IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 28, 2018, accepted October 7, 2018, date of publication October 16, 2018, date of current version November 14, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2876318

New Insights Into Soft-Faults Induced Cardiac Pacemakers Malfunctions Analyzed at System-Level via Model Checking

GHAITH BANY HAMAD[1], MARWAN AMMAR[1], OTMANE AIT MOHAMED[1], (Member, IEEE), AND YVON SAVARIA[2], (Fellow, IEEE)

[1]Electrical and Computer Department, Concordia University, Montreal, QC H3G 1M8, Canada
[2]Groupe de Recherche en Microelectronique et Microsystemes, Polytechnique Montréal, Montreal, QC H3T 1J4, Canada

Corresponding author: Ghaith Bany Hamad (g_banyha@ece.concordia.ca)

ABSTRACT Progressive shrinking of CMOS device sizes has permitted reductions in power consumption and miniaturization of electronic devices. In parallel, modern pacemakers implemented with advanced technologies have proved to be more sensitive than earlier models to soft errors induced notably by external radiations. Traditionally, the analysis of the impact of *soft faults* (SFs), such as those induced by single-event upsets, on the behavior of pacemaker devices, has been carried out by *dynamic radiation ground testing* and *clinical observations*. However, these techniques are expensive. They can only be done very late in the design cycle, after the design is manufactured and in part after it is implanted. This paper presents a new model-based analysis of the impact of SFs on the behavior of cardiac pacemakers at a system level. It is performed by: 1) introducing a new *probabilistic timed automata* (PTA) model; 2) verifying this model against a set of functional properties to ensure it meets its specifications under normal conditions; 3) applying a new methodology to inject SFs at a certain time in the PTA model of the pacemaker and to verify their impact on the pacemaker's behavior is introduced; and 4) identifying different scenarios for SFs that may lead to malfunction, including *oversensing, undersensing*, and *output failure*. The reported formal modeling is done in PRISM and the analysis is done with the Storm model checker.

INDEX TERMS Pacemakers, radiation effects, formal specifications, formal verification, system analysis and design, reliability.

I. INTRODUCTION

The rate at which pacemakers are implanted is increasing on a global scale, with more than 700,000 new pacemakers implanted worldwide each year [1]. These devices operate at a very low voltage to reduce their power consumption and to improve their battery life. Moreover, the input nodes in a pacemaker are very sensitive, since the behavior of the system relies on capturing the intrinsic electrical signals of the heart. Therefore, the sensors of a pacemaker are very susceptible to environmental interferences. In order to achieve the high level of reliability required by these safety-critical systems, the design of the pacemaker must feature different protective measures, such as shielding in hermetic metal cases, signal filtering, inference rejection circuit, and bipolar leads [2]. With such measures, a pacemaker is protected against the everyday sources of electromagnetic radiation, such as cell phones, microwave ovens, and articles surveillance equipment, i.e., these sources are no considerable threat to the functionality of the pacemaker. However, sometimes these devices need to operate in a hostile environment (high density of radiation) that can lead to *soft-faults* which can then induce soft-errors. For these devices, there are two main environments under which they are more affected by soft-errors:

- **Radiation based treatment or exams in hospitals:** in such case, the patient is exposed to high-density external radiations. Several accounts of deaths in pacemaker patients due to Magnetic Resonance Imaging (MRI) were reported by the Food and Drug Administration (FDA) [3]. Another example of a hostile environment in hospitals is the radiotherapy treatment of cancer [4].
- **Latitude and altitude:** Soft-Faults (SFs) due to high-energy protons and neutrons vary with both latitude and altitude. For example, while traveling in an airplane, the density of high-energy protons can be 100-800 times worse than at sea-level. Electrical reset was observed during air travel due to SFs [5].

In the literature, work related to the analysis of the sensitivity of implantable cardiac devices to soft-errors is rather limited. Most of the existing techniques (such as [4]–[11]) are based on *dynamic radiation ground testing*. This analysis provides a very accurate estimation of the vulnerability of the pacemaker. However, it is very complex, expensive, and time-consuming. Therefore, there is a growing need to analyze and estimate the impact of soft-errors due to *soft-faults* on high-level models of such systems.

In this paper, we introduce a novel methodology to quantitatively analyze the vulnerability of pacemakers to soft-errors induced by *soft-faults* in system-level models. This work is distinct from previous works in the following ways:

- A new modeling of the behavior of the *Dual Chamber, Pacing, and Sensing* (DDD) pacemaker, at the system-level, is proposed. In this model, the behavior of each sub-component is modeled as a *Probabilistic Timed Automaton (PTA)*. This is a key aspect of reliable fault analysis since probabilistic, non-deterministic behavior often arises in the presence of soft-faults and component errors. Moreover, we created a catalog of properties based on the DDD pacemaker specification and previous formal analyses of pacemakers in the literature. The correctness of the composite model is proved by verifying it against all these properties using Probabilistic Model Checking (PMC).
- A new formal probabilistic analysis of the impact of *soft-faults* on the behavior of the DDD pacemaker is proposed. The goal of this analysis is to provide full insight into the affected components, injection time, and the impact of SFs on the pacemaker behavior during each Window Of Vulnerability (WOV). This is achieved by extending the PTA of each component of the pacemaker model in order to allow fault injection and to include stochastic transitions that represent *soft-fault* propagation. The proposed technique enables quantitative investigation of SFs propagation for each injection scenario through verification of the extended pacemaker model against a set of Probabilistic Computation Tree Logic (PCTL) properties using probabilistic model checking. With this approach, different possible WOVs of the cardiac cycle have been identified. Each WOV is defined as the time interval within which a *soft-fault* at one component of the pacemaker might impair its behavior. Then, for each WOV, we investigate the impact of a *soft-fault* on the pacemaker and on the proper behavior of the heart. If the injected *soft-fault* results into an observable soft-error, then we identify its impact on the *cardiac cycle* when it is injected and on the following cycles. Thereafter, we link the observed behaviors based on these injection scenarios with the pacemaker malfunction behaviors which were reported in the literature as part of radiation experiments or clinically in patient's records. As it is explained later, a *soft-fault* can lead to undesirable events, such as *Oversensing*, *Undersensing*, and *Output failure* in the pacemaker.

The rest of this paper is organized as follows. Section II-A summarizes the main techniques used to validate the impact of *soft-faults* on the behavior of the pacemaker and the main malfunctions observed in the literature. In Section II-B, we review the existing formal techniques that can be used to analyze the functionality of the pacemaker. In Section III, we give a brief overview of the probabilistic model checking techniques utilized in this work. The main steps of the proposed methodology are summarized in Section IV. In Section V, we explain the behavior of the DDD pacemaker. In Section VI, the modeling of the DDD pacemaker as Probabilistic Timed Automata (PTA) and the analysis of this model using the *Storm* model checker is explained in details. Our proposed probabilistic modeling and analysis of the impact of SFs are explained in detail in Section VII. In Section VIII, the impact of SFs during each WOV is characterized and described. Section IX concludes the work by summarizing our main contributions and results.

II. RELATED WORK

A. EXISTING OBSERVATIONS OF THE IMPACT OF SOFT-FAULTS ON PACEMAKERS

In this section, we summarize the main findings in the literature on the impact of external radiations on the behavior of implantable cardiac pacemakers. For further information, the reader is referred to several reviews in the literature (such as [4] and [7]) that provide full details of these results. In the literature, the presence of soft-errors due to SFs in pacemakers was proven by:

1) *Clinical observations*: In these approaches, data collected from implanted pacemakers are analyzed. The reported results in [10] evaluate the incidence of SFs induced by cosmic neutron radiation in a large population of patients with cardiac implants. Other clinical observations (similar to the work done in [5]) demonstrate that the high density of cosmic radiation during air travel is linked to the electrical reset identified in the cardiac devices implanted in multiple patients. Moreover, different malfunctions were observed in pacemakers implanted in patients who suffer from cancer and have been treated by radiotherapy [9], [12]. This kind of analysis is very time consuming and only possible after pacemakers are implanted and operating in the patient's body.
2) *Dynamic radiation ground testing*: With this approach, accelerated device testing is performed under different radiation fluxes (such as [6] and [9]–[11]). The goal of this analysis is to replicate the observed behavior obtained from patient data and to test the vulnerability of different pacemaker models. This approach is very accurate and enables the testing of the device when exposed to a specific radiation intensity in a highly controlled environment. However, the procedure is very expensive and only possible at post-silicon level (i.e., after the pacemaker is fully manufactured).

Both *clinical observations* and *dynamic radiation ground testing* show that the impact of SFs on the behavior of the pacemaker can be classified into three groups: 1) minor errors which do not impact the system and are only recorded in the data log of the device; 2) moderate reset, not requiring correction by the programmer; and 3) electrical reset, requiring full reprogramming of the device. Based on these results, it is evident that SFs present a real challenge to the reliability of pacemakers. However, existing analysis techniques are resource hungry, time-consuming, and require the pacemaker to be fully manufactured or even implanted.

