Ideal Mode Selection of a Cardiac Pacing System

Dominique Méry[1] and Neeraj Kumar Singh[2]

[1] Université de Lorraine, LORIA, BP 239, 54506 Vandœuvre-lès-Nancy, France
Dominique.Mery@loria.fr
[2] Department of Computer Science, University of York, United Kingdom
neeraj.singh@cs.york.ac.uk, Neerajkumar.Singh@loria.fr

Abstract. Mode transition in any inappropriate mode can be a common cause of any mishap in a complex health-care system. This paper presents an approach for formalizing and reasoning about optimal mode transition in a health-care system that uses several operating modes in various operating states. Modes are formalized and their relation to a state-based formalism is established through a refinement approach. The efficiency of this approach is presented by formalizing an ideal operating mode transition of a cardiac pacemaker case study. An incremental approach is used to develop the system and its detailed design is verified through a series of refinements. The consequence of this approach is to improve system structuring, elicitation of system assumptions and expected functionality, as well as requirement traceability using modes in state-based modeling. Models are expressed in EVENT B modeling language and validated by a model checker tool: ProB.

Keywords: Abstract model, Event-B, Proof-based development, Refinement, Modes, Pacemaker.

1 Introduction

The first permanent pacemaker was implanted in 1958 to a Swedish engineer at Karolinska Hospital, Sweden [9]. First generation of pacemaker focused on efficient performance and more programming modes to cover all kinds of pacemaker-dependent heart disease. For more than 50 years of research and development, there has been a wealth of different features incorporated into pacemaker design, including programmability, telemetry and different pacing modes.

Today, it is estimated that more than half a million people a year get pacemakers worldwide. Over the past 40 years the clinical indications for pacing have increased to include a large number of different cardiac arrhythmias. After several diagnosis tests and discussions, an ideal choice of pacemaker operating mode for an individual patient is considered that can provide the maximum benefits with as many features of normal sinus rhythm as possible [22].

Advanced technology pacemakers can automatically switch from one operating mode to other operating mode. Mode switching aims to achieve an appropriate ventricular rate during periods of atrial arrhythmias by the correct detection of premature atrial events and smooth transitions between atrial-tracking and non-tracking pacing modes [21].

V.G. Duffy (Ed.): DHM/HCII 2013, Part I, LNCS 8025, pp. 258–267, 2013.

Optimal and alternative pacemaker operating modes are fixed for certain diagnosis. If a pacemaker operates in an inappropriate operating mode, then cardiac death is likely. In order to select an inappropriate operating mode, different kinds of complication are possible: pacemaker syndrome [4]; inappropriate atrial tracking of atrial tachyarrhythmias by a DDD or VDD [1]; additional ventricular block in a patient with sinus node dysfunction may not be easily detected initially [2]; lack of physiological heart rate response on exercise [6].

In this paper, we propose mode a transition methodology to structure system specification to facilitate rigorous design and to assure that the system will never switch in an undesirable state. We use term mode in the same sense as [13,7] : both as partitions of the state space, representing different working conditions of the system, and as a way to define control information, structuring system operation. An assessment of the proposed approach is given through a case study, relative to the formal development of an ideal operating mode transition for pacing to allow the pacemaker response to behave as physiologically as possible. We develop an incremental refinement-based formal model of mode selection and prove that the pacemaker will never switch in any undesirable operating mode. In case of any fault, pacemaker either use the preselected alternative operating mode or automatically switch in any alternative operating mode.

The formal specification of an ideal mode transition of the pacemaker is developed in the EVENT B modeling language which is supported by the RODIN platform [18] and generated proof obligations are proved using RODIN proof tools. The main idea is to start with a very abstract model of the system which includes all operating modes of the pacemaker. Details are gradually added to this first model by building a sequence of more concrete events. The relationship between two successive models in this sequence is *refinement* [3,5]. The formal specification must be validated to ensure that they meet the interdisciplinary requirements of mode transition of a pacemaker. Hence, formal specification validation is carried out by both formal modeling and domain experts. Moreover, we use the ProB tool [14] to animate formal specification of mode selection of the pacemaker for analyzing and validating each refinement.

