Multilevel Simulation Methodology for FMECA Study Applied to a Complex Cyber-Physical System

electronics

MDPI

Article

Multilevel Simulation Methodology for FMECA Study Applied to a Complex Cyber-Physical System

Davide Piumatti [1,*], Jacopo Sini [1], Stefano Borlo [2], Matteo Sonza Reorda [1], Radu Bojoi [2] and Massimo Violante [1]

[1] Dipartimento di Automatica e Informatica (DAUIN), Politecnico di Torino, 10129 Turin, Italy; jacopo.sini@polito.it (J.S.); matteo.sonzareorda@polito.it (M.S.R.); massimo.violante@polito.it (M.V.)

[2] Dipartimento di Energia (DENERG), Politecnico di Torino, 10129 Turin, Italy; stefano.borlo@polito.it (S.B.); radu.bojoi@polito.it (R.B.)

* Correspondence: davide.piumatti@polito.it

Received: 4 September 2020; Accepted: 16 October 2020; Published: 21 October 2020

Abstract: Complex systems are composed of numerous interconnected subsystems, each designed to perform specific functions. The different subsystems use many technological items that work together, as for the case of cyber-physical systems. Typically, a cyber-physical system is composed of different mechanical actuators driven by electrical power devices and monitored by sensors. Several approaches are available for designing and validating complex systems, and among them, behavioral-level modeling is becoming one of the most popular. When such cyber-physical systems are employed in mission- or safety-critical applications, it is mandatory to understand the impacts of faults on them and how failures in subsystems can propagate through the overall system. In this paper, we propose a methodology for supporting the failure mode, effects, and criticality analysis (FMECA) aimed at identifying the critical faults and assessing their effects on the overall system. The end goal is to analyze how a fault affecting a single subsystem possibly propagates through the whole cyber-physical system, considering also the embedded software and the mechanical elements. In particular, our approach allows the analysis of the propagation through the whole system (working at high level) of a fault injected at low level. This paper provides a solution to automate the FMECA process (until now mainly performed manually) for complex cyber-physical systems. It improves the failure classification effectiveness: considering our test case, it reduced the number of critical faults from 10 to 6. The remaining four faults are mitigated by the cyber-physical system architecture. The proposed approach has been tested on a real cyber-physical system in charge of driving a three-phase motor for industrial compressors, showing its feasibility and effectiveness.

Keywords: FMECA; system simulation; complex system; cyber-physical system

1. Introduction

Complex systems are composed of devices belonging to different technological areas. For example, cyber-physical systems often include power subsystems implemented by combining power devices, analog low-voltage circuits, and digital devices. Moreover, microcontrollers that execute embedded control software are normally employed. Furthermore, sensors and mechanical devices such as power electrical motors or gears may be present. Complex systems are modular, i.e., composed of numerous subsystems connected to each other, each of them designed to perform a specific function as defined by the precise relationships between its inputs and outputs signals. For each subsystem, a high-level model composed of a set of input–output relationships is first created; this model is called the behavioral model of the subsystem. Afterwards, the subsystem is implemented resorting to different components. In a subsystem, the ensemble of its suitably connected components constitutes a possible low-level

model of the subsystem (the so called structural model). The electrical components are connected to each other creating an electrical model of the subsystem (circuit diagram). Therefore, the circuit diagram of a subsystem corresponding to an electronic circuit represents its structural low-level model. The whole complex system can be simulated resorting to the structural or behavioral models of the different subsystems. Generally, high-level models are used to perform behavioral simulations of the whole system, while structural models are used for detailed simulations of the single subsystem. Usually, each subsystem is simulated at low level by itself to avoid long simulation times.

Many cyber-physical systems are used in safety-critical applications; different international standards have been proposed for handling the design and production of safety-critical applications used in different areas, e.g., aviation, automotive, medical, and industrial. In each of the areas indicated in Figure 1, a dedicated standard has been defined; most of the standards derive from the IEC 61508 [1] which manages the overall life cycle of the product. The fundamental concept introduced in IEC 61508 is that a system must function correctly or, at least, fail in a predictable and safe way. The purpose of the standards, in each area, is to define methods for applying, designing, distributing, and maintaining automatic protection systems for each specific application. In these standards, failure mode, effects, and criticality analysis (FMECA) is listed among the possible techniques for analyzing the items that compose the systems [2–4]. As discussed in [5], an item can be a single specific subsystem, a set of subsystems, or a device present in a subsystem. In general, FMECA is performed after the design to determine if some of the faults that can affect the components prevent the system from satisfying the safety level associated with its functions.

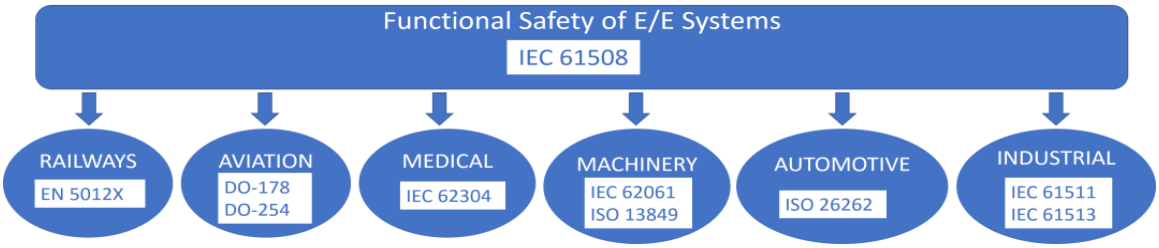

Figure 1. Safety international standards.

These FMECA approaches can be used to study and analyze the effects of a fault affecting power devices and power applications as well, as described in [6–13]. For example, in [6,7,10,11], the FMECA methodology is proposed for the systems dedicated to the generation of electricity, such as solar photovoltaic and wind power. Instead, in [12,13], a methodology for automotive safety-critical applications that use power circuits has been recently proposed, following the ISO26262 standard [5]. The different safety standards require studying the behavior of electrical and electronic systems in the presence of faults, in order to estimate figures as the mean time to failure, and for verifying the effectiveness of the safety mechanisms used to mitigate the effects of the faults, by producing other figures such as the diagnostic coverage. With the growing complexity of the designed systems, it is necessary to introduce strategies that allow analyzing the effects of faults automatically and systematically. These strategies are essential to support the designer of complex systems when dealing with safety-critical ones. The FMECA analysis can be performed with a simulation of the whole system; in fact, in the event of a failure of a subsystem, it is necessary to understand the effects that a failed subsystem has on other subsystems. In this way, the possible propagation of the fault effects through the different subsystems can be studied.

The contribution of this paper is to propose a methodology for effectively performing the FMECA analysis required by international safety standards in different areas. The novelty introduced in this paper mainly lies in the simulation approach proposed, which allows for considering low-level faults in each subsystem and to analyze their impact on the whole cyber-physical system, resorting to high-level simulation to propagate their effects. In particular, the proposed approach can identify the critical

faults, whose number is often overestimated by other approaches. The proposed methodology relies on a multilevel simulation of a cyber-physical system that uses behavioral high-level models and structural low-level models of the different subsystems. In the proposed simulation methodology, all the subsystems of the complex system are described with a behavioral model, except for the target one, called the subsystem under test (SSUT), where the generic fault under analysis is located. The SSUT is described with its structural low-level model to allow for properly modeling faults affecting it. Moreover, the effect of the fault injected in the SSUT is propagated to the other subsystems. This allows one to study how the cyber-physical system behaves in the presence of a fault; in other words, the proposed approach allows for understanding and studying the impact of a fault on the overall cyber-physical system. Since it relies on state-of-the-art commercial environments originally devised to support the system design, the proposed approach is able to automate and reduce the time and effort required by FMECA. Until now, FMECA is substantially a manual process where the safety expert (with the help of designers and application engineers) identifies the effect of the different failure mechanisms that may come from faults affecting the various components; this process is performed using an inductive method. Only in the case of digital circuits, this process is automated through fault simulation [14]. In the case of complex systems, which integrate analog, digital, and even software elements, this process is hard to automate. Only recently, some first automatic approaches have been proposed [13].

The main contributions of the proposed methodology with respect to the literature are:

- We propose a method to perform FMECA, based on commercial electronic design automation (EDA) tools originally introduced for design (not for FMECA), allowing the analysis of a complex cyber-physical system composed of analog, power, digital, and mechanical subsystems. The control software executed by the microcontroller is considered, too.
- The method allows evaluating, in detail, the effects on the whole system of each single fault affecting a single subsystem: for this purpose, the subsystem is simulated at a low level of abstraction, while the stimuli to be applied to the subsystems and the effects of the fault at the system level are computed resorting to a high level of abstraction simulation; the new EDA tools allow one to easily combine low- and high-level models and to effectively perform their combined simulation.
- As a result, the FMECA process can be effectively automated and speeded up, supporting a critical step in today's design flow of many systems.
- Experimental results gathered on a case of study show that by using the proposed approach, not only the safety engineer can more easily identify the critical faults affecting the system, but their number is significantly reduced, mainly due to the masking effects of the dynamic control system often implemented by the software.