B. FORMAL MODELING AND ANALYSIS OF THE FUNCTIONALITY OF THE PACEMAKERS

Pacemakers are safety-critical devices of which faulty behaviors can cause harm or even death. There has been much interest in developing formal verification frameworks to verify the correctness of pacemakers implementations, at different abstraction levels. Existing techniques can be classified into two categories: formal based techniques (such as [13]–[16]) and testing based techniques (such as [17] and [18]). Formal verification techniques are very efficient in providing guarantees about the pacemaker model correctness, as well as locating corner-cases and hard-to-find bugs.

Gomes and Oliveira [13] propose a formal specification of a pacemaker using the *Z* model into Perfect Developer [19]. Thereafter, based on the pacemaker specification, the correctness of the generated model was verified using the ProofPower-Z theorem prover. Tuan *et al.* [14] proposed the modeling of the different operating modes of a pacemaker as a Real-Time System (RTS) formal model. This model was then verified against a number of safety and correctness properties as well as timed constraints using the PAT model checker.

In the work proposed by [15] and [16], a model-based framework for the automatic verification of the functionality of cardiac pacemakers was developed. The authors developed a detailed model of a basic dual-chamber pacemaker. This model is constructed based on the timed automaton [20] (TA) of each of the pacemaker's sub-components. Moreover, in this work, the authors have developed a TA of the heart behavior. The functionality of the pacemaker model has been verified using UPPAAL [21].

In [22], a quantitative functional verification algorithm for implantable pacemakers is proposed by connecting the MATLAB model of the heart, introduced in [23], and the TA model of the pacemaker in PRISM, proposed by [15]. The analysis is performed by combining both models (after exporting the PRISM model to MATLAB) and verifying the pacemaker according to its specifications.

These techniques are designed to detect bugs in the pacemaker implementation (i.e., identification of functional errors). In other words, such techniques assume that the pacemaker always operates in an error-free environment. Therefore, with these techniques, it is not possible to detect or analyze the impact of non-functional faults such as *soft-faults*. The work presented in this paper tries to bridge the gap between high-level functional design verification and physical radiation testing, by providing a technique to perform a high-level analysis of the vulnerability of pacemakers, in hostile environments, at a very early stage of the design cycle.

III. PROBABILISTIC MODEL CHECKING (PMC) & *STORM*

In this work, we use *Storm* [24], a powerful probabilistic symbolic model checker. It employs efficient algorithms and data structures to reduce the number of states and optimize the size of the state-machine to be solved. In addition, *Storm* supports different implementations of Markov chains, namely discrete-time and continuous-time Markov chains and Markov Automata. It also supports a wide range of probabilistic temporal logic to specify the properties to be verified such as PCTL, PCTL*, and Continuous Stochastic Logic (CSL) [25], [26]. *Storm* supports several types of input such as PRISM, JANI, GSPNs, DFTs, cpGCL. In this paper, all the models are built in PRISM language [27]. In PRISM, a model is formed by basic constructs called modules, each designed to express a specific behavior, much like subcomponents of a system. The state of each module is given by a set of finite ranged variables. The global state of the model is determined by the evaluation of the values of the module variables.

IV. STEPS OF THE PROPOSED PACEMAKER ANALYSIS

This paper introduces a unified verification methodology to investigate, at the system-level, the impacts of soft-faults on the behavior of the DDD pacemaker model. As shown in Fig. 1, this methodology comprises the following two main phases:

Phase 1 (Model Construction and Functional Verification): this phase starts by extracting the specification and constructing a system-level model of the pacemaker main components as explained in Section V. Thereafter, the model of the pacemaker is constructed based on the PTA of the subcomponents. This model is verified against a set of functional properties to ensure its correctness. The model and functional verification steps are explained in detail in Section VI.

Phase 2 (Soft-Fault Impact Analysis): this phase operates over the model built and verified in *phase 1*. Based on the pacemaker behavior, different possible SF injection scenarios are identified. In order to analyze the impact of each SF, for each scenario, the models of the pacemaker's subcomponents and of the heart were modified to build the required fault propagation environment. The model and analysis for each injection scenario are explained in Section VII. The observed results are characterized and compared with the reported results from the *dynamic radiation ground testing* and/or *Clinical observations*. These results are summarized and discussed in Section VIII.

V. BEHAVIOR OF THE DDD PACEMAKER

The DDD pacemaker consists of five main components, defined as event-triggered timing cycles. The timing cycles

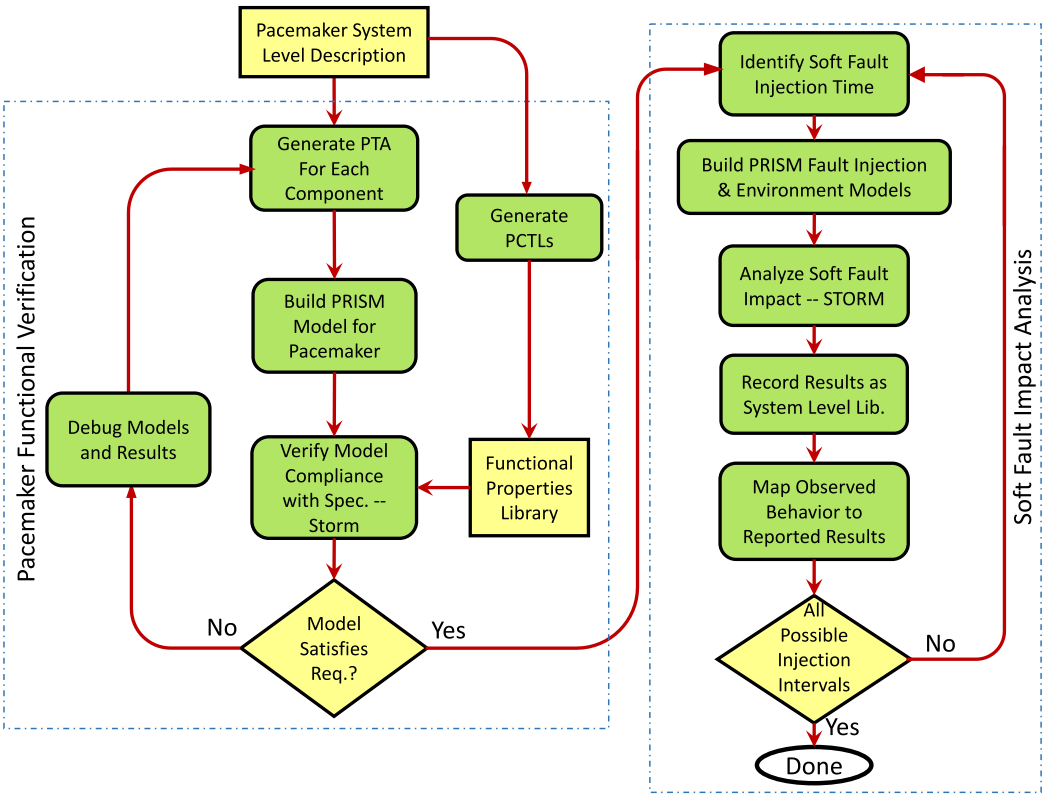


FIGURE 1. Main steps of the proposed analysis of a pacemaker.

communicate with each other through broadcasting channels, shared variables and events. In order to provide optimal hemodynamic benefit to the patient, dual-chamber pacemakers strive to mimic the normal heart rhythm. This pacemaker acts on demand, taking appropriate actions in reaction to what is happening inside the person's heart, at any given time. In this section, we explain the behavior of the main components of the DDD pacemaker. In this pacemaker, the Lower Rate Interval (LRI) is the rate at which the pacemaker will pace the atrium in the absence of intrinsic atrial activity. Similar to single-chamber timing, the lower rate can be converted to a lower rate interval or the longest period of time allowed between atrial events. The LRI component is responsible for measuring the heartbeat rate and keeping it above a defined minimum value. The PTA of the LRI component is shown in Fig. 2(A). According to this figure, the LRI component monitors the ventricular sensing, ventricular pacing, and atrial sensing events. The clock of the component is reset after a ventricular event is sensed. If no atrial event is sensed within a certain amount of time, the LRI component triggers an atrial pacing (after $T_{LRI} - T_{AVI}$).

The time that a DDD pacemaker is required to wait between a sensed or paced atrial event and a ventricular event is called the Atrio-Ventricular Interval (AVI). This behavior is implemented by the AVI component. Fig. 2(B) depicts the PTA of the expected behavior of the AVI component. As shown in this figure, the AVI component sets the longest interval between an atrial and a ventricular event. After the occurrence of an atrial event, if no ventricular event is sensed within a certain time (T_{AVI}), a ventricular pacing is performed. Correlatively, the Upper Rate Interval (URI) is defined as the upper activity rate i.e., the minimum time delay between consecutive sensor-indicated ventricle events. The URI component implements this behavior and its PTA is shown in Fig. 2(C).