The outline of the remaining paper is as follows: Section 2 presents related works. Section 3 represents formal definition of the model system and state the required properties. In Section 4, we explore the stepwise formal development of an ideal operating mode transition in the pacemaker. Finally, Section 5 concludes the paper along with directions for future work.

2 Related Work

A *modal system* is a system characterized by *operation modes*, which coordinates system operations. Many systems are *modal systems*, for instance, space and avionic systems, steam boiler control, transportation and space system and so on. Operational modes denote all the different functional behaviors according to system requirements. Operation modes help to reason about system behaviors by focusing on system properties observed under different situations. In this approach, a system is seen as a set of modes partitioning the system functionality over different operating conditions. The term *assumption* is used to denote different operating conditions and *guarantee* denotes

the functionality ensured by the system under the corresponding assumption. A system may switch from one mode to another one in a number of ways characterized by mode transitions.

By analyzing requirements of pacemaker, we have found that the cardiac pacemaker system is also a *modal system* and it is based on *four-variable model* developed by Parnas and Madey [17]. According to the *four-variable model*, variables are continuous functions over the time and consist of *monitored* variables in the environment that the system responds to, *controlled* variables in the environment that the system is to control, *input* variables through which the software senses monitored variables, and *output* the variables through which the software changes the controlled variables. In the cardiac pacemaker system, *monitored* variables senses an intrinsic heart signal using pacemaker's sensors and *controlled* variables stimulate into the heart using pacemaker's actuator.

The use of operation modes is very common in real-time system. Every operation mode has specific time intervals for operating and modes are changing after a certain time bounds. Modecharts [13] focus on the specification of real-time properties of mode and mode switching. The authors have given the detailed information about the state space partition, various working conditions of the system and define the control information in large state machines. However, modecharts lacks adequate support to specifying and reasoning about functional properties. Some papers [19,10] have also addressed the problem of mode changing in real time system. Dotti et al. [7] have proposed both formalization and a refinement notion for *modal systems*, using existing support for the construction of *modal systems*.

H.D. Macedo, et al. [15] have developed a partial distributed real-time model of a cardiac pacing system using VDM. Gomes et al [11] have developed a formal specification of the pacemaker system using the Z modeling language. According to the paper, they have modelled the sequential model similar to H.D. Macedo et al. works [15]. Recently, a complete formal development of one and two-electrode pacemakers are presented in by D. Méry, et al. [16].

According to our literature survey, the cardiac pacemaker system is a *modal system* and none of the existing approaches for formalizing operating modes of the cardiac pacemaker system has used the refinement approach. We have used both formalization and refinement of a *modal system* for developing a formal specification of bradycardia operating modes of the cardiac pacemaker system using EVENT B modeling language.

3 Operation Modes

In multi-moded systems, the system consists of several operating modes. Each mode produces a different behavior, characterized by a set of functionalities that are carried out by different task sets. Using operating modes in multi-moded systems design offers two key advantages. First, it breaks up the complexity of a system into smaller pieces, making its specification easier. Second, an operating mode groups only those functions necessary in the actual situation. All other tasks are inactive in that mode. Different operating conditions are partitioning the set of modes of the system. Different operating conditions are used to represent *assumption* and *guarantee* denotes the functionality

ensured by the system under the corresponding assumption. A system may switch from one mode to another in a number of ways characterised by *mode transition* [7].