With respect to the works proposed in [3,6–11,13], the approach proposed in this paper for performing the FMECA analysis is more powerful and complete. For example, in [3], the FMECA is performed for a single analog subsystem by injecting the faults at low level, only short-circuiting some components or introducing open circuits in the subsystem. Moreover, the effect of a low-level fault is not propagated to the other subsystems present in the cyber-physical system. Instead, in [6–11], the effect of a fault is propagated to the other subsystems; however, FMECA is performed at high level, modifying the subsystem features: this does not necessarily model their exact behavior in the presence of a fault. Moreover, in [6–11], the high-level faults considered are injected by changing the behavioral input–output relationships of a subsystem; instead, in the approach proposed by us, faults are considered at the level of the circuit diagram or inside a device. Finally, in [13], each fault is again considered at a high level, but the simulator is also able to simulate the behavior of the control software; this aspect is fundamental for analyzing the fault mitigation ability of the control system.

In our work, the low-level fault injection system is similar to the one proposed in [3], while the system-level classifier is similar to the one proposed in [13]. Moreover, in [13], the assessment of the

failure effects is performed at the system level (in the specific case applied to the entire vehicle dynamics). The methodologies of both [3] and [13] can assess the embedded software effects. This capability has been kept also in our approach.

The proposed methodology has been applied on a three-phase motor control system used in industrial applications, such as industrial compressors and forced ventilation systems. In particular, the SSUT considered is the power supply unit (PSU) subsystem used to power the three-phase inverter. In the SSUT, catastrophic faults are considered as affecting the power devices assembled in the PCB. In particular, the fault list is generated in accordance with the PCOLA/SOQ [15] standard. This standard considers the possible short circuits and open circuits present between the devices of the SSUT considered; moreover, faults internal to a power device are also considered. Thanks to the enhanced capabilities of the FMECA environment proposed in this paper, we were able to prove that only a limited subset of the faults that may arise in the electrical and electronic (E/E) subsystems are really critical from the point of view of the whole system safety. Using the proposed approach, the designer and safety experts can focus on the really critical faults when devising efficient in-field and end-of-the-manufacturing test strategies.

In this paper, a power subsystem is considered as a case of study; however, the proposed approach is general and can be adopted for the FMECA of any type of complex cyber-physical system.

The paper is organized in different sections. Section 2 provides the reader with some background about E/E systems modeling and the analog fault models considered. Furthermore, the FMECA practices are discussed. Section 3 describes the proposed approach, and Section 4 outlines the case study we considered. Section 5 shows the experimental results we obtained. Finally, Section 6 draws some conclusions.

2. Background

In this section, the concepts reported are related to the analog power subsystems considered in this paper. Initially, some concepts used for modeling the system at low level and high level are reported. Afterwards, the analog fault models typically adopted in the analog circuits are discussed. Finally, in the last subsection, some concepts related to FMECA are discussed.

2.1. Behavioral and Structural Models of Power Electronic

In general, the E/E systems are composed of different dedicated subsystems. Each subsystem performs a specific task; a subsystem receives in input some electrical quantities and produces other quantities in output. The different subsystems are interconnected creating a high-level block diagram of the overall system [16–18]. In the high-level block diagram, the outputs of each subsystem are connected to the inputs of other subsystems. For each subsystem, it is possible to identify a high-level behavioral model [16]; this model is characterized by a set of equations that describe the relationships between the inputs and outputs of the subsystem. For example, an amplifier receives an input of a voltage signal that varies over time. The amplifier produces in output a new signal proportional to the input one. The gain of the amplifier describes the proportion between the input and the output signals. In a simulator, it is possible to perform a simulation of the overall system by inserting the behavioral model of each subsystem present in the whole system.

Afterwards, each subsystem must be implemented. For the E/E systems, each subsystem is implemented with an electronic circuit. Therefore, a circuit diagram of each subsystem is produced. The circuit diagram is the structural model of an electrical subsystem, it is composed of different electrical components commercially available. The circuit diagram represents a new subsystem model; in particular, it represents a low-level implementation model. For example, an amplifier can be modeled at the behavioral model with the relation Vout = Vin·G; at the circuit diagram level, the amplifier is composed of numerous electrical components, e.g., transistors and resistors, in order to obtain a circuit that implements the relationship Vout = Vin·G. This circuit diagram represents a possible low-level model of the amplifier.

During the design of the overall system, the development of a high-level block diagram is a step normally performed, in particular, for a system composed of different subsystems. Therefore, the overall block diagram of the whole system is usually available and well defined already in the early phase of the complex system design.

In this paper, the two kinds of models are used: high-level behavioral models and low-level structural models; the second one is a possible implementation of the subsystem. For the power subsystem, the circuit diagram model is considered. Obviously, in a circuit simulator, it is possible to insert the circuit diagram of each subsystem and simulate the overall system at low level. However, this simulation strategy is not recommended due to the high simulation times required. Usually, each subsystem is simulated separately with the circuit diagram, whereas the simulations of the overall system are performed at high level using the behavioral models of each subsystem.

2.2. Multilevel Simulation

The multilevel simulation is a practice commonly adopted for performing simulations of systems composed of different subsystems. The idea of multilevel simulation is to combine low-level and high-level models in one simulator. In multilevel simulations, only one subsystem is simulated at low level, resorting to the structural model of the subsystem; the remaining subsystems are simulated at high level, resorting to the behavioral models. Moreover, the multilevel simulation strategy allows for performing simulations of complex mixed-domain systems, i.e., systems that involving low-voltage subsystems, high-voltage power subsystems, digital subsystems or microcontrollers, mechanical subsystems, and so on. In addition, the embedded software executed by the microcontrollers is simulated, too.

Different multilevel simulation solutions are proposed in different papers [19–24]. For example, a multilevel simulation strategy oriented to the mixed-signals integrated circuit design is proposed in [19]. In particular, the different problems relating to the interfacing of the different domains are discussed in [19]. In [20], a multilevel simulator for a mechatronic system is proposed; the simulator discussed in [20] is used to simulate the control system of an electric motor. Instead, a power inverter used to drive a DC motor for an electrical vehicle is simulated in [22]. The multilevel simulator proposed is built with PSIM [25] and MATLAB/SIMULINK [26] tools. Finally, in [23,24], a multilevel simulation of a mono-domain system is proposed. In particular, the systems proposed in [23,24] are composed only of electrical subsystems. In [23], the Ana1og-circuits Multilevel SIMulation (AMSIM) is proposed. As discussed in [23], the advantages of the AMSIM simulation strategy used in the design phase of the system are discussed.

In this paper, we propose the use of multilevel simulation for performing a FMECA analysis focusing on the faults affecting a subsystem including some power devices.

2.3. Analog Fault Models

With the introduction of some new analog fault simulators, such as DefectSim [27] by Mentor Graphics and TestMAX [28] by Synopsys, the catastrophic fault [29–31] model became widely used. Catastrophic faults consist of open circuits or short circuits in the circuit diagram of a subsystem. Typically, these faults are modeled in the circuit diagram by introducing some electrical switches. A further switch is added in series to each electrical component present in the circuit diagram to model an open circuit fault. Instead, the short circuit fault is modeled by inserting an additional switch in parallel to each component. In addition, it is also possible to introduce further catastrophic faults related to the network topology. In other words, some switches are inserted between nodes that are normally unconnected. Therefore, the topological faults introduce a further possible short circuit in the subsystem circuit diagram.

In the analog circuits, another possible fault model considered is the parametric fault [29–31]. Parametric faults are variations of one parameter of a component outside its nominal range. For example, in a resistor with a nominal value of resistance and a certain tolerance value in respect to its nominal

value, a parametric fault is modeled by inserting a resistance whose value is outside the range defined by its tolerance.

Moreover, it is possible to consider only a single fault at a time or different faults at the same time [29–31]; in the first case, a single point fault scenario is considered, while in the second one, a multiple point faults scenario is considered. In this work, a single point scenario with different catastrophic faults is considered.

2.4. FMECA

For those E/E items in charge to perform safety- or mission-relevant operations, it is needed to evaluate their reliability level. Usually, this parameter is expressed through metrics representing how much time they can operate safely, i.e., without any safety goal violation [32]. Usually, no discrete component is capable of ensuring such a level of reliability by itself; therefore, the effort is shifted to the architecture for adopting solutions as redundancy, monitoring, and so on.

An item can react to a critical failure in two different ways. The simpler one is to bring the system into a safe state. Such a system is defined as fail-safe. The other, smarter but more expensive with respect to the previous, is to continue to provide the function even in the case a failure happens. In this second case, the system is fail-operational. If the function is provided with a lower but yet effective quality, we have a graceful degradation approach.

In the usual design process, the first step is to identify the potential failures that can affect the considered item. There are various manuals to be followed during this phase, and the most promising is the one jointly released by AIAG &VDA in June 2019 [33]. By following it, each failure is classified by an action priority (AP). It can assume only three values: high, medium, and low. A first AP value is assigned to the system by itself, then is updated taking into account the possible detection and/or mitigation measures that it is possible to apply. At this point, the requirements determined during the failure mode and effect analysis (FMEA) and the risk level associated with the item functionality are combined to obtain the requirements. After the item has been designed, FMECA has to be performed on it. The FMECA [34] is usually performed for aerospace application and the expected result is a risk priority number (RPN) for each one of the failure modes of the item components. RPN is defined as the product between the severity, the occurrence, and the detection capability embedded in the item. On the other hand, the FMECA [32] is performed in the automotive environment and classifies all the failure modes into four groups as a combination of Safe/Dangerous and Detected/Undetected. The FMECA process is essentially a manual process, where the designer identifies different failure mechanisms and studies their behavior. To support FMECA execution, a simulation-based approach has been proposed in [35], where a methodology based on a simulation framework is used that employs behavioral models. When evaluating system outputs in the presence of faults, the Safe/Dangerous–Detected/Undetected classification is highly dependent on the specific application. For the sake of this work, we rather classify faults as critical or non-critical [36] (in terms of divergence with respect to the design requirements) since we want to obtain the maximum level of generality. In other words, the proposed classification is independent of the application, because the proposed classification considers how much the system affected by a fault deviates from its nominal behavior present in the fault-free scenario.

