliability, power quality and renewable energies.

MohammadReza Zolghadri received his B.S and M.S. degree from Sharif University of Technology, Terhran, IRAN and Ph.D. in Electrical Engineering from Institute National Polythechnique de Grenoble (INPG), Grenoble, FRANCE all in Electric al Engineering in 1989, 1992 and 1997 respectively. Since 1997 he is with the department of Electrical Engineering at Sharif University of Technology. From 2000 to 2003 he was a senior researcher in the Electronics Lab of SAM Electronics Co. Tehran, IRAN. From 2003 to 2005 he was a visiting professor in the North Carolina A&T State University, Greensboro,NC, USA. His fields of interests are: application of power electronics in renewable energy systems and HEV, variable speed drives, and modeling and control of power electronic converters. Dr Zolghadri is the author of more than 70 publications in power electronics and variable speed drives.