A pair Γ/Δ is used to characterize a mode, where $\Gamma(\alpha)$ is an assumption predicate over the current system state and $\Delta(\alpha, \alpha')$ is a relation over the current and next states as the guarantee of the system. A set of variables α is characterising a system state and constrained by an invariant $I(\alpha)$. An invariant $I(\alpha)$ is to limit the possible states by excluding undesirable states. In this paper, we have assumed that a system is working in only in one mode at a time. We are not considering here multiple operating modes case such as mode overlapping and mode interference. It is an interesting challenge that cannot be sufficiently addressed in this paper due to space limitations. Following formal representation presents that mode assumptions (Γ) are mutually exclusive and exhaustive in respect to a model invariant. $\oplus$ is a set partitioning operator.

$$I(\alpha) = \Gamma_1(\alpha) \oplus \ldots \oplus \Gamma_n(\alpha) \tag{1}$$

A mode transition is an atomic step to switch from one mode to other mode. It is convenient to characterise a mode transition by a pair of assumptions. Assuming that mode is assigned an index, a mode transition from Γ_p/Δ_p to Γ_q/Δ_q is a relation on mode indices $p \rightsquigarrow q$. In general, a system has an initial transition $\top \rightsquigarrow r$ and it terminates by terminating transitions $t \rightsquigarrow \bot$ after switches into some system mode Γ_r/Δ_r. In mode transition system enters at least in one operation mode without switching $\top \rightsquigarrow \bot$.

There are restrictions on the way mode assumptions and guarantees are formulated. The states described by a guarantee must be wholly included into valid model states:

$$I(\alpha) \wedge \Gamma(\alpha) \wedge \Delta(\alpha, \alpha') \Rightarrow I(\alpha') \tag{2}$$

The assumption and guarantee of a mode must be noncontradictory. i.e. a mode should permit a concrete implementation:

$$\exists \alpha, \alpha'.(I(\alpha) \wedge \Gamma(\alpha) \Rightarrow \Delta(\alpha, \alpha')) \tag{3}$$

A system is characterised by a collection of modes and a vector of mode transitions:

$$\begin{gathered} \Gamma_1/\Delta_1, \ldots \Gamma_n/\Delta_n \\ p_1 \rightsquigarrow q_1, \ldots p_n \rightsquigarrow q_n \end{gathered} \tag{4}$$

An operation mode (m, α) represents the state of a system, where m is an index of current operation mode and α is current system state. Above discussed system can be understand as follows: when system is operating in mode m the state of model variables changes so that the next state is any state α' satisfying both the corresponding guarantee $\Delta(\alpha, \alpha')$ and the modes assumption $\Gamma(\alpha')$:

$$\frac{\Gamma_m(\alpha) \wedge \Delta_m(\alpha, \alpha') \wedge \Gamma_m(\alpha')}{\langle m, \alpha \rangle \rightarrow \langle m, \alpha' \rangle} \tag{5}$$

Equation 6 represents that if there is a mode transition starting from a current mode, the transition could be enabled to switch the system into a new operation mode.

$$\frac{m \rightsquigarrow n \wedge \Gamma_m(\alpha) \wedge \Gamma_n(\alpha')}{\langle m, \alpha \rangle \rightarrow \langle n, \alpha' \rangle} \tag{6}$$

Above given two activities (5 and 6) compete with each other: a non-deterministic choice is made between the two activities and an initiating transition must find an initial state without referring to any previous state.

$$\frac{\top \rightsquigarrow k \wedge \Gamma_k(\alpha)}{\langle \top, undef \rangle \rightarrow \langle k, \alpha \rangle} \tag{7}$$

In equation 7, $undef$ represents a system state before to the execution of an initial transition. System termination is handled by the above given switching rule. All three rules (5,6,7) assume that an invariant holds in current and new states: $I(\alpha) \wedge I(\alpha')$. This is a consequence of given conditions 1 and 3. Refinement technique is used to build operation mode based model by introducing new modes and transitions of the system. A various kind of refinement techniques can be used to develop the concrete model.

4 Formal Analysis of Ideal Modes

This section presents a refinement based formal specification of an ideal operating mode transition to allow the pacemaker response to behave as physiologically as possible. A paper [20] presents a set of diagnoses and corresponding optimal, alternative and inappropriate operating modes of a cardiac pacemaker. The optimal mode of pacing should be considered for most patients. Alternative modes should be regarded as being less satisfactory, but acceptable in some group of patients - for example those who are disabled by another disease, those with very intermittent symptoms, or those who have short life of expectancy because of another disease [20]. The bradycardia operating modes(see Table-1 are prescribed by ACC/AHA/NASPE and BPEG working committee [22].