For historical reasons, the FMECA process is based on the assumption that all or most of the components of the circuit are discrete (like resistors, capacitors, and diodes) and that there are not too many. This was often true in the past, while today many devices are modular and may also correspond to complex integrated circuits or commercial off-the-shelf (COTS) submodules.

As a result, FMECA poses four types of challenges:

(1) The time required for simulating the whole system at low level (e.g., with SPICE) is completely unacceptable; simulating different parts of the system at different abstraction levels is a feasible solution, but implies the availability of an environment where models of the different modules

can be easily integrated, where the simulation at different levels is supported and where signals flow from one module to the other even when they are described at different levels;

(2) The circuit diagram of the COTS components at the different levels (including the most detailed ones) is not always available, so multilevel simulation is not always possible;

(3) The failure patterns of digital electronics are different from those of analog ones; hence, the choice of the most representative and suitable fault model is not given;

(4) The effect of embedded software must be considered, too.

In the method proposed in this paper, we adopt state-of-the-art EDA tools which in principle allow dealing with every E/E system. In particular, for the case of study demonstration, we chose two different commercial tools, one for the low-level simulations, and one for the high-level simulations. Both these possibilities allow executing the microcontroller software code, too, making it possible to assess the effect of the embedded software. These tools are also frequently adopted during the design phase in most industrial environments. Our approach can be applied also in the other two phases of the development: during the concept phase, with only high-level models, and during the validation of the item. In this way, we can set up an iterative design approach, where the item is redesigned and tested over and over again until the safety requirements are met.

3. Proposed Approach

The idea of the proposed approach is to combine high-level and low-level models of the different subsystems present in a complex system to obtain a multilevel simulator. The multilevel simulator is then used to perform the FMECA study of a complex system affected by a fault, as discussed in Section 2.4. The proposed approach is useful for identifying the critical faults, i.e., the faults that modify the behavior of the complex system obtaining behaviors different from the expected one. In other words, the complex system assumes an unwanted behavior that does not conform to its design specifications. In the proposed approach, the effects of the fault are observed only in some accessible points of the complex system. In particular, the electrical signals present on the output ports of the complex system (such as voltages or currents), or the physical quantities handled by mechanical actuators (such as the angular speed of a motor shaft) are considered. In the presence of a critical fault, the quantities considered are not complying with the design specification expected values. In the approach, different low-level faults are considered in one of the subsystems present in the complex system. The faults are generated according to a chosen fault model. The proposed approach is shown for a generic SSUT of the complex system; it is performed in eight steps, illustrated below, and each step is shown in the diagram of Figure 2.

1. Block diagram generation. The block diagram of the overall complex system is obtained. Usually, this block diagram is defined during the first phase of the system design. The block diagram shows the name of each subsystem and the connections between the different subsystems, as discussed in Section 2.1.
2. Preparation of the behavioral models for the subsystems. In this step, the behavioral model of each subsystem present in the whole complex system is prepared. It can be obtained from the design phase of the complex system, or by identifying the transfer function between the inputs and the outputs of the considered subsystem. Therefore, the relationships between the inputs and outputs of each subsystem are explained as discussed in Section 2.1. The behavioral model of each subsystem is inserted in the block diagram of the whole system identified in the previous step.
3. High-level system simulation. With the high-level models of each system now built, it is possible to perform a first functional simulation of the overall complex system at high level. In other words, it is possible to apply some external stimuli and to verify the stimulus response of the complex system. The stimuli applied must comply with the system design specifications, and the stimulus response provided by the complex systems must comply with the complex system

design specifications. Generally in a cyber-physical system, the stimuli applied to the system are electrical, while the response to the stimulus is obtained on the mechanical actuator.

4. Subsystem under test. The subsystem in which the faults are injected is now chosen. The SSUT is replaced in the block diagram with its low-level implementation, i.e., with its low-level structural model.
5. Model check. Now, it is possible to perform a new functional simulation of the overall system. The purpose of this new simulation is to check again the system response to the stimuli applied to the complex system. The response to the stimuli must comply with the system design specifications. The stimulus response trend obtained is called the gold response, and it is obtained in a fault-free scenario. The gold response complies with the complex system design specification, too.
6. Faults list generation. In this step, the list of the considered faults is generated. The fault list is obtained in accordance with the SSUT fault model chosen. In the literature for each fault model, there is an algorithm able to generate the list of the possible faults.
7. Fault effect simulation. For each fault considered, a functional simulation is performed by applying a stimulus to the complex system. A functional stimulus is an input signal that complies with the system design specifications. The saboteur injects a single catastrophic fault [37] into the SSUT structural model at the start of a simulation, as discussed in [3].
8. Fault effect evaluation. A classifier [13,35] compares the stimulus response obtained from the complex systems with the gold response previously obtained in the fault-free scenario. The injected fault is considered critical if the stimulus response is not compliant with the design specification, in other words, coherently with the definition of fault contained in the FMECA manuals [33,34], if the fault effect produces a difference in respect to the item design requirements. In particular, during the system design phase, different maximum tolerance values are established for each electrical quantity present in the complex system. The fault is classified as critical if the value obtained in the simulation exceeds the maximum accepted tolerance.

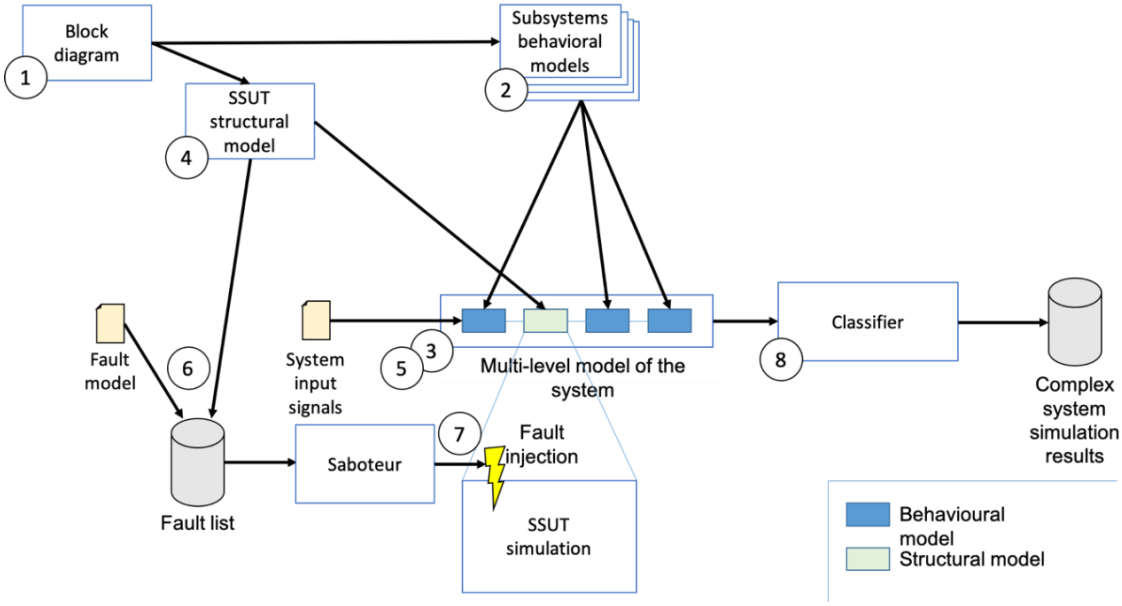

Figure 2. The proposed approach diagram.

The approach proposed can be applied to all the different SSUTs present in the complex system. However, for the purpose of this paper, the proposed approach is used on an E/E power subsystem. As discussed in Section 2, the structural model for the E/E subsystems is the circuit diagram. In addition,

as discussed in Section 2.3 and in [38,39], in this paper, we consider only the catastrophic faults model applied at the SSUT circuit diagram level or inside a power device. The catastrophic faults are modeled by adding different electrical switches in the circuit diagram of the SSUT. In particular, the serial switches, the parallel switches, and the topological switches are added with some rules now discussed [37]. The fault list is generated by introducing:

- A switch in series to each device present in the circuit diagram;
- A switch in parallel to each device present in the circuit diagram;
- Different short circuits switches between different nodes of the circuit diagram, in particular, between nodes that are normally unconnected in the SSUT circuit diagram.

The input stimuli are electrical quantity, as voltage, applied to the printed circuit board (PCB) input ports. The stimulus response is the PCB electrical response obtained on the PCB output ports, as voltage or current. Moreover, the stimulus response can be observed also by the mechanical actuator connected to the PCB output port, e.g., the angular speed of the electric motor connected to the PCB output ports.

With the proposed approach, it is possible to evaluate the effect of the faults through a multilevel simulation environment, as discussed in Section 2.1. As discussed in Section 2.4, for the FMECA analyses, it is necessary to evaluate the impact of the faults on the whole complex system. In the proposed strategy, only one SSUT is considered at a time. This approach offers a good compromise between the simulation time and the study of the low-level fault. The multilevel simulation allows a fast simulation of the whole complex system that includes the low-level fault injected in the structural model of the SSUT.