The Post-Ventricular Atrial Refractory Period (PVARP) is the period of time after a ventricular pace or sense, when the atrial channel is in a refractory state. In other words, the occurrence of atrial senses during this period is identified by the pacemaker but do not initiate the *A-V* interval (AR). This behavior is implemented by the PVARP component, with its PTA depicted in Fig. 2(E). The purpose of the PVARP component is to avoid premature atrial contractions. This inhibits the beginning of an irregular *A-V* interval, which would cause the pacemaker to pace at a higher than desired rate. Similarly, the Ventricular Refractory Period (VRP) is designed to avoid restarting the *V-A* interval due to a noise wave. This behavior is performed by the VRP component. Fig. 2(D) depicts the PTA of this functionality. As shown in Fig. 2, ventricular sensed events, occurring in the noise sampling portion of the ventricular refractory period, are identified but will not restart the *V-A* interval.

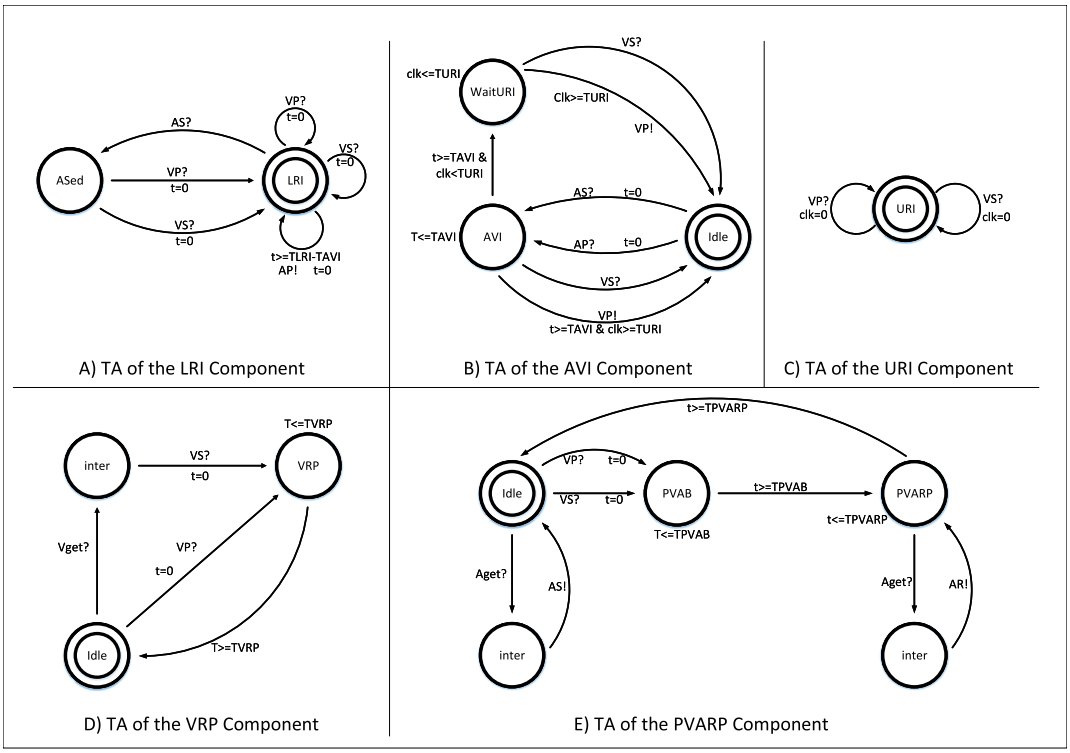


FIGURE 2. TAs of the components of the DDD pacemaker. (a) TA of the LRI Component. (b) TA of the AVI component. (c) TA of the URI component. (d) TA of the VRP component. (e) TA of the PVARP component.

VI. PTA MODELING & FUNCTIONAL ANALYSIS OF PACEMAKER

In Section V, the main components of the DDD pacemaker (LRI, AVI, PVARP, and VRP) are described as PTAs. As explained before, the functionality of each of these components is mainly controlled by the tight synchronization with the other components. In these PTA, the behavior of the real-time system is controlled through a finite set of *clocks* χ. The values of these clocks range over the domain $\mathbb{R}_{\geq 0}$ (i.e., non-negative real numbers). A function $\nu : \chi \rightarrow \mathbb{R}_{\geq 0}$ is referred to as a clock valuation. The set of all clock valuations is denoted by $\mathbb{R}^{\chi}_{\geq 0}$. For any $\nu \in \mathbb{R}^{\chi}_{\geq 0}$, $t \in \mathbb{R}_{\geq 0}$, and $X \subseteq \chi$, we use $\nu{+}t$ to denote the clock valuation which increments all clock values in ν by t, such that $\nu(X) + t$. We use $\nu[X := 0]$ for the clock valuation in which clocks in X are reset to 0. The set of clock constrains over χ, denoted $\zeta(X)$ is defined inductively by the syntax:

$$\zeta ::= true | x \leq d | c \leq x | x + c \leq y + d | \neg\zeta | \zeta \wedge \zeta$$

where $x, y \in \chi$ and $c, d \in \mathbb{N}$. The clock valuation ν satisfies the clock constraint $\zeta(X)$, denoted by $\nu \models \zeta$, if and only if X resolves to true after substituting each clock $X \in \chi$ with the corresponding clock value $\nu(X)$.

Definition 1: A probabilistic timed automaton is defined by a tuple $A = (L, L_0, \chi, Act, P, \mathscr{L})$*, where:*

- *L is a finite number of states.*
- L_0 *is the initial state.*
- χ *is a finite set of clocks.*
- *Act is a finite set of actions over L.*
- *inv:* $L \rightarrow \zeta(X)$ *is an invariant condition.*
- *P is a probabilistic transition function* $L \times \zeta(X) \times Dist(2^X \times L)$.
- $\mathscr{L} : L \rightarrow 2^{AP}$ *is a labeling function assigning atomic propositions to different states.*

In a PTA, a state $(l, \nu) \in L \times \mathbb{R}^{\chi}_{\geq 0}$ such that $\nu \models inv(l)$. In any state (l, ν), there is a non-deterministic choice of either making a discrete transition or letting time pass. A discrete transition can be made according to any $(l, g, p) \in P$, with current state l being enabled and zone g is satisfied by the current clock valuation ν. The probability of moving to location l' and resetting all clocks in X to 0 is given by $p(X, l')$. The option of letting time pass is available only if the invariant condition $inv(l)$ is satisfied while time elapses.

The PTA for each component is represented as a separate PRISM module. In order to accurately construct a system-level model of the pacemaker, the parallel composition of the PTAs of the subcomponents is required. The proposed methodology introduces the soft-fault (SF) injection module. This module is responsible for injecting, tracking, and synchronizing SFs across all other components, during the analysis when a fault is injected. Furthermore, the addition of the SF injection module requires significant alterations to all other modules of the pacemaker. The main challenge is to ensure that these additions and new components do not impact the core functionality of the pacemaker device, as stated in its specifications. Therefore, the proposed PTA

TABLE 1. Results of the verification of the functional properties of the pacemaker.

Component		Verified Property	Testing Time	Result
AVI	AVI.1	$Pmax =? [F\ (VP = 1)\&(VS = 0)]$	$\geq T_{AVI}$	Pass
	AVI.2	$Pmax =? [F\ (VP = 1)]$	$\geq T_{URI}$	Pass
	AVI.3	$Pmax =? [F\ (VP = 1)]$	$< T_{URI}$	Pass
	AVI.4	$Pmax =? [F\ (VP = 1)]$	$< T_{AVI}$	Pass
	AVI.5	$Pmax =? [F\ (VS = 1)\&(VP = 0)]$	$\leq T_{AVI}$	Pass
	AVI.6	$Pmax =? [F\ (VS = 0)\&(VP = 1)]$	$\geq T_{AVI}$	Pass
LRI	LRI.1	$Pmax =? [F\ (AP = 1)]$	$< T_{LRI} - T_{AVI}$	Pass
	LRI.2	$Pmax =? [F\ (AS = 0)\&(AP = 1)]$	$\geq T_{LRI} - T_{AVI}$	Pass
	LRI.3	$Pmax =? [F\ (AS = 1)]$	$\leq T_{LRI} - T_{AVI}$	Pass
	LRI.4	$Pmax =? [F\ (AS = 1)\&(AP = 0)]$	$\leq T_{LRI} - T_{AVI}$	Pass
	LRI.5	$Pmax =? [F\ (AS = 0)\&(AP = 1)]$	$\geq T_{LRI} - T_{AVI}$	Pass
PVARP	PVARP.1	$Pmax =? [F\ (AR = 1)]$	$\leq T_{PVARP}$	Pass
	PVARP.2	$Pmax =? [F\ (AS = 0)]$	$\leq T_{PVARP}$	Pass
	PVARP.3	$Pmax =? [F\ (AS = 1)]$	$\geq T_{PVARP}$	Pass
VRP	VRP.1	$Pmax =? [F\ (VRP = 1)\&(VS2 = 1)]$	$\leq T_{VRP}$	Pass
	VRP.2	$Pmax =? [F\ (VRP = 1)\&(VP2 = 1)]$	$\leq T_{VRP}$	Pass
	VRP.3	$Pmax =? [F\ (VRP = 0)\&(Idle3 = 1)]$	$\geq T_{VRP}$	Pass