Table 1. Bradycardia operating modes of a cardiac pacemaker system

Category	Chambers Paced	Chambers Sensed	Response to Sensing	Rate Modulation
Letters	**O**-None **A**-Atrium **V**-Ventricle **D**-Dual(A+V)	**O**-None **A**-Atrium **V**-Ventricle **D**-Dual(A+V)	**O**-None **T**-Triggered **I**-Inhibited **D**-Dual(T+I)	**R**-Rate Modulation

We begin by defining an EVENT B context in which we declare two new constants $Diagnosis$ and OP_Modes that represent an enumerated set of diagnoses and a set of possible operating modes, respectively. These constants are extracted from the articles [20,22].

$$axm1 : Diagnosis = \{SND, AVB, SND\_and\_AVB, CAF\_with\_AVB, CSS, MVVS, NO\_DISEASE\}$$
$$axm2 : OP\_Modes = \{AAI, AAIR, VVI, VDD, DDD, DDI, DDDR, DDIR, VVIR, OOO\}$$

Two new variables ($Diag\_OP$ and $diag$) are introduced in machine context. The variable $Diag\_OP$ is represented as a subset of optimal operating modes ($Diag\_OP \subseteq OP\_Modes$) and other variable $diag$ is represented as $diag \in Diagnosis$. Safety

properties of the system are represented by two new invariants $(inv1, inv2)$. The first property states that when heart has not any disease then pacemaker operates in OOO operating mode. According to the Table-1, in OOO operating mode pacemaker is an ideal state. The next safety property states that if any diagnosis is member of $Diagnosis \setminus \{NO\_DISEASE\}$ then operating mode $(Diag\_OP)$ becomes a subset of all possible operating modes (OP_Modes).

Two significant events $(Heart\_OK$ and $Heart\_KO)$ are introduced in the abstract model. The event $Heart\_KO$ represents that the heart is not OK and heart has any diagnosis condition. Guard $(grd1)$ of event $Heart\_KO$ states that a variable $heart\_dig$ is a member of $Diagnosis \setminus \{NO\_DISEASE\}$, then set of operating modes (OP_Modes) and heart diagnosis $(heart\_dig)$ are assigned in an atomic step to the variables $Diag\_OP$ and $diag$, respectively. The detection of a diagnosis and corresponding operating modes are made more specific in the refined layers. $Heart\_OK$ event models the non-detection of any diagnosis in the heart. The $Heart\_KO$ and $Heart\_OK$ events should be viewed together are modeling the possible outcome of a diagnosis which indicates either selection of required optimal operating modes or heart is OK.

$$
\begin{array}{l}
inv1 : diag = NO\_DISEASE \wedge \\
\quad Diag\_OP \neq \varnothing \\
\quad \Rightarrow \\
\quad Diag\_OP = \{OOO\} \\
\\
inv2 : diag \in Diagnosis \\
\quad \setminus \{NO\_DISEASE\} \\
\quad \Rightarrow \\
\quad Diag\_OP \subseteq OP\_Modes
\end{array}
$$

EVENT Heart_KO
ANY $heart\_dig$
WHERE
grd1 : $heart\_dig \in diagnosis \setminus \{NO\_DISEASE\}$
THEN
act1 : $Diag\_OP : |Diag\_OP' \subseteq OP\_Modes$
act2 : $diag := heart\_dig$
END

EVENT Heart_OK
ANY $heart\_dig$
WHERE
grd1 : $heart\_dig \notin diagnosis \setminus \{NO\_DISEASE\}$
THEN
act1 : $Diag\_OP := \{OOO\}$
act2 : $diag := heart\_dig$
END