4. Case Study

This section describes the three-phase motor control system considered as a case study. This complex system can be used in different industrial applications as industrial compressors and forced ventilation systems. Besides, this complex system can be used in different household appliances that require motors for working, too. The case study is particularly relevant as some of these applications are safety-critical, while others are mission-critical. In both cases, being able to accurately and automatically estimate the failure mode effects is of paramount importance. Initially, the overall cyber-physical system is described, then the different subsystems that compose it are described. Finally, the power supply unit (PSU) subsystem, chosen as the SSUT, is described.

4.1. The Motor Control System Overview

The system considered is used for managing a 2.2 kW three-phase electrical motor. The motor considered operates at 400 V phase–phase with two polar pairs active simultaneously, and with a current of about 6 A for phase. The motor considered has an angular speed of about 3000 RPM. The block diagram of the whole system is shown in Figure 3. The whole complex system is implemented on a single PCB, as shown in Figure 4. Moreover, Figure 4 shows the circuits of the different subsystems, too.

The system considered implements a speed and current control for the three-phase motors. The PCB is composed of nine subsystems connected together, as shown in Figure 3. In the PCB, there is a high-voltage PSU; the purpose of the PSU subsystem is to provide a direct current (DC) high voltage useful for powering the power electronics present on the PCB. A second low-voltage PSU is present on the PCB. This second PSU is used to provide additional DC low voltages used for powering the different low-voltage subsystems present in the PCB. The high-voltage PSU accepts an input and electrical grid voltage of 110 or 250 V RMS, with a frequency of 50 or 60 Hz. Moreover, the PSU is equipped with an electromagnetic compatibility (EMC) filter. The EMI filter consists of a common mode choke and film capacitor used to reduce the conduction electromagnetic emission caused by the PSU switching.

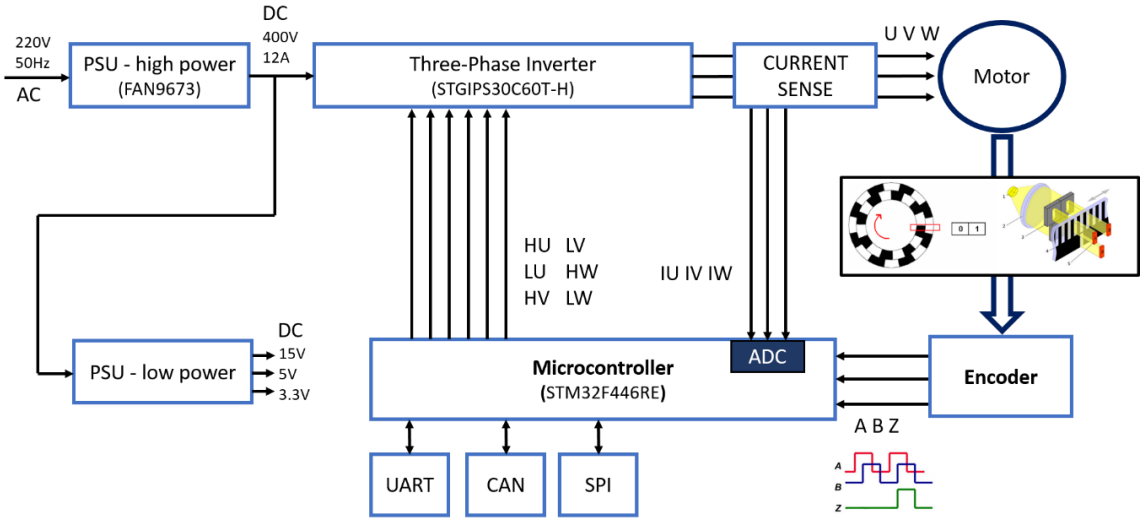

Figure 3. Block diagram of the whole electrical/electronic (E/E) system.

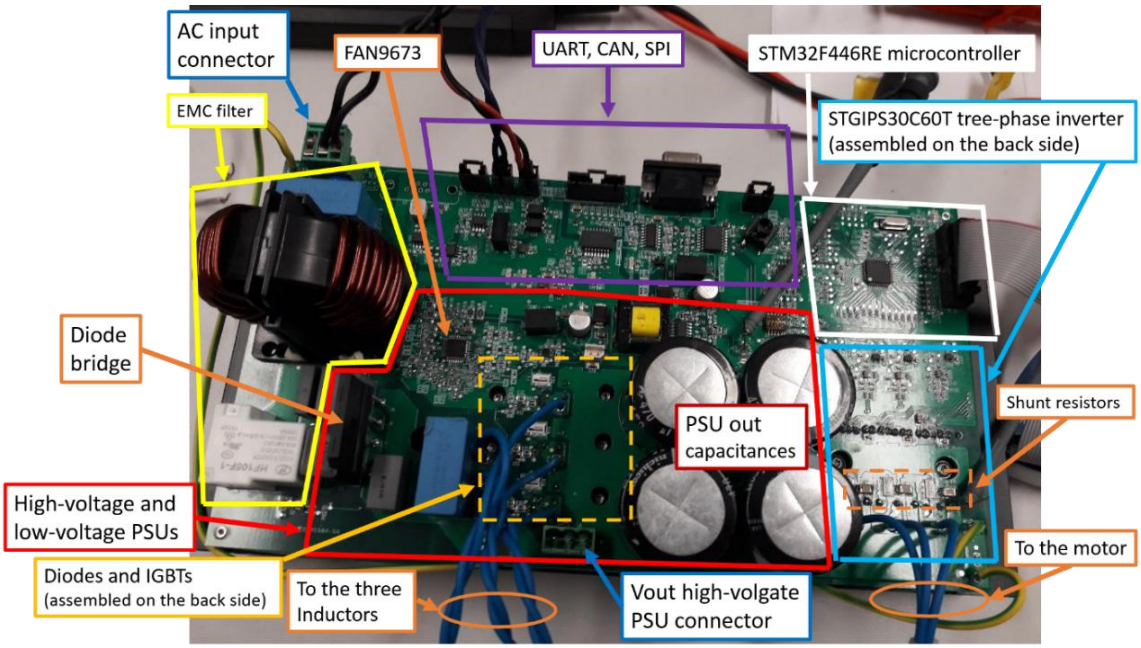

Figure 4. The different subsystems on the printed circuit board (PCB).

4.1.1. The Three-Phase Inverter Subsystem

The subsystem that implements the three-phase inverter is designed around the STGIPS30C60T [40] integrated device produced by STMicroelectronics. This power device is internally composed of three half H-bridges. Each half H-bridge is used to drive one of the three phases of the motor. Internally to the STGIPS30C60T integrated device, there are 6 power insulated gate bipolar transistors (IGBTs) that implement the three half H-bridges; furthermore, the low-voltage electronics used to drive the IGBTs are present inside of the STGIPS30C60T device. The digital input signals accepted by the integrated circuit are compliant with the transistor–transistor logic (TTL) at 5 V, or with the complementary metal-oxide semiconductor (CMOS) logic at 3.3 V. Instead, the power IGBTs can operate up to 600 V with currents of 9 A for each half H-bridge.

4.1.2. The Current Sense Subsystem

The current absorbed by the motor is measured with a dedicated subsystem. Three shunt resistors are used to measure the current present in each phase of the motor. The voltage drop on each shunt resistor is measured by an instrumentation amplifier. The voltage drop measured on the shunt resistor is converted by one of the analog to digital converter (ADC) units integrated into the microcontroller.

4.1.3. The Encoder Subsystem

An encoder is present on the motor shaft, and it is used to measure the angular speed of the motor. An apposite subsystem is present on the PCB for managing the encoder and for interfacing the encoder with the microcontroller.

4.1.4. The Microcontroller Subsystem

The microcontroller subsystem executes the motor control software. In particular, the microcontroller receives in input the values of the currents measured on the three phases of the motor and the motor angular speed. Considering the different input data received, the microcontroller processes three pulse-width modulation (PWM) signals, one for each phase of the motor. The PWM signals are applied to the three-phase inverter subsystem. The aim of the software control implemented in the microcontroller is maintaining constant the angular speed of the motor. Furthermore, the microcontroller can introduce a second control system relating to the current absorbed by the motor. This second automatic current control can be excluded if not necessary. The STM32F446RE [41] is a 32-bit microcontroller developed by STMicroelectronics. This microcontroller is based on an ARM CORTEX M4 core and it operates at 180 MHz. The STM32F446RE is used for real-time control applications. It is equipped with an adaptive real-time accelerator (ART Accelerator) [42] used to speed up the reading and writing operations performed on RAM and flash memories. The chosen microcontroller is able to run the control software with the correct processing times required by this real-time system. Moreover, the microcontroller is equipped with a sufficient number of peripherals necessary to perform the different tasks required. In particular, 4 different timers are used in addition to the ADC. Furthermore, different communication peripherals are used, such as the universal asynchronous receiver–transmitter (UART), the controller area network-bus (CAN-bus), and the serial peripheral interface (SPI). Moreover, the microcontroller is equipped with 512 KB of flash memory sufficient to contain the control software implemented and 256 KB of RAM memory.

4.1.5. The Communication Subsystems

Finally, in the PCB, there are three distinct subsystems dedicated to the communication. Through one of the three possible interfaces (UART, CAN, or SPI), it is possible to communicate with the microcontroller for obtaining the motor angular speed measured or the current values present in each phase of the motor. Furthermore, through the communication interfaces, it is possible to modify the angular speed value that the control system must maintain. By default, this value is set to 3000 RPM.

4.1.6. The Power Supply Unit Subsystems

In this subsection, the high-voltage PSU is discussed. In particular, its low-level implementation is shown; in other words, the circuit diagram of the PSU and the role of each electrical component is analyzed. The circuit diagram of the high-voltage PSU is shown in Figure 5.