model is verified against a catalog of functional Probabilistic Computation Tree Logic (PCTL [28]) properties obtained from an extensive review of the literature. The verified set of properties are shown in Table 1. These properties were asserted using the probabilistic model checker Storm, which provides a unique trade-off between performance and modularity, the supported solvers, and a wide range of supported modeling languages. These properties are defined based on the timing requirements of the pacemaker to verify: (i) whether the pacemaker rectifies any abnormal heart behavior by providing necessary pacing, and (ii) that the pacemaker does not induce anomalous heart behaviors by providing unnecessary pacing to the heart. We confirmed that the verified properties exhaustively verify the behavior of the pacemaker with existing analysis such as [15], [22], and [29]. This analysis proves that our model correctly implements the functionality of the pacemaker. It is important to note that, in this analysis, no faults were injected through the fault injection component. As shown in Table 1, for each component, a set of properties are verified to validate its timing requirements. The timing under which the property is verified is shown in the fourth column. The verified properties in Table 1 are the following:

- **AVI.1**: A ventricular pacing can only happen if no ventricular event is sensed within *TAVI*.
- **AVI.2**: A ventricular pacing can only happen at a time which is equal or greater than *TURI*.
- **AVI.3**: There is no reachable state where a ventricular pacing happens before the *TURI* time finished.
- **AVI.4**: There is no reachable state where a ventricular pacing happens before the *TAVI* time interval finishes.
- **AVI.5**: If a ventricular sensing happens, a ventricular pacing will not occur.
- **AVI.6**: If no ventricular sensing is detected within the expected time, a ventricular pacing will occur.
- **LRI.1**: An atrial pacing event will never take place while *test* before the *TLRI-TAVI* time interval finishes.
- **LRI.2**: If an atrial sensing is not detected within the time limit, an atrial pacing will take place.
- **LRI.3**: An atrial sensing may happen if *test* time is less or equal to *TLRI-TAVI*.
- **LRI.4**: If an atrial sensing happens, an atrial pacing will not take place.
- **LRI.5**: If an atrial sensing does not happen within the expected time, an atrial pacing will take place.
- **PVARP.1**: An atrial event happening during the time frame $t \leq$ *TPVARP* will be considered as an AR.
- **PVARP.2**: No atrial sensing will happen over time $t \leq$ *TPVARP*.
- **PVARP.3**: An atrial sensing may only happen when the time is greater than *TPVARP*.
- **VRP.1**: A ventricular sensing will not happen if time is lesser than *TVRP*.
- **VRP.2**: A ventricular pacing will not happen if time is lesser than *TVRP*.
- **VRP.3**: Component VRP will stop filtering ventricular signals when the time is greater than *TVRP*.

VII. NON-FUNCTIONAL ANALYSIS OF THE PACEMAKER VULNERABILITY TO SOFT-FAULTS

In this section, we present a system-level injection and analysis mechanism replicating the effects of *soft-faults* on the behavior of the pacemaker during the A-V cycle. A new formal analysis is proposed to model and analyze each of the SF scenarios, as well as to provide new insights on the affected component, injection time, and the impact on the pacemaker behavior. The main components of the proposed analysis are shown in Fig. 3. In this analysis, the pacemaker model proposed in Section VI is extended to incorporate the impact of SFs.

A. SOFT-FAULT CLASSIFICATION

One of the main objectives of this work is to propose a vulnerability analysis of the DDD pacemaker to temporal faults

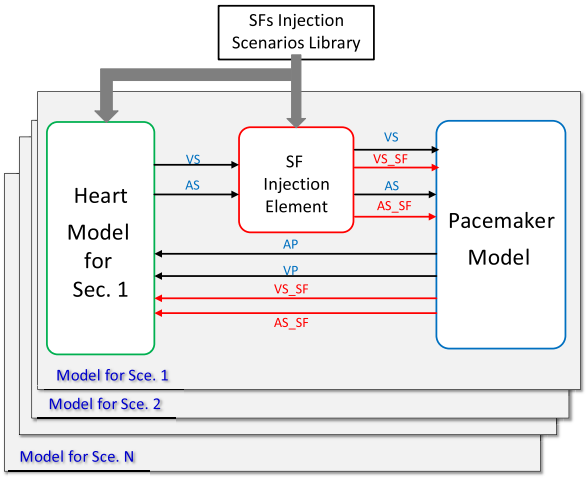


FIGURE 3. Proposed formal analysis of *soft-faults* propagation.

at the system-level. At such high level, most of the details of the system's hardware implementation are abstracted and not yet available. Consequently, details of the sources of the vulnerability and their characteristics are not defined. In order to address these issues when modeling and analyzing temporal faults, soft-faults were introduced. A soft-fault is an abstract high-level view of any single temporal fault that can impact the pacemaker behavior for one cardiac cycle or more. Based on our review of the literature, soft-faults in pacemakers are mainly induced by ionizing radiation, which can be classified into two categories: 1) Total Ionizing Dose Effects (TIDs); and 2) Single-Event Effects (SEEs) [4]. TIDs occur due to the charge accumulation in the oxide layers of the device. The oxide layers suffer from degradation if the radiation exposure is above a certain threshold (normally between 10-50 Gy (Gray units)). These levels of exposure are common in medical treatments such as radiotherapy, x-rays, and fluoroscopy. It is reported in several papers (e.g., in [4]) that Implantable Cardiac Devices (ICDs) become increasingly sensitive to temporal errors over time (i.e., soft-faults). On the other hand, SEEs occur due to exposure to high concentrations of high Linear Energy Transfer (LET) particles, depositing sufficient charge to disturb normal circuit operation. Unlike total dose effects, single-event effects are ubiquitous. Thus, their significance to device reliability is of greater importance. Recent ground radiation experiments and clinical observations show that the most relevant sources of soft-faults are Single-Event Upsets (SEUs), usually originating from cosmic radiation, electromagnetic interference, or radiotherapy. Other types of single-effects are reported in the literature to have a negligible probability of occurrence [4].

In the proposed modeling and analysis, *soft-faults* are introduced into the pacemaker model by adding two input nodes to the pacemaker model, namely *AS_SF* and *VS_SF*. These inputs carry the SF signal i.e., it is possible to distinguish between the native (*AS* and *VS*) and faulty (*AS_SF* and *VS_SF*) events. *AS_SF* indicates the presence of an SF at the atrial input node of the pacemaker. *VS_SF* indicates the presence of an SF at the ventricular input node of the pacemaker. In our model, the pacemaker reacts to *AS_SF* and *VS_SF* as it reacts to *AS* and *VS*, respectively. As explained before, all the components of the pacemaker are synchronized based on the timing requirements of the pacemaker. It is important to note that in this analysis, it is assumed that the SFs are initiated outside the pacemaker. In other words, SFs are propagating through the main inputs (*AS, VS*). This can be justified due to the unavailability of a physical implementation of the pacemaker in this system-level analysis.

B. FORMAL MODELING AND ANALYSIS OF SOFT-FAULTS

In order to investigate the SF propagation, we extended the models of each component to include stochastic transitions representing SF propagation. This is achieved through the SF Injection Element (SIE), which interrupts the communication between the pacemaker and the heart. The SIE component tracks and processes the native *AS* and *VS* signals based on the desired injection scenario. Another purpose of the SIE component is to generate either *AS_SF* and *VS_SF* to derive the desired injection scenarios, as shown in Fig. 3. The focus of this paper is the analysis of the pacemaker model. Thus, an abstract model of the heart is used to cover only the desired behaviors. Our heart model is based on a simple synchronous communication protocol. The model is pre-programmed to release signals after a certain amount of time has passed, and to verify if the correct signals were received within a pre-defined threshold. For each scenario in our analysis, the models of the heart and of the SIE are slightly tweaked to produce the desired inputs for the pacemaker. For example, to produce the desired inputs for Sce.1, the heart model is assumed to function normally (cyclic generation of *AS* and *VS* signals). However, the SIE is programmed to eventually inject an *AS_SF* signal in the pacemaker. This injection happens after the native *VS* event, but prior to the next native *AS* event. Other scenarios may require different setup, such as Sce.4. In this scenario, the heart model is assumed to fail to produce an *AS* event, after a few cycles. When this happens, the SIE immediately injects an *AS_SF* signal in the pacemaker. Modifying the heart and SF models in this way greatly facilitates the conduct of the experiments.

The modeling is done by generating the PTA of each component presented in Section VI. Each PTA is then represented as an individual PRISM module. For the purpose of our experiments, the pacemaker model is synchronized to a very simplistic heart model. Our heart model is a two-steps process that will generate one of two signals at different time delays. These time delays are defined based on the A-V cycle time delays used in [30]. The PTA of the system is obtained by the parallel composition of all PTAs of all the components of the pacemaker and the heart.