4.1 First Refinement

In the abstract model, we have seen that diagnosis is detected in a single atomic step. This refinement level presents CSS and $MVVS$ diagnoses. A new event $Heart\_CSS\_MVVS$ is introducing in this refinement. The $Heart\_CSS\_MVVS$ event is a refinement of the abstract $Heart\_KO$ event and represents successful detection of CSS and $MVVS$ diagnoses. First invariant $(inv1)$ represents an optimal operating modes on the successful detection of CSS and $MVVS$ diagnoses. Second invariant $(inv2)$ states that the diagnoses $MVVS$ and CSS guarantees never select any inappropriate operating modes.

$$
\begin{array}{l}
inv1 : diag \in \{MVVS, CSS\} \wedge Diag\_OP \neq \varnothing \\
\quad \Rightarrow \\
\quad Diag\_OP = \{DDI\} \\
\\
inv2 : diag \in \{MVVS, CSS\} \wedge Diag\_OP \neq \varnothing \\
\quad \Rightarrow \\
\quad Diag\_OP \not\subseteq \{AAI, VDD, VVI\}
\end{array}
$$

EVENT Heart_CSS_MVVS
REF $Heart\_KO$
ANY $heart\_dig$
WHERE
grd1 : $heart\_dig \in \{CSS, MVVS\}$
THEN
act1 : $Diag\_OP := \{DDI\}$
act2 : $diag := heart\_dig$
END

4.2 Second Refinement

In the second refinement, two events $Heart\_MVVS$ and $Heart\_CSS$ represent a refinement of $Heart\_CSS\_MVVS$ event. This refinement level presents CSS and $MVVS$ diagnoses separately as a result of final detection of CSS and $MVVS$

diagnoses and their optimal operating modes. Other two events $Heart\_AVB$ and $Heart\_SND$ are also introduced in this level as refinement of $Heart\_KO$ event. We have given an example of a refined event $Heart\_SND$. Other event ($Heart\_AVB$) is also refined in a similar way. A set of new invariants ($inv1 - inv8$) are introduced as safety properties. These invariants state that the pacemaker uses optimal operating modes according to the diagnoses(CSS, MVVS, SND, AVB) and never switch in any inappropriate mode.

$inv1 : diag = CSS \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP = \{DDI\}$
$inv3 : diag = CSS \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP \not\subseteq \{AAI, VDD\}$
$inv4 : diag = MVVS \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP = \{DDI\}$
$inv2 : diag = MVVS \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP \not\subseteq \{AAI, VDD, VVI\}$
$inv5 : diag = SND \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP = \{AAIR\}$
$inv6 : diag = SND \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP \not\subseteq \{VVI, VDD\}$
$inv7 : diag = AVB \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP = \{DDD\}$
$inv8 : diag = AVB \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP \not\subseteq \{AAI, DDI\}$

EVENT Heart_SND
REF $Heart\_KO$
ANY $heart\_dig$
WHERE
grd1 : $heart\_dig = SND$
THEN
act1 : $Diag\_OP := \{AAIR\}$
act2 : $diag := heart\_dig$
END

4.3 Third Refinement

This refinement level presents as similar to the previous two refinements. Two events $Heart\_Cr\_AF\_with\_AVB$ and $Heart\_SND\_and\_AVB$ are introduced as a refinement of the event $Heart\_KO$, which are used to detect the CAF_with_AVB and SND_with_AVB diagnoses. Some new invariants ($inv1 - inv4$) are introduced as safety properties. These invariants state that the diagnosis SND_with_AVB and CAF_with_AVB are used always optimal operating modes and never operate in any inappropriate modes.

