Iwona GROBELNA, Michał GROBELNY
UNIWERSYTET ZIELONOGÓRSKI, PODGÓRNA 50, 65-246 ZIELONA GÓRA

Inhibitor and enabling arcs in logic controller design

mgr inż. Iwona GROBELNA

Absolwentka Wydziału Elektrotechniki, Informatyki i Telekomunikacji Uniwersytetu Zielonogórskiego oraz Fachhochschule Giessen-Friedberg (Niemcy). Od marca 2008 zatrudniona na stanowisku asystenta w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego. Zainteresowania naukowe obejmują metody weryfikacji specyfikacji systemów osadzonych.
Członek Polskiego Towarzystwa Informatycznego.

e-mail: I.Grobelna@iie.uz.zgora.pl

mgr inż. Michał GROBELNY

W roku 2007 ukończył studia na Wydziale Elektrotechniki, Informatyki i Telekomunikacji Uniwersytetu Zielonogórskiego. Absolwent Zintegrowanych Studiów Zagranicznych Uniwersytetu Zielonogórskiego i Fachhochschule Giessen-Friedberg (Niemcy). Zainteresowania naukowe obejmują metody specyfikacji osadzonych systemów sterowania.
Członek Polskiego Towarzystwa Informatycznego.

e-mail: M.Grobelny@weit.uz.zgora.pl

Abstract

The article presents a novel approach to rule-based logic controller specification and its verification. Proposed abstract model is suited for formal verification (using model checking technique) as well as for logic synthesis (using hardware description language VHDL). Special focus is put on Interpreted Petri Nets with inhibitor and enabling arcs, their realization in rule-based model and their interpretation in another logic controller specification technique, namely UML Activity Diagrams (in version 2.x).

Keywords: logic controller specification, formal verification, Petri Nets inhibitor and enabling arcs, UML Activity Diagrams.

Łuki zakazujące i zezwalające w projektowaniu sterowników logicznych

Streszczenie

Artykuł przedstawia nowatorskie podejście do regułowej specyfikacji sterownika logicznego oraz jej weryfikacji (walidacji). Proponowany abstrakcyjny model logiczny jest dogodny zarówno do formalnej weryfikacji modelowej, jak również do syntezy logicznej (język opisu sprzętu VHDL). Szczególną uwagę poświęcono łukom zakazującym i zezwalającym interpretowanych sieci Petriego, ich realizacji w abstrakcyjnym modelu logicznym i interpretacji w innej postaci specyfikacji zachowania sterownika logicznego – diagramach aktywności języka UML (w wersji 2.x).

Słowa kluczowe: specyfikacja sterownika logicznego, formalna weryfikacja, łuki zakazujące i zezwalające sieci Petriego, diagramy aktywności języka UML.

1. Introduction

The process of logic controllers design [1] usually starts with specification. In this phase system properties and main functionality goals are specified. It is important, that the specification is verified (validated) [2] before implementation, as it allows to early discovering possible errors. Subtle errors in the specification may influence oncoming phases or even the whole venture, especially by dependable embedded logic controllers where requirements address beside high quality also reliability, availability, safety and secureness.

Behavior (functionality of designed system) may be formalized using various techniques [3], one of them are Petri Nets [4] [5] or UML Activity Diagrams [6].

Model checking technique [7] can be used to verify model description and check whether it satisfies some defined requirements.

The novel approach proposes to use a rule-based logical model presented at RTL-level, suitable both for formal verification (model checking) and for logical synthesis.

The article is structured as follows. Section 2 presents some basic information about specification of logic controllers, focusing especially on Petri Nets and UML Activity Diagrams. Section 3 illustrates proposed design system for (reconfigurable) logic controllers and introduces a rule-based logical model. Section 4 proposes interpretation of inhibitor and enabling arcs in UML Activity Diagrams. Section 5 presents novel approach to formal verification of logic controller programs with computer deduction in temporal logic and describes synthesis method, in form of rapid prototyping. The paper concludes with a short summary.

2. Specification of logic controller behaviour

In this section some background on logic controller specification techniques is provided, focusing especially on Petri Nets, Control Interpreted Petri Nets (Signal Interpreted Petri Nets) and UML Activity Diagrams (in version 2.x).

An example of Signal Interpreted Petri Net [8] with inhibitor and enabling arcs is presented in Fig. 1.

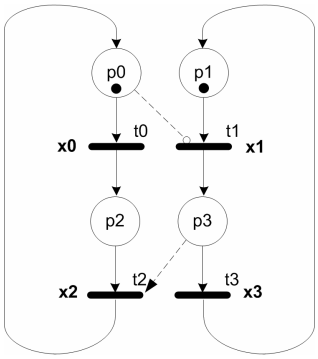

Fig. 1. A sample Signal Interpreted Petri Net [8]
Rys. 1. Przykładowa sieć Petriego typu SIPN [8]

HUMAN CAPITAL
HUMAN – BEST INVESTMENT!
Lubuskie
Worth your while
EUROPEAN UNION
EUROPEAN
SOCIAL FUND

Authors are scholars within Sub-measure 8.2.2 Regional Innovation Strategies, Measure 8.2 Transfer of knowledge,
Priority VIII Regional human resources for the economy Human Capital Operational Programme co-financed by European Social Fund and state budget.