Table 2 shows the library of properties used to verify the different SF injection scenarios in different components. The second column depicts the time at which the SF is

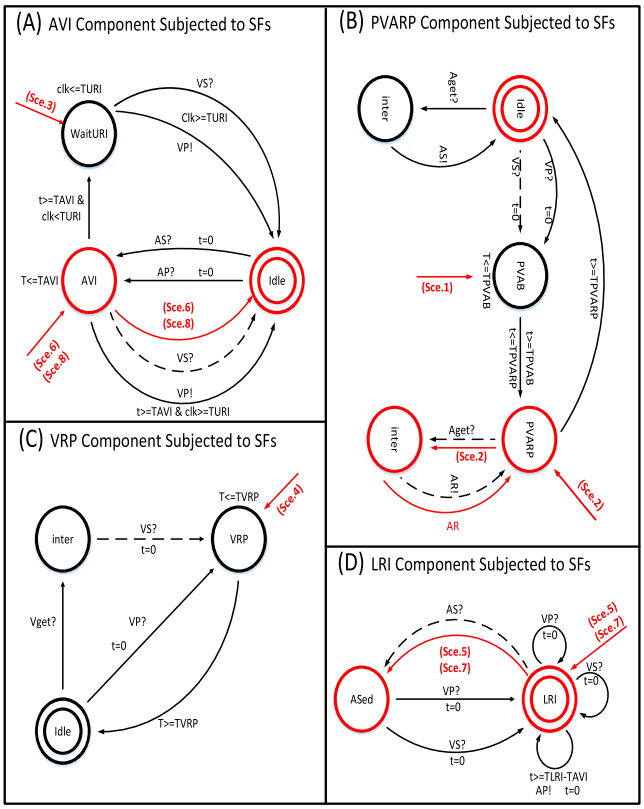


FIGURE 4. Effects of SFs on the pacemaker components. (a) AVI component subjected to SFs. (b) PVARP component subjected to SFs. (c) VRP component subjected to SFs. (d) LRI component subjected to SFs.

injected. The previous conditions under which the SF is injected are specified in the third column. These conditions indicate which component in the pacemaker is active and the next expected action. The fourth column shows the time in which the pacemaker model is verified against the desired properties. The properties verified in each injection scenario are reported in the fifth column. The sixth column shows the maximum probability of the verified event occurring, ranging from 0 to 1 (where 1 means 100% probability). The corresponding timing diagram of each scenario is shown in the last column of the table.

Fig. 4 illustrates an abstract view of the different SFs injection scenarios analyzed in this work. In each sub-figure, the default state is represented by two concentric circles. The red arrows entering the components show the SF injection states in the system. Finally, the red states and transitions show which subcomponents are impacted by the SF injection. Fig. 4(A) depicts the scenario where an SF may be injected while the *AVI* component operates in the *AVI* state. The effects of such SFs are verified as shown in Table 2 (identified as *Sce.6* and *Sce.8*). Such SF may be interpreted by the system as a native VS signal. In other words, the component may erroneously identify the SF as a native ventricular activity. This event causes the state of the component to change to *idle*. However, this SF can have different effects on the behavior of the pacemaker, which are detailed in Section VIII. In the case of *Sce.6*, after the pacemaker senses an AS from the heart, the SIE injects an SF at a random time before TAVI. In this scenario, we verified three properties. With the first property, we verify that the injected SF is eventually received by the pacemaker and recognized as a native VS. The second property verifies that after the SF is received, the pacemaker resets its clock and waits for a time interval $TLRI - TAVI$, but no AS is sensed within that time interval. The third property verifies that the pacemaker erroneously releases an atrial pacing (i.e., wrong atrial pacing due to *oversensing*). Similarly, in *Sce.8* the SIE injects an SF at a random time before TAVI, once the pacemaker senses an AS. In this scenario, we first verify that the injected SF is received by the pacemaker and recognized as a native VS. The second property verifies that after the SF is received, the pacemaker does not release the required ventricular pacing.

Another possible scenario that was analyzed is the injection of an SF in the *AVI* component at the ventricular input (VS_SF) after a native VS is sensed (i.e., during *TURI*). In this case, the *AVI* component operates in *WaitURI* state. This is shown in Table 2 and Fig. 4(A) as *Sce.3*. The pacemaker model is verified against three properties. With the first property, we verify that the incidence of SF is perceived by the pacemaker during time interval *TURI*. The second property is designed to verify that the next ventricular event is only received after time interval *TURI*. With the last property, we verify that the next atrial event (AS) is sensed within the time interval *TURI* after the native VS.

The PTA in Fig. 4(B) shows the two effects that SFs may produce on component *PVARP*, identified by *Sce.1*, and *Sce.2*. In *Sce.1*, after the pacemaker senses a native VS, the SIE randomly injects an SF at the atrial input (AS_SF) during the time interval *TPVAB* (while the *PVARP* component at state PVAB). The injection of this SF is shown in Table 2. First, we verify that the injected SF is received by the pacemaker at the specified time. The second property allows us to verify that this SF has no impact on the system. This is done by checking that the next atrial event is correctly sensed by the pacemaker (i.e., no change in the A-V timings). In *Sce.2*, the SIE injects the SF at AS_SF2 after VS is sensed (i.e., during time interval $TPVAB < T < TPVARP$). The first two properties check if the injected SF is received by the pacemaker and characterized as a native AR event. The last property is used to verify that this SF does not affect the timing of the pacemaker. This is achieved by ensuring that the next AS event occurs within the time period $TLRI - TAVI$ as expected. Fig. 4(C) shows the effect of injecting an SF at the *VRP* component. In this scenario (Sce.4), after a native VS or VP event, the SIE injects an SF at a random time within the time interval *TVRP*. In order to verify this scenario, the first property is used to check that the injected SF is received by the pacemaker at the specified time. The second property verifies that the SF is completely masked by validating that the next AS is sensed within the TLRI-TAVI.

Finally, the PTA in Fig. 4(D) shows the scenarios where an SF is injected at the *LRI* component, represented by *Sce.5*

TABLE 2. Results of the verification of the non-functional properties related to the impact of SFs on the pacemaker.

Scenario	Injection Time	Previous Events	Verification Time	Verified PCTL Property	Result	Situation illustrated in figure
Sce.1	$Tinj < TPVAB$	VS=1	$Tver > Tinj$	Pmax=? [F (AS_SF=1)]	1	Fig. 6
			$Tver > TLRI - TAVI$	Pmax=? [F (AS=1)]	1	
Sce.2	$Tinj > TPVAB$	VS=1	$Tver < TPVARP$	Pmax=? [F (AS_SF=1)]	1	Fig. 7
			$Tver < TPVARP$	Pmax=? [F (AR=1)]	1	
			$Tver > TLRI - TAVI$	Pmax=? [F (AS=1)]	1	
Sce.3	$Tinj < TURI$	VS=1	$Tver < Tinj$	Pmax=? [F (VS_SF=1)]	1	Fig. 8
			$Tver > TURI$	Pmax=? [F (VS=1)]	1	
			$Tver < TURI$	Pmax=? [F (AS=1)]	1	
Sce.4	$Tinj < TVRP$	VP/VS=1	$Tver < Tinj$	Pmax=? [F (VS_SF=1)]	1	Fig. 9
			$Tver > TLRI - TAVI$	Pmax=? [F (AS=1)]	1	
Sce.5	$Tinj > TPVARP$	VS=1	$Tver > Tinj$	Pmax=? [F (AS_SF=1)]	1	Fig. 10
			$Tver <= TAVI$	Pmax=? [F (VS=1)]	0	
			$Tver >= TAVI$	Pmax=? [F (VP=1)]	1	
Sce.6	$Tinj < TAVI$	AS=1	$Tver < TAVI$	Pmax=? [F (VS_SF=1)]	1	Fig. 11
			$Tver <= TLRI - TAVI$	Pmax=? [F (AS=1)]	0	
			$Tver >= TLRI - TAVI$	Pmax=? [F (AP=1)]	1	
Sce.7	$TPVARP < Tinj < TLRI - TAVI$	VS=1	$Tver >= Tinj$	Pmax=? [F (AS_SF=1)]	1	Fig. 12
			$Tver <= TLRI - TAVI$	Pmax=? [F (AP=1)]	0	
Sce.8	$Tinj <= TAVI$	AS=1	$Tver >= Tinj$	Pmax=? [F (VS_SF=1)]	1	Fig. 13
			$Tver >= TAVI$	Pmax=? [F (VP=1)]	0	

and *Sce.7*. In *Sce.5*, after a VS is sensed, the SIE randomly injects an SF at AS_SF within the time interval $TVARP < Tinj < TLRI - TAVI$. By verifying the properties for this scenario as shown in Table 2, we observed that this SF can be sensed by the pacemaker as a native AS. Next, we verify that the pacemaker component resets its internal clock and proceeds to wait for the VS signal, (i.e., the pacemaker has erroneously transitioned to the *ASed* state), breaking the A-V cycle synchronization. As shown in the result of the verification of the second property of scenario *Sce.5*, the occurrence of the SF causes the pacemaker to ignore the VS signal. In the last property, we verify that the pacemaker eventually applies an erroneous VP on the heart. In *Sce.7*, after a VS is sensed, the SIE randomly injects an SF at AS_SF. Similarly to *Sce.5*, this SF is injected within the time interval $TVARP < Tinj < TLRI - TAVI$. However, this SF has a different impact, which is verified as shown in Table 2. The first property confirms that this SF has been sensed as a native AS. The second property validates that such SF prevents the pacemaker from releasing the required AP.