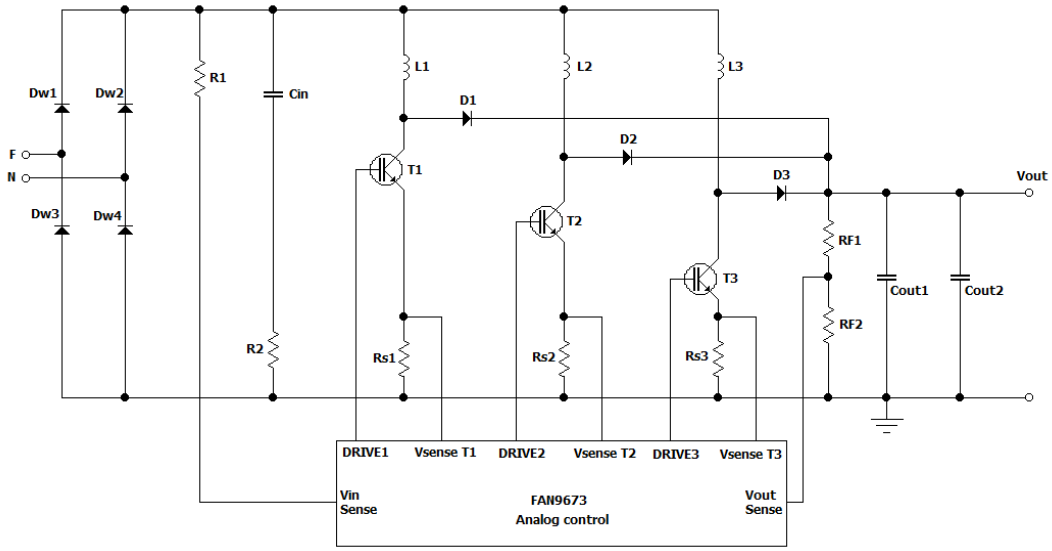

Figure 5. The circuit diagram of the high-voltage power supply unit (PSU) subsystem.

The considered PSU is used to supply the high voltage needed for the electrical motor. In particular, this PSU supplies an output DC voltage constant of 400 V with a maximum ripple of ±7 V. The maximum current delivered by the PSU is 12 A. The PSU accepts in input an AC voltage of 110 or 250 V RMS, with a frequency of 50 or 60 Hz. The PSU is composed of a diode bridge (Dw1, Dw2, Dw3, Dw4), three boost cells, and an analog controller. Each boost cell is composed of an inductor, a diode, and an IGBT. Moreover, two capacitances (Cout1, Cout2) are placed at the PSU output. During the design of the PSU, the STTH12S06 [43] and STGF19NC60 [44] devices were chosen, respectively, for the diode and IGBT of the boost cells. Moreover, the analog controller chosen is the FAN9673 [45] developed by the ON Semiconductor. The FAN9673 controller measures the current on the three IGBTs in differential mode through the shunt resistances (Rs1, Rs2, Rs3). In addition, the input voltage on the CIN capacitor and the output voltage of the PSU are measured by the FAN9673 controller. The output voltage is measured resorting to a voltage divider (RF1, RF2). The FAN9673 produces three signals (DRIVE1, DRIVE2, DRIVE3) that are applied to the IGBTs (T1, T2, T3). The aim of the control system is to obtain a sinusoidal shape of the current absorbed from the electrical grid and with a power factor almost unitary. Each boost cell is controlled by the FAN9673 for operating in continuous conduction mode (CCM). An independent control signal is produced for each IGBT of each boost cell. The signal controls are a square wave with a frequency of 60 kHz and a variable duty cycle.

5. Experimental Results

This section reports the experimental results obtained with the approach discussed in Section 3 applied to the case study described in Section 4. For the purpose of this paper, an E/E power subsystem is chosen as the SSUT. The proposed method can be repeated for all the subsystems present in the case study complex system. In particular, in this section, the different catastrophic faults considered in the high-voltage PSU subsystem are discussed. Afterwards, the effect of each fault on the whole system is shown.

5.1. The PSU Fault Considered

Different possible faults present in the circuit diagram of the high-voltage PSU are considered [46]. In particular, some faults are considered among the three components that compose one of the three boost cells of the PSU. The faults considered are defined in accordance with the PCOLA/SOQ metric [15]. Out of the different categories listed by the standard, Correctness, Orientation, Alignment, and Quality are not of interest for the sake of this work since they regard circuit manufacturing defects, while we

are interested only in those faults that can happen during the item operations. Only the Open and Short faults are considered between the devices, considering that Presence is equal to an open circuit in a simulation. However, the PCOLA/SOQ metric is not particularly efficient for the defects inside the device; only the Live attribute of the PCOLA metric provides general information about the device working. Therefore, some possible faults present inside the IGBT device are considered, as discussed in [38,39]. The next two subsections discuss the faults considered in the high-voltage PSU and inside the IGBT power device.

5.1.1. Boost Cell Faults

Nine different catastrophic faults are considered in a single boost cell of the PSU. The three boost cells present in the PSU are equivalent and placed in parallel; therefore, it is possible to study the effect of the faults in one of the three boost cells indistinctly. The boost cell considered is composed of D1, T1, and L1 components. Nine electrical switches are placed between the diode, the inductance, and the IGBT device of the boost cell, as discussed in Section 3. The faults considered are shown in Figure 6.

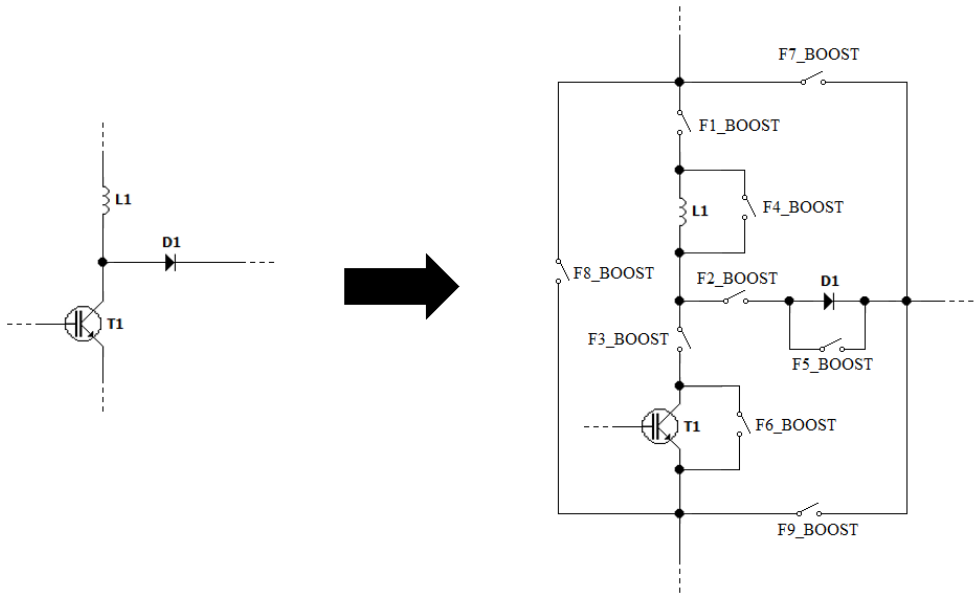

Figure 6. The faults considered in the boost cell.

The faults F1_BOOST, F2_BOOST, and F3_BOOST are placed in series to the diode, the IGBT, and the inductance; these three faults introduce three open circuits in the PSU boost cell. The faults F4_BOOST, F5_BOOST, and F6_BOOST are placed in parallel to each component, these three faults introduce a short circuit for each component. Finally, faults F7_BOOST, F8_BOOST, and F9_BOOST introduce further possible short circuits between the boost cell nodes normally unconnected.

5.1.2. IGBT Power Device Faults

Further faults can be considered inside the SSUT devices, too. For each device of the SSUT, it is possible to derive an equivalent electrical model of the device, as discussed in [38]. In the equivalent electrical model of the device, it is possible to consider some faults. In this paper, we consider the faults of the IGBT device. In [38], an equivalent electrical model of the IGBT is derived and different catastrophic faults are considered. Figure 7 shows the equivalent electrical model of the IGBT with 23 switches that model 23 catastrophic faults. The meaning of each IGBT catastrophic fault is discussed in [38,39].

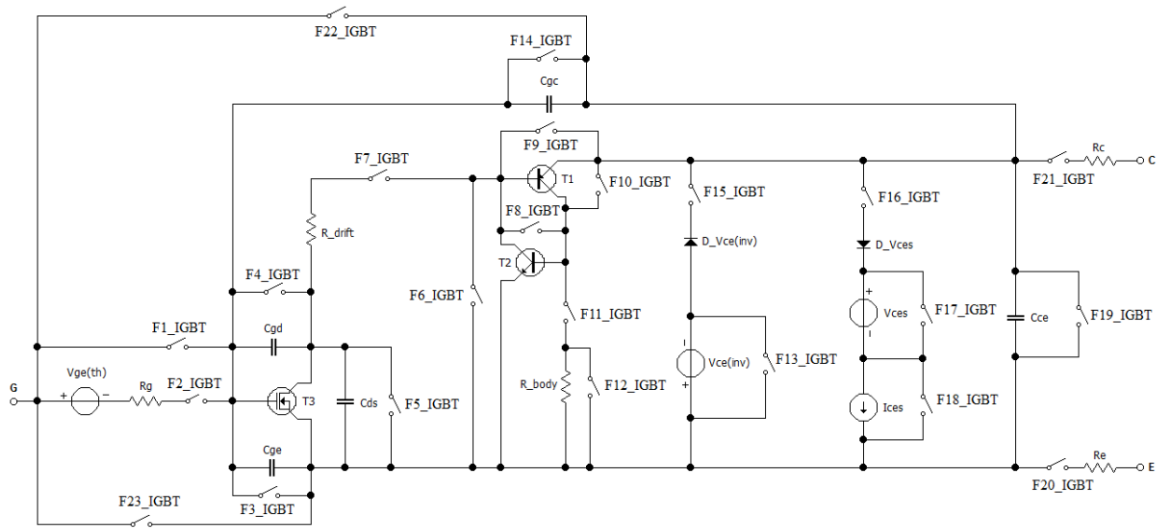

Figure 7. The faults considered inside the insulated gate bipolar transistor (IGBT) device.