$inv1 : diag = SND\_and\_AVB \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP = \{DDDR, DDIR\}$
$inv2 : diag = SND\_and\_AVB \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP \not\subseteq \{AAI, VVI\}$
$inv3 : diag = CAF\_with\_AVB \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP = \{VVIR\}$
$inv4 : diag = CAF\_with\_AVB \wedge Diag\_OP \neq \varnothing \Rightarrow Diag\_OP \not\subseteq \{AAI, DDD, VDD\}$

4.4 Fourth Refinement

This is the final refinement of the system. A new variable alternative operating modes ($Diag\_OP\_Alter$) is defined as a subset of bradicardia operating modes in $inv1$. We introduce the alternative operating modes [20] in all events for each diagnosis. New guards and actions are introduced for modeling alternative operating modes in all events of the last refinement. No any new events are introduced in this refinement level. A set of new invariants ($inv2 - inv7$) is introduced for checking desired behavior of the optimal and alternative operating modes transition. New introduced invariants generate proof obligations. Generated proof obligations are discharged by RODIN proof tool and state that the pacemaker system never use any inappropriate mode for all given diagnoses.

$inv1 : Diag\_OP\_Alter \subseteq OP\_Modes$
$inv2 : diag = SND \wedge Diag\_OP \neq \varnothing \Rightarrow (Diag\_OP = \{AAIR\} \wedge$
$Diag\_OP\_Alter = \{AAI\} \wedge Diag\_OP \not\subseteq \{VVI, VDD\} \wedge Diag\_OP\_Alter \not\subseteq \{VVI, VDD\})$
$inv3 : diag = AVB \wedge Diag\_OP \neq \varnothing \Rightarrow (Diag\_OP = \{DDD\} \wedge$
$Diag\_OP\_Alter = \{VDD\} \wedge Diag\_OP \not\subseteq \{AAI, DDI\} \wedge Diag\_OP\_Alter \not\subseteq \{AAI, DDI\})$
$inv4 : diag = SND\_and\_AVB \wedge Diag\_OP \neq \varnothing \Rightarrow (Diag\_OP = \{DDDR, DDIR\} \wedge$
$Diag\_OP\_Alter = \{DDD, DDI\} \wedge Diag\_OP \not\subseteq \{AAI, VVI\} \wedge Diag\_OP\_Alter \not\subseteq \{AAI, VVI\})$
$inv5 : diag = CAF\_with\_AVB \wedge Diag\_OP \neq \varnothing \Rightarrow (Diag\_OP = \{VVIR\} \wedge Diag\_OP\_Alter = \{VVI\} \wedge$
$Diag\_OP \not\subseteq \{AAI, DDD, VDD\} \wedge Diag\_OP\_Alter \not\subseteq \{AAI, DDD, VDD\})$
$inv6 : diag = CSS \wedge Diag\_OP \neq \varnothing \Rightarrow (Diag\_OP = \{DDI\} \wedge Diag\_OP\_Alter = \{DDD, VVI\} \wedge$
$Diag\_OP \not\subseteq \{AAI, VDD\} \wedge Diag\_OP\_Alter \not\subseteq \{AAI, VDD\})$
$inv7 : diag = MVVS \wedge Diag\_OP \neq \varnothing \Rightarrow (Diag\_OP = \{DDI\} \wedge Diag\_OP\_Alter = \{DDD\} \wedge$
$Diag\_OP \not\subseteq \{AAI, VVI, VDD\} \wedge Diag\_OP\_Alter \not\subseteq \{AAI, VVI, VDD\})$

We have described here only summary informations about each refinement in form of very basic description of the operation mode handling of a cardiac pacemaker using incremental refinement-based approach and omit detailed formalisation of events and proof details due to limited space. To find complete formal representation of an ideal operating mode transition in the cardiac pacemaker see[1].

4.5 Validation of Models

There are two main validation activities in EVENT B ; *consistency checking* and *model analysis*. Both are complementary for designing a consistent system. This section conveys the validity of the model by using a ProB tool [14] and Proof Statistics. "Validation" refers to the activity of gaining confidence that the developed formal models are consistent with the requirements, which have been extracted from the articles [20,22]. We have used the ProB tool [14] that supports *automated consistency checking* of EVENT B machines via model checking [8] and constraint-based checking [12]. Animation using ProB worked very well and we have then used ProB to validate the EVENT B formal specification according to the desired behavior of the system. We have validated the complete formal specification of an ideal operating mode transition in the pacemaker.