VIII. NEW INSIGHTS ON POSSIBLE PACEMAKER MALFUNCTIONS INDUCED BY SOFT-FAULTS

In the previous sections, we introduced a system-level analysis approach designed to provide a high-level view of several possible scenarios that may lead to pacemaker malfunctions. Each of these high-level scenarios can be mapped to different occurrences of low-level faults. In this section, based on the results of the analysis introduced in Section VII, and on the behavior of the pacemaker explained in Section V, the observed pacemaker malfunctions are mapped to physical-level analysis results reported in the literature. These reports may originate from several different sources of errors at lower-level such as SEUs and Multiple-Bit Upsets (MBUs). Based on these reports and on our analysis results, different Windows Of Vulnerabilities (WOVs) are investigated. A WOV is defined as the time interval in which an SF at one component can impair the behavior of the pacemaker. For each WOV in the *cardiac cycle*, if the SF results in an observed soft-error, then we identify its impact on the *cardiac cycle* where it is injected and in the future cycles. For all these scenarios, the results of our analysis are characterized. Furthermore, we have researched the literature to identify if the observed behaviors are reported in *clinical observations* and/or as results of *dynamic radiation ground testing* of pacemakers. In the next subsections, we explain all the SFs injection scenarios, which are classified into a given subsection based on their eventual impacts. Following are some general considerations pertaining to all scenarios:

- The impact of each SF scenario (Fig. 5 to 12) is demonstrated for two cardiac cycles, as seen from the pacemaker.
- Native heart events (atrial sensing (AS) and ventricular sensing (VS)) are identified in the figures by solid black pulses, where the peaks represent atrial activity and the valleys represent ventricular activity.
- Dotted pulses represent missing or masked cardiac events or pacing events.
- Red pulses represent either SF-induced events or pacing events (atrial pacing (AP) or ventricular pacing (VP)) which are results of the injected SF.
- The "Without SF" timeline shows the expected natural progression of the cardiac cycles in the pacemaker.
- The "With SF" timeline shows the progression of the cardiac cycles in the pacemaker side after the SF has affected the system.

A. SF-INDUCED PACEMAKER OVERSENSING

Oversensing is a phenomenon in which the pacemaker inappropriately recognizes external electrical signals and noise

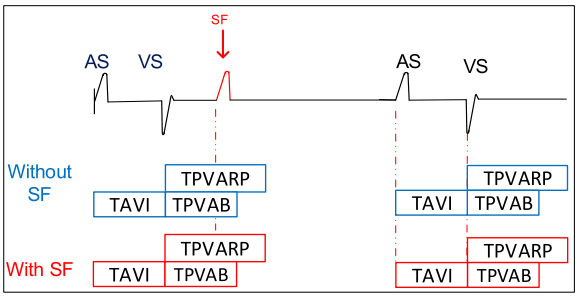


FIGURE 5. Timing diagram of SF at *Aget* during TPVAB.

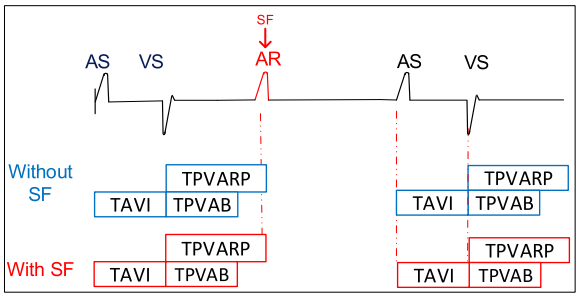


FIGURE 6. Timing diagram of an SF at *Aget* during TPVARP.

as native cardiac activity, and pacing is inhibited. Normally, the main sources of oversensing are large P or T waves, skeletal muscle activity, and lead contact problems. Moreover, it is reported in [31] that most common sources of electromagnetic interference (such as cellular phones) may cause pacemaker oversensing. SF-induced oversensing may lead to a disruption in the pacemaker cycle, causing undesired behavior. Through clinical tests, Hurkmans *et al.* [32] has reported that ionizing radiation can cause different functional inconsistencies due to sensing interference in implantable cardiac pacemakers, even leading to complete loss of function in some cases. We were able to identify the following SFs scenarios which can lead to oversensing, based on commonly adopted pacing modes, described in [33]:

1) SF AT *Aget* DURING TPVAB

As explained in Section V, the PVARP component operates in the refractory period TPVAB to prevent the recognition of electrical signals (generated from P or T waves, skeletal muscle activity or lead contact problems) as a native cardiac activity. Additionally, an SF which is injected at input node *Aget* at time that is less than TPVAB is also considered as refractory noise. Therefore, the impact of the oversensing of such SF will be natively masked, as shown in Fig. 5.

2) SF AT *Aget* BEFORE TVARP

An oversenseing can happen in the second refractory period of the PVARP component. However, this SF will be masked if it arrives before TPVAB interval at the PVARP component. Another oversensing scenario that was considered happens if the input node *Aget* is affected by an SF during the period of time after TPVAB but before TVARP (i.e., $TAVI < T < (TLRI - TAVI)$). As shown in Fig. 6, this SF will be characterized as an *Aget* event which is then fed to the pacemaker as an *AR* event. In this case, the SF does not directly impact the pacemaker behavior, but it can impact the efficiency of the diagnosis algorithms.

3) SF AT *Vget* DURING TURI AND TVRP

Fig. 7 shows a scenario where multiple SFs occur during time interval $T < TURI$. In this scenario, we assume that all the SFs induce ventricular sensing signals in the system. Since the URI component limits the ventricular pacing rate in the

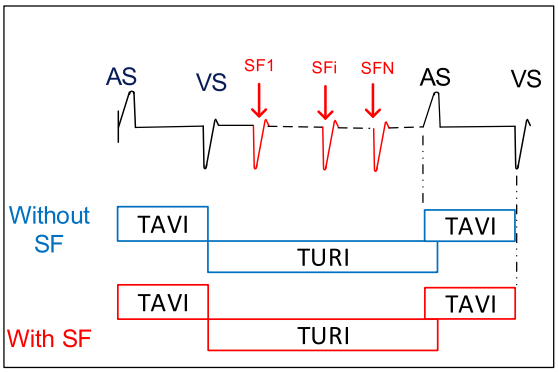


FIGURE 7. Timing diagram of the impact of SF during TURI.

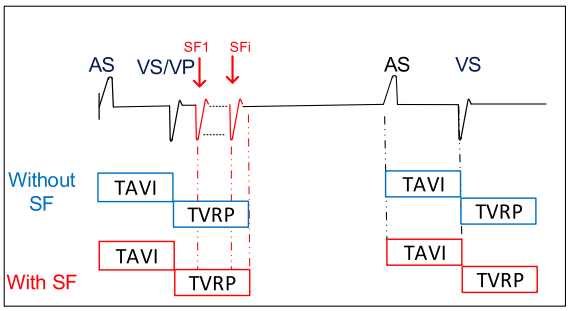


FIGURE 8. Timing diagram of the impact of SF during TVRP.

system, all ventricular oversensing induced by SFs during TURI are masked by the pacemaker. Therefore, such SFs do not have any impact on the pacemaker behavior. Similarly, Fig. 8 shows a scenario where multiple SFs are injected during time interval $T <= TVRP$. During this time interval, the SFs are interpreted as refractory noise waves and are masked by the system.

4) SF AT *Aget* AFTER TVARP

Another possible oversensing scenario is when an SF is injected at input node *Aget* at a time after TVARP. This SF can have an impact on the pacemaker behavior. This SF will be characterized as an actual *AS* event. As shown in Fig. 9, once this SF is characterized as an *AS*, then the pacemaker resets the clock and starts waiting for a *VS*, for a time interval of TAVI. This SF will have two implications: 1) it can mask the

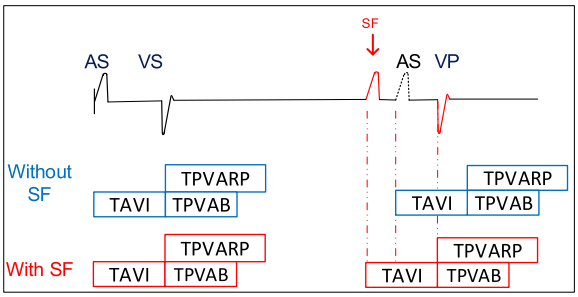


FIGURE 9. Timing diagram of SF-induced oversensing.