5.2. The FMECA Results

In this subsection, the results of the simulations are reported. In particular, the voltages present on the three phases of the motor, the currents present in each motor phase, and the angular speed measured on the motor shaft are considered. These electrical quantities are measured on the output ports of the PCB, while the angular speed of the motor shaft is measured with an additional tachometer placed on the motor shaft itself. Moreover, the voltage supplied by the high-voltage PSU to the whole system is considered. In the case study considered, the PSU Vout connector can be used for measuring the high-voltage PSU, as shown in Figure 4.

During the system design, in addition to the nominal values of each considered quantity, the maximum accepted tolerances are also defined. Table 1 reports the nominal values and the associated tolerances for each quantity considered. These tolerances are defined by the system designer.

Table 1. The nominal values and tolerances defined in the system design phase (* defined by the complex system designer).

	Nominal Value	Tolerance *	Tolerance Range
U, V, W voltage	400 V	1%	396–404 V
U, V, W current	6 A	2%	5.88–6.12 A
Angular speed	3000 RPM	5%	2850–3150 RPM
Vout high-voltage PSU	400 V	1%	396–404 V

Table 2 shows the results obtained in the fault-free scenario and for each fault considered. All the measures are performed with the system in a steady state. All experiments are performed by applying a voltage of 250 V RMS at 50 Hz to the system input port. The injected fault is classified as critical if one of the quantities considered exceeds the tolerance range defined in Table 1.

In Table 2, it is possible to identify 6 faults classified as critical and 26 faults classified as non-critical. The impact of the six faults classified as critical on the overall complex system is particularly significant. A mitigation strategy must be implemented to detect these critical faults. Therefore, it is necessary to identify a test strategy to verify the presence of critical faults. In general, the effects of the non-critical faults are compensated by the FAN9673 analog controller, as discussed in [39]. In other words, the PSU control system acts on the IGBTs for compensating the effect of the injected fault.

Table 2. Critical fault results.

Faults	U, V, W Voltage [V]	U, V, W Current [A]	Angular Speed [RPM]	Vout High-Voltage PSU [V]	Critical
Fault-free	402	6.08	2979	400 V with ±7 V of ripple	-
F1_BOOST	402	5.98	2979	400 V with ±7 V of ripple	NO
F2_BOOST	263	4.26	1222	265 V with ±25 V of ripple	YES
F3_BOOST	402	5.98	2979	400 V with ±7 V of ripple	NO
F4_BOOST	402	5.98	2979	400 V with ±7 V of ripple	NO
F5_BOOST	377	7.90	1718	Vout instable	YES
F6_BOOST	398	5.89	2866	397V with ±10 V of ripple	NO
F7_BOOST	398	5.89	2866	397 V with ±10 V of ripple	NO
F8_BOOST	0	0	0	0 V	YES
F9_BOOST	0	0	0	0 V	YES
F1_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F2_IGBT	398	5.89	2866	397 V with ±10 V of ripple	NO
F3_IGBT	398	5.89	2866	397 V with ±10 V of ripple	NO
F4_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F5_IGBT	398	5.89	2866	397 V with ±10 V of ripple	NO
F6_IGBT	398	5.89	2866	397 V with ±10 V of ripple	NO
F7_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F8_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F9_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F10_IGBT	398	5.89	2866	397 V with ±10 V of ripple	NO
F11_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F12_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F13_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F14_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F15_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F16_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F17_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F18_IGBT	402	5.98	2979	400 V with ±7 V of ripple	NO
F19_IGBT	398	5.89	2866	397 V with ±10 V of ripple	NO
F20_IGBT	398	5.87	2866	397 V with ±10 V of ripple	NO
F21_IGBT	398	5.93	2866	399 V with ±8 V of ripple	NO
F22_IGBT	312	4.90	1585	300 V with ±20 V of ripple	YES
F23_IGBT	312	4.90	1585	300 V with ±20 V of ripple	YES

For the sake of completeness, the simulation results obtained in the fault-free scenario and injecting the F5_BOOST critical fault are shown. As shown in Figure 6, this fault short-circuits the diode present in the boost cell. Figure 8 shows the fault-free scenario output trends; in the fault-free scenario, the outputs of the system have the expected behavior compliant with the design specifications. In particular, Figure 8a shows the angular speed of the motor. Figure 8b shows the V (red), W (blue) and

U (green) voltages applied to the motor. Finally, Figure 8c shows the V (red), W (blue) and U (green) currents present in each motor phase.

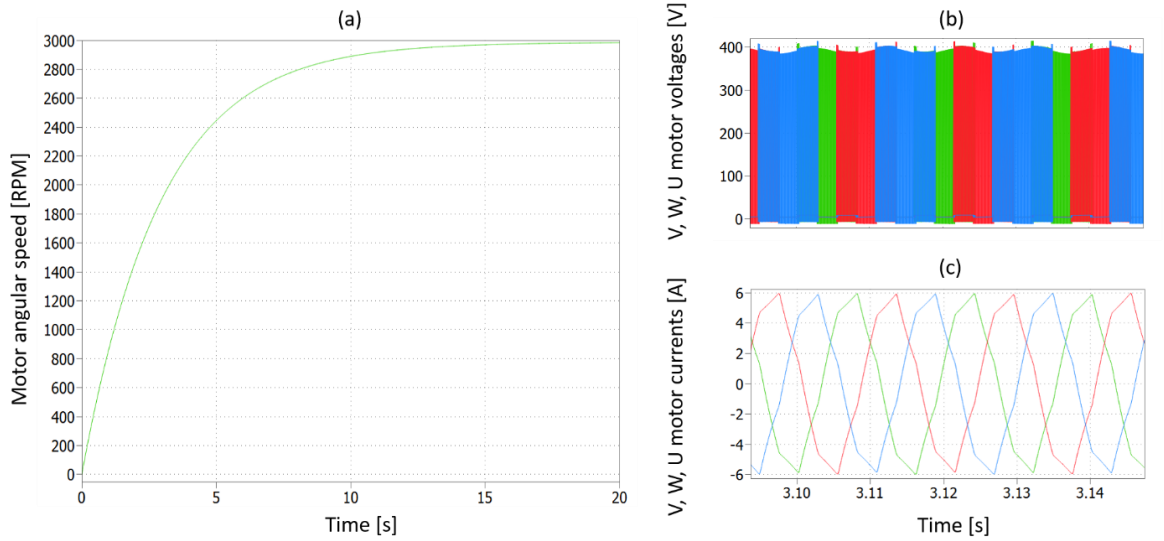

Figure 8. Free-fault simulation results.

Figure 9 shows the behavior of the system affected by the F5_BOOST critical fault. In particular, it is possible to see the unstable behavior of the high-voltage PSU, in Figure 9a. Due to this fault, the PSU voltage supplied to the electric motor is not stable over time. The motor is periodically switched off and the correct trend in the three phases of the motor does not assume the sinusoidal trend expected. The angular speed control system is unable to bring the motor to the required speed. Figure 9b shows an average angular speed of about 1700 RMP with continuous dangerous accelerations and decelerations. Figure 9c shows the V (red), W (blue) and U (green) voltages applied to the motor due to the injected fault. In addition, Figure 9d shows the V (red), W (blue) and U (green) currents present in each motor phase. The trend of the currents shown in Figure 9d is due to the injected fault.

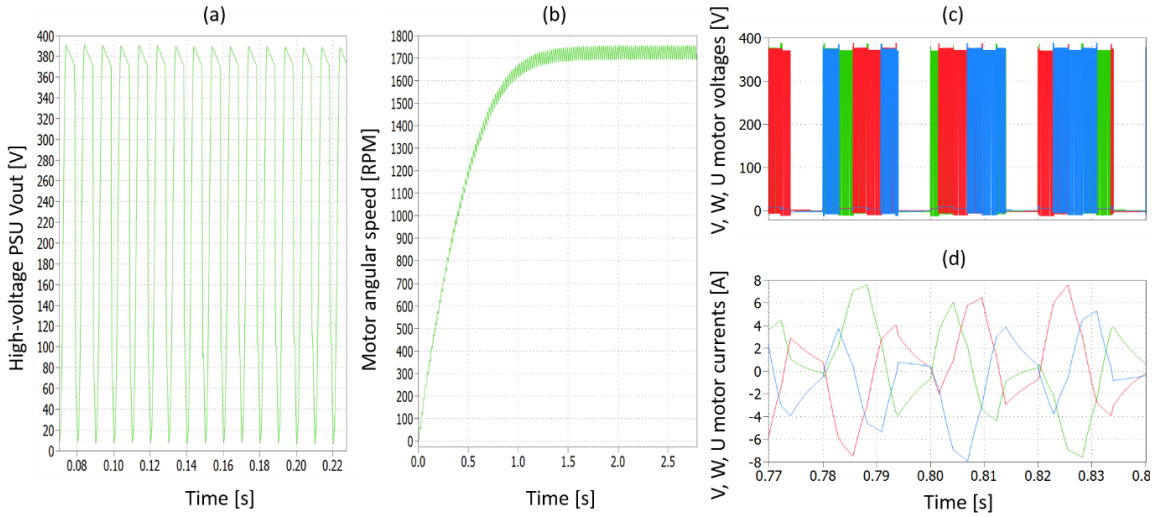

Figure 9. Fault F5_BOOST simulation results.