The Table-2 is expressing the proof statistics of the development in the RODIN tool. These statistics measure the size of the model, the proof obligations generated and discharged by the RODIN prover, and those are interactively proved.

Table 2. Proof Statistics

Model	Total number of POs	Automatic Proof	Interactive Proof
Abstract Model	8	8(100%)	0(0%)
First Refinement	10	6(60%)	4(40%)
Second Refinement	61	60(98%)	1(2%)
Third Refinement	34	33(97%)	1(3%)
Fourth Refinement	48	37(77%)	11(23%)
Total	161	144(89%)	17(11%)

The complete development of an ideal operating mode selection or automatic mode transition in the pacemaker system results in 161(100%) proof obligations, in which 144(89%) are proved automatically by the RODIN tool. The remaining 17(11%) proof obligations are proved interactively using RODIN tool. In order to guarantee the correctness of the system, we have established various invariants in stepwise refinement. Most of the proofs are automatically discharged. It should be noted that the manual proofs were not difficult. Proofs are quite simple, and achieved with the help of a simple *click* operation. The stepwise refinement of the system helps to achieve a high degree of automatic proof.

5 Conclusion and Future Works

In this paper the notions of modes transition are formally defined. These notions allow explicit characterization of various system conditions, through expressing assumptions,

[1] Available at
`http://www.loria.fr/~singhnne/mywork/opmode/OpModePacemaker.pdf`

and the properties of the system working under such conditions, through the use of guarantees. The complexity of design is reduced by structuring systems using modes and by detailing this design using refinement. This approach makes it easier for the developers to map requirements to models and to trace requirements. For quick understanding, we have applied this mode transition methodology for formalizing and reasoning about optimal mode transition using a cardiac pacemaker case study. The formal model has covered the general diagnoses and their optimal and alternative operating modes [20,22] of the pacemaker system. This case study indicates that inappropriate mode transition can be cause of many problems (see Section 1). It also suggests that such an approach can yield a viable model that can be useful for validation and correct selection of optimal modes for any diagnosis. More precisely, we have presented a formal development of optimal mode transition and prove it that it will never switch in any inappropriate operating mode. This proposed technique intend to assist in the design process of system where correctness and safety are important issues.

We have outlined how an incremental refinement approach to the ideal mode transition system allows us to achieve a very high degree of automatic proofs using RODIN tool. The approach we have taken is not specific to Event-B. We believe a similar approach could be taken using other state-based notations such as ASM, TLA or Z. RODIN proof tool was used to generate the hundreds of proof obligations and to discharge those obligations automatically and interactively. Another key role of the tool was in helping us to discover appropriate gluing invariants to prove the refinements. Without this level of automated support, making the changes to the refinement chain that we did make would have been far too tedious. In summary some key lessons are that incremental development with small refinement steps, appropriate abstractions at each level and powerful tool support are all invaluable in this kind of formal development. Finally, we have validated the mode selection system using the ProB model checker as validation tool and verified the correctness of desired behavior of the system according the medical domain experts.

In the future, we have planned to extend this mode transition methodology for real time system and apply on very high sensitive hybrid systems: avionic, cruise control, atomic plants, medical devices and so on. All kind of hybrid systems use several operating modes in real time where an inappropriate mode transition is a very common cause of mishap.

Acknowledgement. Neeraj Kumar Singh was supported by grant awarded by the Ministry of University and Research.