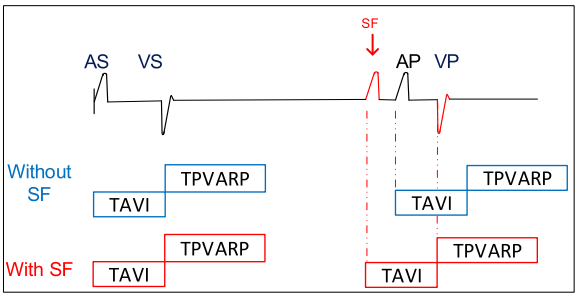


FIGURE 11. SF-induced undersensing.

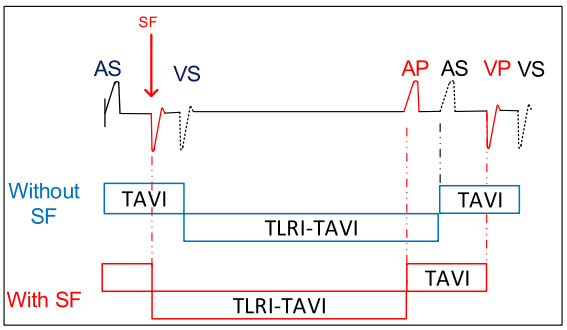


FIGURE 10. Timing diagram of SF-induced oversensing.

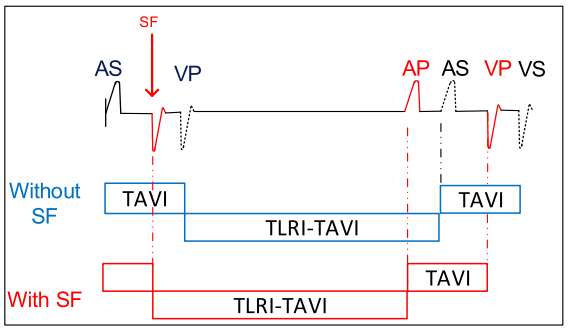


FIGURE 12. Output failure due to missed ventricular pacing.

actual *AS* that is released from the heart during TAVI. Shortly after this *SF*, the real atrial event occurs but it will be filtered because the pacemaker is expecting a ventricular event after *T*=*TAVI*; and 2) it will impact the behavior of the pacemaker for the next A-V cycles. This happens after the pacemaker waits for the time TAVI without the heart releasing the *VS*. Therefore, the pacemaker will have to release a ventricular pacing (VP). This pacing is not required in the normal operation and will affect the *cardiac* cycle and may lead to an arrhythmia, among other issues.

5) SF AT *Vget* WITHIN TAVI

Fig. 10 depicts a possible SF injection scenario that can lead to *oversensing* at the input node *Vget* within TAVI. In this scenario, after an atrial sensing or pacing event, the pacemaker is affected by an SF which is interpreted as a ventricular sensing before the real ventricular events happen in the heart. After the time period $T = TLRI - TAVI$, the pacemaker will be expecting to sense an atrial event. The absence of the atrial sensing, due to the fact that the system's clock is ahead of time, will cause the pacemaker to perform an erroneous atrial pacing, followed by another ventricular pacing as shown in Fig. 10.

B. SF-INDUCED PACEMAKER UNDERSENSING

Undersensing is the failure to sense, and it occurs when the pacemaker fails to recognize spontaneous myocardial depolarization. In other words, the pacemaker fails to sense native cardiac activity. One possible scenario of an SF-induced undersensing is shown in Fig. 11. In this scenario, an *SF* is injected during time interval $TPVARP < T < (TLRI - TAVI)$. This *SF* is interpreted by the pacemaker as an atrial sensing. However, in this scenario, the SF occurrence causes the atrial activity to pass undetected. The issue is further aggravated since the pacemaker registers this SF as a *AS* signal and starts waiting for the ventricular event at the *VS* signal. After the time period *TAVI*, a ventricular pacing is erroneously applied. Therefore, the pacemaker sends an inappropriate pacing pulse to the heart. Clinically, this is normally recognized by the generation of unnecessary pacing signals and can lead to skipped beats or palpitations, among other cardiac issues [31]. Several malfunctions related to pacemaker sensing are described in [34].

C. SF-INDUCED OUTPUT FAILURE

Output failure of a pacemaker occurs when an expected pacing stimulus is not generated. In the literature, multiple causes of output failure are identified including *oversensing* (section VIII-A), pacemaker runaway, lead displacement, and electrical interference. In Fig. 12, we construct a scenario where the incidence of an SF produces an electrical signal that is interpreted by the pacemaker as an early ventricular sensing. These effects have been reported by different researchers in the literature, in works such as [32] and [33]. Output failure due to an ionizing radiation is a major concern, especially in devices with low battery charge (e.g. low battery voltage due to overdue pacemaker replacement [34]). An example of this phenomenon is shown in Fig. 12. In this example, the injected soft-fault will prevent the pacemaker from generating the necessary ventricular pacing. Furthermore, this SF will impact the pacemaker behavior in the next cycles. For

instance, the pacemaker will be expecting an atrial event after time *TLRI-TAVI* and the pacemaker performs an atrial pacing on the heart when the AS is not sensed. Thus, the pacemaker will apply the pacing to the wrong heart chamber.

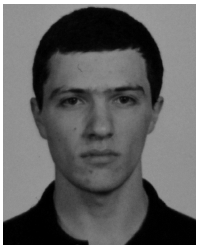
IX. CONCLUSION

In this paper, we proposed a new methodology for quantitative and automated verification of the impact of SFs on the behavior of the DDD pacemaker at the system-level. The correctness of PTA implementation is proven through model checking of a set of PCTL properties, defined based on the specifications and in agreement with the literature. We introduced a new approach to inject soft faults at certain time windows in the pacemaker model and construct an extended PTA model of the SF propagation. The proposed modeling and analysis were performed using the *Storm* probabilistic model checker. New insights on the SEU-induced malfunctions of pacemakers, such as *oversensing*, *undersensing*, and *output failure* are provided. The results of this analysis can be very useful towards improving the tolerance of the DDD pacemaker to soft-faults, by providing the necessary insight to help mitigating detected malfunctions.