Finally, we provide the reader with a comparison between the results we obtained and those from the methods proposed in [3,6–11,13].

In particular, the approach discussed in [3] does not consider the whole cyber-physical system but only a single SSUT. Hence, the considered faults are considered critical if the stimulus response is not compliant with the high-voltage PSU design specification. To show the advantage of the method proposed in this paper, we applied the approach proposed in [3] to our test case: in this way, we identified 10 critical faults. By then applying to them our method, we could discover that four of these faults do not produce critical failures on the outputs of the whole system, because the control system implemented by the microcontroller subsystem is able to compensate the fault effect.

On the other hand, the approaches proposed in [6–11,13] are not directly comparable with the one proposed in this paper, since they consider faults at the behavioral level only.

5.3. Environment Setup

The experiments were performed with the PLECS circuit simulator and Mathworks Simulink environment, as discussed in Section 3. PLECS is a simulator specifically designed for simulating power circuits, analog circuits, and mechanical actuators. Moreover, PLECS allows the execution of C code through a particular functional block, called "C-Scripts" [47]. Using the "C-Scripts" block, it is possible to simulate the embedded software executed by the microcontroller. In the complex system, a timer integrated into the microcontroller is configured for executing the motor control software every 62.5 µs; this behavior is replicated on the simulator, too. Every 62.5 µs, PLECS interrupts the electrical simulation and executes the control software executed by the microcontroller. After the control software has been executed, the outputs of the microcontroller are updated and the PLECS electrical stimulation is resumed. The period of 62.5 µs was chosen by the control software designer.

The simulations have been performed on a PC equipped with an AMD FX-8370 8 cores processor operating at 4 GHz and 32 GB of RAM memory with a frequency of 1333 MHz. In this paper, 32 faults are considered; a total of 9 faults are considered in the circuit diagram boost cell, and a further 23 faults are considered inside the IGBT device of the boost cell. Each simulation is performed in a single fault scenario. The simulation results of each fault are automatically processed with some MATLAB scripts in order to identify critical faults.

As far as CPU time is concerned, simulating 20 s of the whole system with all the electrical subsystems modeled at electrical low level (SPICE level) requires approximately 170 min of CPU; moreover, the simulation is performed with the microcontroller subsystem modeled at the behavioral level. Conversely, when using the proposed multilevel simulation, 30 min of CPU time is needed, approximately. This highlights the effectiveness of the multilevel simulation approach proposed in this paper that allows a performance speed-up. Considering our case study, we improved it by about six times. Regarding the possibility to perform the simulation of the microcontrollers, this is not possible at all since we do not have a schematic of these items; however, considering an open-source 32-bit microcontroller, a simulation of the RISC-V [48] microcontroller subsystem at the register transfer level (RTL) requires approximately an additional 20 min for each simulation.

From the technical point, we have implemented the proposed approach of Figure 2 with the PLECS simulator [47], which is incorporated in, and handled by, the Mathworks Simulink [26] environment. The whole simulation environment is managed with numerous MATLAB [26] scripts. Therefore, different steps of the proposed approach shown in Figure 2 are performed automatically, for example, the simulations, the fault injection, the data collection, and the data post-processing processes are automatically performed by the Mathworks Simulink environment. The behavioral models of the different subsystems and the circuit diagram of the SSUT are read from the complex system design and integrated with PLECS.

6. Conclusions

Cyber-physical complex systems are composed of numerous suitably interconnected subsystems. For each subsystem, a behavioral high-level model and/or a detailed structural low-level model may be available. In this paper, a methodology for studying the impact of the low-level faults on the overall

complex system behavior is proposed. The proposed approach is based on multilevel simulations that involve behavioral and structural models of the subsystems present in the complex system. The multilevel simulation is a good trade-off between the time required for the fault simulation and the accuracy needed to model the low-level faults considered. Moreover, the availability of commercial environments allowing the easy integration of different models allows today to resort to it without significant investments in terms of EDA tools and integration into existing design flows. The proposed methodology is particularly useful for performing an FMECA study, as required by the international safety standards for the safety-critical applications used in different areas. The proposed approach allows an automatic analysis of the effects of different faults. Moreover, the proposed methodology allows identifying the critical faults at the system level, i.e., the faults that not only modify the behavior of the system, but can also cause serious, dangerous consequences.

The proposed approach is based on high-level and low-level models managed by generic simulation environments. For each different case study, the safety engineer can quickly set up an environment able to simulate the overall cyber-physical system. In this simulator, only the SSUT is modeled at a low level, while the rest is modeled resorting to high-level (e.g., behavioral) descriptions. The approach is very general and relies on the availability of different kinds of models and of the corresponding simulators (for example, a circuit simulator for analog SSUTs, a mechanical simulator for SSUTs corresponding to mechanical components, and so on). This approach is possible using modern and versatile simulation tools, such as MATLAB. The proposed approach is generic because it is possible to simulate different types of cyber-physical systems by using or developing the appropriate low- and high-level models.

The proposed methodology was evaluated resorting to a real-life case study. In particular, it has been evaluated on a control system for a three-phase electric motor. The proposed approach is general and can be applied to any subsystem; however, for the purpose of this paper, the proposed approach is applied to a high-voltage PSU subsystem. We have considered the catastrophic faults model used in the analog and power subsystems. The faults considered in this paper are injected in the circuit diagram low-level model or inside a single power device of the SSUT, as an IGBT. The effect of the faults is observed on the PCB output ports and on the mechanical actuator driven by the PCB.

In the case study considered, only six faults were classified as critical, i.e., the impact of these faults on the cyber-physical system leads the system out of the desired operating specifications defined in the design phase. Based on these results, a mitigation strategy could be devised for reducing the impact of critical faults. This paper proposes a starting point for systematic and automatic identification of critical faults in a cyber-physical system; moreover, the multilevel simulator built is useful for evaluating the mitigation strategies introduced to compensate the effects of the identified critical faults. Currently, we are evaluating the proposed approach on other safety-critical applications.

Author Contributions: Conceptualization, D.P., J.S., S.B., M.S.R., R.B. and M.V.; methodology, D.P. and J.S.; investigation, D.P. and J.S.; resources, S.B. and R.B.; data curation, D.P., J.S. and S.B.; writing, D.P., J.S., S.B., M.S.B., R.B. and M.V.; supervision, M.S.B., R.B. and M.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: The research activity was supported by the Power Electronics Innovation Center (PEIC) of the Politecnico di Torino.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gall, H. Functional Safety IEC 61508/IEC 61511 the Impact to Certification and the User. In Proceedings of the IEEE/ACS International Conference on Computer Systems and Applications, Doha, Qatar, 31 March–4 April 2008.
2. Rivett, R.; Habli, I.; Kelly, T. Automotive Functional Safety and Robustness Never the Twain or Hand in Glove? In Proceedings of the CARS Critical Automotive Applications: Robustness & Safety, Paris, France, 4 September 2015.