References

1. Castellanos Jr., A., Lemberg, L., Rodriguez-Tocker, L., Berkovits, B.V.: Atrial synchronized pacemaker arrhythmias: revisited. Am. Heart, Pub. Med. 2, 199–208 (1968)
2. Steinbach, K., Forohner, K., Meisl, F.: Atrial stimulation. In: Perez Gomez, F. (ed.) Cardiac Pacing, p. 629 (1985)
3. Abrial, J.-R.: Modeling in Event-B: System and Software Engineering. Cambridge University Press (2009) (forthcoming book)
4. Ausubel, K., Furman, S.: The Pacemaker Syndrome. Annals of Internal Medicine 103(3), 420–429 (1985)

5. Back, R.: On correct refinement of programs. Journal of Computer and System Sciences 23(1), 49–68 (1979)
6. Allen, A., Clarke, M.: Rate responsive atrial pacing resulting in pacemaker syndrome. PACE 10, 1209 (1987)
7. Dotti, F.L., Iliasov, A., Ribeiro, L., Romanovsky, A.: Modal systems: Specification, refinement and realisation. In: Breitman, K., Cavalcanti, A. (eds.) ICFEM 2009. LNCS, vol. 5885, pp. 601–619. Springer, Heidelberg (2009)
8. Grumberg, O., Clarke, E.M., Peled, D.: Model Checking. MIT Press (1999) ISBN 978-0262032704
9. Elmqvist Rune, S.A.: An implantable pacemaker for the heart. In: Medical Electronics. International Conference on Medical Electronics, vol. 2, pp. 253–254. Iliffe, London (1959)
10. Fohler, G.: Realizing changes of operational modes with a pre run-time scheduled hard real-time system. In: Proceedings of the Second International Workshop on Responsive Computer Systems, pp. 287–300. Springer (1992)
11. Gomes, A.O., Oliveira, M.V.M.: Formal specification of a cardiac pacing system. In: Cavalcanti, A., Dams, D.R. (eds.) FM 2009. LNCS, vol. 5850, pp. 692–707. Springer, Heidelberg (2009)
12. Jackson, D.: Alloy: a lightweight object modelling notation. ACM Trans. Softw. Eng. Methodol. 11(2), 256–290 (2002)
13. Jahanian, F., Mok, A.K.: Modechart: A specification language for real-time systems. IEEE Trans. Softw. Eng. 20(12), 933–947 (1994)
14. Leuschel, M., Butler, M.: Michael Leuschel and Michael Butler. In: Araki, K., Gnesi, S., Mandrioli, D. (eds.) FME 2003. LNCS, vol. 2805, pp. 855–874. Springer, Heidelberg (2003)
15. Macedo, H.D., Larsen, P.G., Fitzgerald, J.S.: Incremental Development of a Distributed Real-Time Model of a Cardiac Pacing System Using VDM. In: Cuellar, J., Sere, K. (eds.) FM 2008. LNCS, vol. 5014, pp. 181–197. Springer, Heidelberg (2008)
16. Méry, D., Singh, N.K.: Functional behavior of a cardiac pacing system. International Journal of Discrete Event Control Systems 1 (2010) (in Press)
17. Parnas, D.L., Madey, J.: Functional documents for computer systems. Sci. Comput. Program. 25(1), 41–61 (1995)
18. Project RODIN. Rigorous open development environment for complex systems (2004), `http://rodin-b-sharp.sourceforge.net/`
19. Real, J., Crespo, A.: Mode change protocols for real-time systems: A survey and a new proposal. Real-Time Syst. 26(2), 161–197 (2004)
20. Report. Recommendations for pacemaker prescription for symptomatic bradycardia. British Heart Journal 66(2),185–189 (1991)
21. Sutton, R., Stack, Z., Heaven, D., Ingram, A.: Mode switching for atrial tachyarrhythmias. The American Journal of Cardiology 83(5, suppl. 2), 202–210 (1999)
22. Epstein, A.E., DiMarco, J.P., Ellenbogen, K.A., Estes III, N.A.M., Freedman, R.A., Gettes, L.S., Gillinov, A.M., Gregoratos, G., Hammill, S.C., Hayes, D.L., Hlatky, M.A., Newby, L.K., Page, R.L., Schoenfeld, M.H., Silka, M.J., Stevenson, L.W., Sweeney, M.O.: ACC/AHA/HRS 2008 Guidelines for Device-Based Therapy of Cardiac Rhythm Abnormalities. Circulation, 117(21):2820–2840 (2008)