REFERENCES

[1] H. G. Mond and A. Proclemer, "The 11th world survey of cardiac pacing and implantable cardioverter-defibrillators: Calendar year 2009—A world society of arrhythmia's project," *Pacing Clin. Electrophysiol.*, vol. 34, no. 8, pp. 1013–1027, 2011.
[2] R. Beinart and S. Nazarian, "Effects of external electrical and magnetic fields on pacemakers and defibrillators: From engineering principles to clinical practice," *Circulation*, vol. 128, no. 25, pp. 2799–2809, 2013.
[3] S. L. Pinski and R. G. Trohman, "Interference in implanted cardiac devices, part II," *Pacing Clin. Electrophysiol.*, vol. 25, no. 10, pp. 1496–1509, 2002.
[4] T. Zaremba, A. R. Jakobsen, M. Søgaard, A. M. Thøgersen, and S. Riahi, "Radiotherapy in patients with pacemakers and implantable cardioverter defibrillators: A literature review," *EP Europace*, vol. 18, no. 4, pp. 479–491, 2016.
[5] A. M. Ferrick, N. Bernstein, A. Aizer, and L. Chinitz, "Cosmic radiation induced software electrical resets in ICDs during air travel," *Heart Rhythm*, vol. 5, no. 8, pp. 1201–1203, 2008.
[6] A. Trigano, G. Hubert, J. Marfaing, and K. Castellani, "Experimental study of neutron-induced soft errors in modern cardiac pacemakers," *J. Intervent. Cardiac Electrophysiol.*, vol. 33, no. 1, pp. 19–25, 2012.
[7] T. Zaremba *et al.*, "Risk of device malfunction in cancer patients with implantable cardiac device undergoing radiotherapy: A population-based cohort study," *Pacing Clin. Electrophysiol.*, vol. 38, no. 3, pp. 343–356, 2015.
[8] R. Bagur *et al.*, "Radiotherapy-induced cardiac implantable electronic device dysfunction in patients with cancer," *Amer. J. Cardiol.*, vol. 119, no. 2, pp. 284–289, 2016.
[9] H. Hashii *et al.*, "Comparison of the effects of high-energy photon beam irradiation (10 and 18 MV) on 2 types of implantable cardioverter-defibrillators," *Int. J. Radiat. Oncol. Biol. Phys.*, vol. 85, no. 3, pp. 840–845, 2013.
[10] P. D. Bradley and E. Normand, "Single event upsets in implantable cardioverter defibrillators," *IEEE Trans. Nucl. Sci.*, vol. 45, no. 6, pp. 2929–2940, Dec. 1998.
[11] T. Zaremba, A. R. Jakobsen, A. M. Thøgersen, L. Oddershede, and S. Riahi, "The effect of radiotherapy beam energy on modern cardiac devices: An *in vitro* study," *Europace*, vol. 16, no. 4, pp. 612–616, 2014.
[12] S. K. Souliman and J. Christie, "Pacemaker failure induced by radiotherapy," *Pacing Clin. Electrophysiol.*, vol. 17, no. 3, pp. 270–273, 1994.
[13] A. O. Gomes and M. V. M. Oliveira, "Formal specification of a cardiac pacing system," in *Proc. Int. Symp. Formal Methods*. Berlin, Germany: Springer, 2009, pp. 692–707.
[14] L. A. Tuan, M. C. Zheng, and Q. T. Tho, "Modeling and verification of safety critical systems: A case study on pacemaker," in *Proc. 4th Int. Conf. Secure Softw. Integr. Rel. Improvement*, 2010, pp. 23–32.
[15] Z. Jiang, M. Pajic, S. Moarref, R. Alur, and R. Mangharam, "Modeling and verification of a dual chamber implantable pacemaker," in *Proc. Int. Conf. Tools Algorithms Construct. Anal. Syst.* Berlin, Germany: Springer, 2012, pp. 188–203.
[16] Z. Jiang, M. Pajic, R. Alur, and R. Mangharam, "Closed-loop verification of medical devices with model abstraction and refinement," *Int. J. Softw. Tools Technol. Transf.*, vol. 16, no. 2, pp. 191–213, 2014.
[17] Z. Jiang, M. Pajic, and R. Mangharam, "Model-based closed-loop testing of implantable pacemakers," in *Proc. IEEE/ACM 2nd Int. Conf. Cyber-Phys. Syst.*, Apr. 2011, pp. 131–140.
[18] Z. Jiang, M. Pajic, and R. Mangharam, "Cyber–physical modeling of implantable cardiac medical devices," *Proc. IEEE*, vol. 100, no. 1, pp. 122–137, Jan. 2011.
[19] D. Crocker, "Perfect developer: A tool for object-oriented formal specification and refinement," Formal Methods Europe, Uppsala, Sweden, Tech. Rep., 2003.
[20] R. Alur and D. L. Dill, "A theory of timed automata," *Theory Comput. Scince*, vol. 126, no. 2, pp. 183–235, 1994.
[21] G. Behrmann *et al.*, "Uppaal 4.0," in *Proc. 3rd IEEE Int. Conf. Quant. Eval. Syst. (QEST)*, Sep. 2006, pp. 125–126.
[22] T. Chen, M. Diciolla, M. Kwiatkowska, and A. Mereacre, "Quantitative verification of implantable cardiac pacemakers," in *Proc. IEEE 33rd Real-Time Syst. Symp. (RTSS)*, Dec. 2012, pp. 263–272.
[23] G. D. Clifford, S. Nemati, and R. Sameni, "An artificial vector model for generating abnormal electrocardiographic rhythms," *Physiol. Meas.*, vol. 31, no. 5, pp. 595–609, 2010.
[24] C. Dehnert, S. Junges, J.-P. Katoen, and M. Volk, "A STORM is coming: A modern probabilistic model checker," in *Proc. Int. Conf. Comput. Aided Verification*. Cham, Switzerland: Springer, 2017, pp. 592–600.
[25] C. Baier and J.-P. Katoen, *Principles of Model Checking*. Cambridge, MA, USA: MIT Press, 2008.
[26] F. Ciesinski and M. Größer, "On probabilistic computation tree logic," in *Validation of Stochastic Systems*. Berlin, Germany: Springer, 2004, pp. 147–188.
[27] M. Kwiatkowska, G. Norman, and D. Parker, "PRISM 4.0: Verification of probabilistic real-time systems," in *Computer Aided Verification*. Berlin, Germany: Springer, 2011, pp. 585–591.
[28] A. Bianco and L. de Alfaro, "Model checking of probabilistic and nondeterministic systems," in *Proc. Int. Conf. Found. Softw. Technol. Theor. Comput. Sci.* Berlin, Germany: Springer, 1995, pp. 499–513.
[29] S. Shuja, S. K. Srinivasan, S. Jabeen, and D. Nawarathna, "A formal verification methodology for DDD mode pacemaker control programs," *J. Elect. Comput. Eng.*, vol. 2015, Aug. 2015, Art. no. 939028.
[30] Boston Scientific. *Reference Guide to Pacemakers, ICDs, and Leads*. Accessed: Oct. 2018. [Online]. Available: https://www.bostonscientific.com/content/dam/bostonscientific/quality/documents/MRG_Apr2011_approved.pdf
[31] R. M. Wachter, L. Goldman, and H. Hollander, *Hospital Medicine*. Baltimore, MD, USA: Williams & Wilkins, 2005.
[32] C. W. Hurkmans, E. Scheepers, B. G. Springorum, and H. Uiterwaal, "Influence of radiotherapy on the latest generation of implantable cardioverter-defibrillators," *Int. J. Radiat. Oncol., Biol., Phys.*, vol. 63, no. 1, pp. 282–289, 2005.
[33] A. G. Rapsang and P. Bhattacharyya, "Pacemakers and implantable cardioverter defibrillators—General and anesthetic considerations," *Brazilian J. Anesthesiol.*, vol. 64, no. 3, pp. 205–214, 2014.
[34] D. F. Ortega *et al.*, "Runaway pacemaker: A forgotten phenomenon?" *EP Europace*, vol. 7, no. 6, pp. 592–597, 2005.

GHAITH BANY HAMAD received the B.Sc. degrees in electrical and computer engineering from Hashemite University, Jordan, in 2009, the M.A.Sc. degree in electrical and computer engineering from Concordia University, Montreal, QC, Canada, in 2011, and the Ph.D. degree from the Electrical Engineering Department, Polytechnique Montréal, Montreal, in 2017. His research interest includes multi-level reliability analysis, cyber-physical systems, non-functional formal verification, and radiation effects.



MARWAN AMMAR received the B.S. degree in computer science and information systems from the Regional University of the State of Rio Grande do Sul, Ijui, Brazil, in 2008, and the M.S. degree in electrical and computer engineering from Concordia University, Montreal, QC, Canada, in 2011, where he is currently pursuing the Ph.D. degree in electrical and computer engineering. His research interest includes high-level modeling and analysis, formal verification techniques, and radiation effects on digital circuits.

OTMANE AIT MOHAMED (M'01) is currently a Professor with the Department of Electrical and Computer Engineering, Concordia University. His research interests include hardware verification, early reliability analysis of cyber-physical systems, and radiation effects on microelectronics systems.

He is also a member of the Concordia Hardware Verification Group, the Concordia Cyber Security Center, and Concordia Institute of Aerospace Design and Innovation, a long-lasting member of ACM, and an Executive Member of Microsystems Strategic Alliance of Québec (ReSMiQ). He was the General Chair and the Program Chair of several conferences, such as IEA-AIE 2018, CIIA 2018, and TOPHOL 2012. He also served as a reviewer for several related conferences and journals. He is also a Licensed Engineer with the province of Quebec.

YVON SAVARIA (S'77–M'86–SM'97–F'08) received the B.Ing. and M.Sc.A degrees in electrical engineering from École Polytechnique Montréal in 1980 and 1982, respectively, and the Ph.D. degree in electrical engineering from McGill University in 1985. Since 1985, he has been with Polytechnique Montréal, where he is currently a Professor with the Department of Electrical Engineering.

He has carried work in several areas related to microelectronic circuits and microsystems, such as testing, verification, validation, clocking methods, defect and fault tolerance, effects of radiation on electronics, high-speed interconnects and circuit design techniques, CAD methods, reconfigurable computing and applications of microelectronics to telecommunications, aerospace, image processing, video processing, radar signal processing, and digital signal processing acceleration. He is currently involved in several projects that relate to aircraft embedded systems, radiation effects on electronics, asynchronous circuits design and test, green IT, wireless sensor networks, virtual networks, machine learning, computational efficiency, and application specific architecture design. He has published 140 journal papers and 440 conference papers. He holds 16 patents. He was the thesis advisor of 160 graduate students who completed their studies.

Dr. Savaria was the Program Co-Chairman of NEWCAS 2018. He has been a consultant or was sponsored for carrying research by Bombardier, CNRC, Design Workshop, DREO, Ericsson, Genesis, Gennum, Huawei, Hyperchip, ISR, Kaloom, LTRIM, Miranda, MiroTech, Nortel, Octasic, PMC-Sierra, Technocap, Thales, Tundra, and VXP. He is a member of the Regroupement Stratégique en Microélectronique duQuébec (RESMIQ) and the Ordre des Ingénieurs du Québec. He was a member of the CMC Microsystems Board. He was a Tier 1Canada Research Chair in design and architectures of advanced microelectronic systems from 2001 to 2015. He received the Synergy Award of the Natural Sciences and Engineering Research Council of Canada in 2006.