3. Bagalini, E.; Sini, J.; Reorda, M.S.; Violante, M.; Klimesch, H.; Sarson, P. An automatic approach to perform the verification of hardware designs according to the ISO26262 functional safety standard. In Proceedings of the 18th IEEE Latin American Test Symposium (LATS), Bogota, Colombia, 13–15 March 2017.
4. Çetin, E.N. FMECA applications and lessons learnt. In Proceedings of the Annual Reliability and Maintainability Symposium (RAMS), Palm Harbor, FL, USA, 26–29 January 2015.
5. ISO26262. Road Vehicles—Functional Safety. 2011. Available online: https://www.iso.org/obp/ui/#iso:std:iso:26262:-1:ed-1:v1:en (accessed on 17 December 2018).
6. Peyghami, S.; Davari, P.; F-Firuzabad, M.; Blaabjerg, F. Failure Mode, Effects and Criticality Analysis (FMECA) in Power Electronic based Power Systems. In Proceedings of the 21st European Conference on Power Electronics and Applications (EPE '19 ECCE Europe), Genova, Italy, 3–5 September 2019.
7. Cickaric, L.S.; Katic, V.A.; Milic, S. Failure Modes and Effects Analysis of Urban Rooftop PV Systems—Case Study. In Proceedings of the International Symposium on Industrial Electronics (INDEL), Banja Luka, Bosnia and Herzegovina, 1–3 November 2018.
8. Zhang, Z.; Hao, M. Failure Mode and Effects Analysis of UAV Power System Based on Generalized Dempster-Shafer Structures. In Proceedings of the 2019 IEEE International Conference on Unmanned Systems (ICUS), Beijing, China, 17–19 October 2019.
9. Banerjee, P.; Pandey, K. Implementation of Failure Modes and Effect Analysis on the electro-hydraulic servo valve for steam turbine. In Proceedings of the IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), Delhi, India, 4–6 July 2016.
10. Sastry, A.; Kulasekaran, S.; Flicker, J.; Ayyanar, R.; Tamizhmani, G.; Roy, J.; Srinivasan, D.; Tilford, I. Failure modes and effect analysis of module level power electronics. In Proceedings of the 42nd Photovoltaic Specialist Conference (PVSC), New Orleans, LA, USA, 14–19 June 2015.
11. Rastayesh, S.; Bahrebar, S.; Bahman, A.S.; Sørensen, J.D.; Blaabjerg, F. Lifetime Estimation and Failure Risk Analysis in a Power Stage Used in Wind-Fuel Cell Hybrid Energy Systems. *Electronics* **2019**, *8*, 1412. [CrossRef]
12. Singh, V.; Pahuja, G. Failure Modes and Effects Analysis Using Fuzzy Logic for Electric Vehicle Inverter. In Proceedings of the 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India, 18–19 May 2018.
13. Sini, J.; D'Auria, M.; Violante, M. Towards Vehicle-Level Simulator Aided Failure Mode, Effect, and Diagnostic Analysis of Automotive Power Electronics Items. In Proceedings of the 2020 IEEE Latin-American Test Symposium (LATS), Maceio, Brazil, 30 March–2 April 2020.
14. Floridia, A.; Sanchez, E.; Reorda, M.S. Fault Grading Techniques of Software Test Libraries for Safety-Critical Applications. *IEEE Access* **2019**, *7*, 63578–63587. [CrossRef]
15. Parker, K.P. A New Process for Measuring and Displaying Board Test Coverage. In Proceedings of the Apex 2003, Anaheim, CA, USA, 4 September 2003. Available online: https://www.keysight.com/upload/cmc_upload/All/Apex_KParker_010903.pdf (accessed on 4 September 2003).
16. Borutzky, W. Combining behavioral block diagram modeling with circuit simulation. In *Computer Aided Systems Theory—EUROCAST '89*; Pichler, F., Moreno-Diaz, R., Eds.; EUROCAST 1989. Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 1990; Volume 410. [CrossRef]
17. Hellerstein, J.L.; Diao, Y.; Parekh, S.; Tilbury, D.M. *Feedback Control of Computing Systems*; Wiley-IEEE Press: Hoboken, NJ, USA, 2004.
18. Hick, H.; Bajzek, M.; Faustmann, C. Definition of a system model for model-based development. *SN Appl. Sci.* **2019**, *1*, 1074. [CrossRef]
19. Rutenbar, R.A.; Gielen, G.G.E.; Antao, B.A. Multilevel and mixed-domain simulation of analog circuits and systems. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **1996**, *15*, 68–82.
20. Pletea, I.-V.; Alexa, D.; Goras, T. Multilevel modeling and simulation of a switched reluctance machine. In Proceedings of the 24th International Spring Seminar on Electronics Technology, Concurrent Engineering in Electronic Packaging, ISSE 2001, Conference Proceedings (Cat. No.01EX492), Calimanesti-Caciulata, Romania, 5–9 May 2001; pp. 248–252.
21. Shen, M. Fast Simulation Model of Hybrid Modular Multilevel Converters for CPU. In Proceedings of the 2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE), Xiamen, China, 18–20 October 2019; pp. 32–36.

22. Liu, H.; Tolbert, L.M.; Ozpineci, B.; Du, Z. Hybrid multilevel inverter with single DC source. In Proceedings of the 2008 51st Midwest Symposium on Circuits and Systems, Knoxville, TN, USA, 10–13 August 2008; pp. 538–541.
23. Wu, M.; Wang, W. The multilevel simulation of analog circuits. In Proceedings of the IEEE APCCAS 2000, 2000 IEEE Asia-Pacific Conference on Circuits and Systems, Electronic Communication Systems. (Cat. No.00EX394), Tianjin, China, 4–6 December 2000; pp. 497–500.
24. Van Duijsen, P.J. Multilevel modeling and simulation of power electronic systems. In Proceedings of the 1993 Fifth European Conference on Power Electronics and Applications, Brighton, UK, 13–16 September 1993; Volume 4, pp. 347–352.
25. PSIM tool, PowerSim, User Manual. Available online: https://powersimtech.com/products/psim/ (accessed on 1 May 2020).
26. MATLAB Tool, Mathworks, User Manual. Available online: https://it.mathworks.com (accessed on 1 May 2020).
27. Mentor. Testing Analog/Mixed-Signal Circuits Tessent DefectSim. Available online: http://s3.mentor.com/public_documents/datasheet/products/silicon-yield/tessent-defectsim-ds.pdf (accessed on 12 June 2020).
28. Synopsys. Synopsys TestMAX CustomFault. Available online: https://www.synopsys.com/content/dam/synopsys/implementation&signoff/datasheets/testmax-customfault-ds.pdf (accessed on 30 July 2020).
29. Slamani, M.; Kaminska, B. Analog circuit fault diagnosis based on sensitivity computation and functional testing. *IEEE Des. Test Comput.* **1992**, *9*, 30–39. [CrossRef]
30. Arabi, A.; Bourouba, N.; Belaout, A.; Ayad, M. Catastrophic faults detection of analog circuits. In Proceedings of the 7th International Conference on Modelling, Identification and Control (ICMIC), Sousse, Tunisia, 18–20 December 2015.
31. Duhamel, P.; Rault, J. Automatic test generation techniques for analog circuits and systems: A review. *IEEE Trans. Circuits Syst.* **1979**, *26*, 411–440. [CrossRef]
32. ISO26262 Standard. Available online: https://www.iso.org/standard/43464.html (accessed on 12 November 2011).
33. AIAG & VDA. *AIAG & VDA FMEA Handbook*; FMEAAV-1; AIAG & VDA: Southfield, MI, USA, 2019.
34. ECSS. *ECSS-Q-ST-30-02C Handbook*; ECSS: Cologne, Germany, 2009.
35. Sini, J.; Violante, M. An Automatic Approach to Perform FMEDA Safety Assessment on Hardware Designs. In Proceedings of the 2018 IEEE 24th International Symposium on On-Line Testing and Robust System Design (IOLTS), Platja d'Aro, Spain, 2–4 July 2018.
36. Milanovič, M.; Rodič, M.; Truntic, M. Functional safety in power electronics converters. In Proceedings of the 2017 19th International Conference on Electrical Drives and Power Electronics (EDPE), Dubrovnik, Croatia, 4–6 October 2017.
37. Piumatti, D.; Borlo, S.; Reorda, M.S.; Bojoi, R. Assessing the effectiveness of different test approaches for power devices in a PCB. *IEEE J. Emerg. Sel. Top. Power Electron.* **2020**, *1*. [CrossRef]
38. Piumatti, D.; Reorda, M.S. Assessing Test Procedure Effectiveness for Power Devices. In Proceedings of the 2018 Conference on Design of Circuits and Integrated Systems (DCIS), Lyon, France, 14–16 November 2018.
39. Piumatti, D.; Borlo, S.; Mandrile, F.; Reorda, M.S.; Bojoi, R. Assessing the Effectiveness of the Test of Power Devices at the Board Level. In Proceedings of the XXXIV Conference on Design of Circuits and Integrated Systems (DCIS), Bilbao, Spain, 20–22 November 2019.
40. STGIPS30C60T-H. Three-Phase Inverter, STMicroelectroncs, Datasheet. Available online: https://www.st.com/en/power-modules/stgips30c60t-h.html (accessed on 10 April 2015).
41. STM32F446RE Microcontroller, STMicroelectroncs, Datasheet. Available online: https://www.st.com/en/microcontrollers-microprocessors/stm32f446re.html (accessed on 28 July 2020).
42. STMicroelectroncs. System-Adaptive Real-Time Accelerator ART. Available online: https://www.st.com/content/ccc/resource/training/technical/product_training/group0/7d/83/8c/1f/3a/1c/43/1e/STM32H7-System-Adaptive_Real-Time_Accelerator_ART/files/STM32H7-System-Adaptive_Real-Time_Accelerator_ART.pdf/_jcr_content/translations/en.STM32H7-System-Adaptive_Real-Time_Accelerator_ART.pdf (accessed on 27 May 2020).
43. STTH12S06 Power Diode, STMicroelectroncs, Datasheet. Available online: https://www.st.com/en/diodes-and-rectifiers/stth12s06.html (accessed on 2 February 2020).
44. STGF19NC60 Power IGBT, STMicroelectroncs, Datasheet. Available online: https://www.st.com/content/st_com/en/products/power-transistors/igbts/stpower-igbts-600-650v/stgf19nc60hd.html (accessed on 19 October 2020).

45. FAN9673 PFC/PSU Analog Controller, ON Semiconductor, Datasheet. Available online: https://www.onsemi.com/products/power-management/ac-dc-controllers-regulators/power-factor-controllers/fan9673 (accessed on 20 June 2019).
46. Sunter, S. Analog Fault Simulation—A Hot Topic! In Proceedings of the 25th IEEE European Test Symposium (ETS), Tallinn, Estonia, 25–29 May 2020.
47. PLECS Tool, Plexim, User Manual. Available online: https://www.plexim.com/plecs (accessed on 5 June 2020).
48. Gautschi, M.; Schiavone, P.D.; Traber, A.; Loi, I.; Pullini, A.; Rossi, D.; Flamand, E.; Gurkaynak, F.K.; Benini, L. Near-threshold risc-v core with dsp extensions for scalable iot endpoint devices. *IEEE Trans. Very Large Scale Integr. Syst.* **2017**, *25*, 2700–2713. [CrossRef]