3. System for logic controllers design and rule-based logical model

In this section schema of proposed system for logic controllers design is presented and described (Fig. 2).

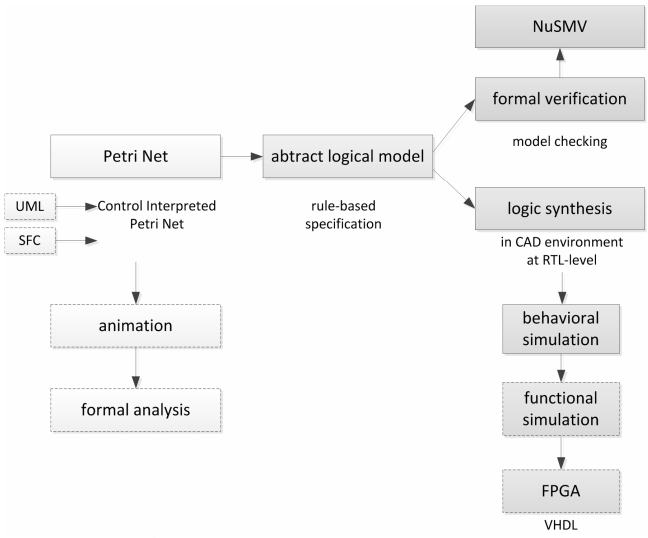

Fig. 2. Schema of proposed system
Rys. 2. Schemat proponowanego systemu

Rule-based logical model is introduced, focusing especially on Petri net inhibitor and enabling arcs interpretation.

4. Interpretation in UML Activity Diagrams

In this section interpretation of Petri net inhibitor and enabling arcs in UML Activity Diagrams is provided.

5. Formal verification and synthesis

In this section formal verification method of rule-based abstract logical model using model checking technique [9] is presented, together with its synthesis method (in form of rapid prototyping) [10].

6. Conclusions

Proposed novel approach to verification of embedded logic controller specification allows to detect some subtle errors on an early stage of system development. Rule-based representation in temporal logic is presented on RTL- level and is easy to formally verify using model checking technique and to synthesize using hardware description languages into reconfigurable logic controller.

The approach differs from the one presented in [8], where logic controller specification is verified in SPIN model checker [11]. Although the referred approach allows to verify Signal Interpreted Petri Nets, it does not guarantee that the resulting implementation will be valid according to primary specification. In the novel approach presented in the paper, an additional established rule-based logical model makes the specification abstract and more general, enabling as well formal verification as logic synthesis.

Results of the work include the assurance that verified behavioral specification in temporal logic will be an abstract program of matrix reconfigurable logic controller. Hence, logic controller program (its implementation) will be valid according to its primary specification. This may shorten the duration time of logic controllers development process (as early discovered errors are faster corrected) and, consequently, save money (as project budgets will not be exceeded).

7. References

[1] Adamski M., Karatkevich A., Wegrzyn M. (ed.): Design of embedded control systems. Springer 2005 (USA).
[2] Kropf T.: Introduction to Formal Hardware Verification. Berlin Heidelberg: Springer-Verlag, 1999.
[3] Gomes L., Barros J.P., Costa A.: Modeling formalisms for embedded system design. Embedded Systems Handbook, Taylor & Francis Group, 2006.
[4] David R., Alla H.: Discrete, Continuous, and Hybrid Petri Nets. Berlin Heidelberg: Springer-Verlag, 2010.
[5] Girault C., Valk R.: Petri Nets for Systems Engineering, A Guide to Modelling, Verification and Applications. Berlin Heidelberg: Springer-Verlag, 2003.
[6] Unified Modelling Language homepage: *www.uml.org*
[7] Clarke E. M., Grumberg O., Peled D. A.: Model checking. Cambridge, The MIT Press, 1999.
[8] Ribeiro Ó. R., Fernades J.M.: Translating synchronous Petri Nets into PROMELA for verifying behavioural properties. International Symposium on Industrial Embedded Systems, 2007, s. 266-273.
[9] Grobelna I., Adamski M.: Model checking of control interpreted Petri nets. Mixed Design of Integrated Circuits and Systems - MIXDES 2011 : proceedings of the 18th international conference, 2011, s. 621-626.
[10] Grobelna I.: Formal verification of embedded logic controller specification with computer deduction in temporal logic. Electrical Review, 2011, nr 12a, s. 47-50.
[11] SPIN model checker homepage: *http://spinroot.com*