

Michał GROBELNY

UNIwersytet Zielonogórski, Licealna 9, 65-417 ZIELONA GÓRA

Odwzorowanie hierarchicznych interpretowanych sieci Petriego sterowania z makromiejscami w diagramach aktywności UML

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). Od października 2011 zatrudniony na stanowisku asystenta w Katedrze Mediów i Technologii Informatycznych Uniwersytetu Zielonogórskiego. Zainteresowania naukowe obejmują metody specyfikacji osadzonych systemów sterowania. Członek Polskiego Towarzystwa Informatycznego.



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

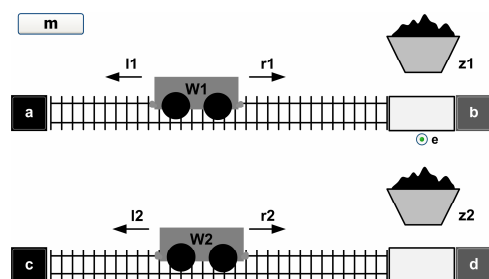
1. Wprowadzenie

Faza specyfikacji behawioralnej sterownika logicznego [1, 2] jest jednym z pierwszych etapów cyklu projektowego. Na tym etapie ustalany jest sposób pracy i zachowania projektowanego systemu. Specyfikacja może mieć postać nieformalną i formalną. Formalne metody specyfikacji graficznej mogą zostać wykonane przy wykorzystaniu wielu technik takich jak abstrakcyjne maszyny stanów ASM, sieci SFC, sieci Petriego [2, 3, 4] czy diagramy maszyny stanowej i diagramy aktywności UML [2, 5, 6, 7]. Każda z powyższych technologia ma swoje zalety oraz rzeszę zwolenników. Niemniej jednak może tak się zdarzyć, że w jednym z projektów zostanie użyta więcej niż jedna metoda specyfikacji. Fakt taki może wynikać chociażby z udziału więcej niż jednego zespołu rozproszonego po świecie wykorzystującego na co dzień różnych technologii. Pomocna w takiej sytuacji może być transformacja pomiędzy używanymi typami specyfikacji [5, 6].

Artykuł skupia się na dwóch z wcześniej wymienionych technologii, czyli na interpretowanych sieci Petriego sterowania oraz diagramach aktywności dostosowanych do opisu procesów sterowania. Coraz większa złożoność projektów sprawia, że projekty oparte są na metodach hierarchicznego przedstawiania zachowania systemu sterowania.

2. Proces sterowania i jego graficzna specyfikacja

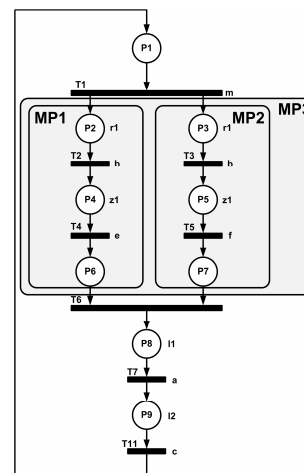
Odwzorowanie przedstawiono posługując się przykładem sterowania dwoma wózkami zaprezentowanym na rys. 1.



Rysunek 1. Rzeczywisty model omawianego procesu sterowania.
Figure 1. Real model of described control process.

Specyfikacja procesu sterowania przedstawiona została na rys. 2 z wykorzystaniem interpretowanej sieci Petriego

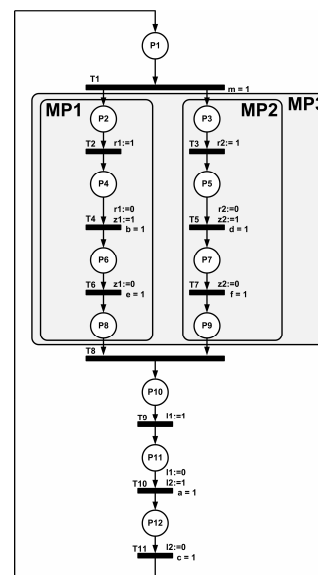
sterowania. Na diagramie przedstawione zostały możliwe makromiejscia MP1, MP2 i MP3.



Rysunek 2. Specyfikacja procesu sterowania z makromiejscami
Figure 2. Specification of control process with macroplaces

Ponadto w rozdziale przedstawiono proces sterowania oraz jego interpretację z wykorzystaniem sieci Petriego.

3. Zamiana wejść typu Moore'a na wejścia typu Mealy'ego



Rysunek 3. Specyfikacja procesu sterowania z wykorzystaniem wyjść typu Mealy'ego.
Figure 3. Specification of control process with usage of Mealy outputs.

Ta cecha diagramów aktywności wymusza wykorzystanie wyjść typu Mealy'ego oraz takie przekonstruowanie procesu sterowania aby wszystkie wyjścia były tego typu. W przypadku omawianego odwzorowania wspomniana zamiana wyjść z wyjść



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



Lubuskie

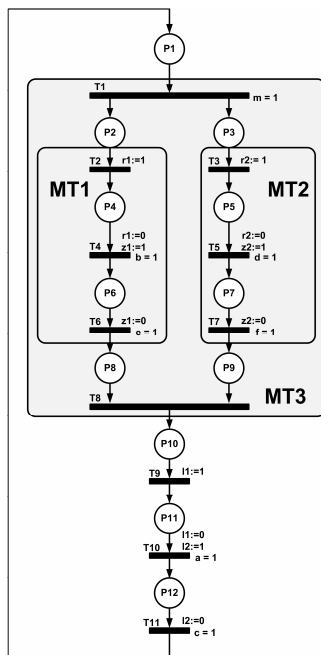
UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



typu Moore'a na wyjścia typu Mealy'ego realizowana jest na specyfikacji sieci Petriego. Przykładowa sieć Petriego po zmianie przedstawiona jest na rysunku 3. Niniejszy rozdział opisywał będzie szczegółowo proces zamiany miejsc. Ze względu na specyfikę diagramów aktywności UML, które przedstawiają akcje wykonywane przez proces nie ma możliwości reprezentacji wyjść typu Moore'a.

4. Zmiana makromiejsc w makrotranzycji

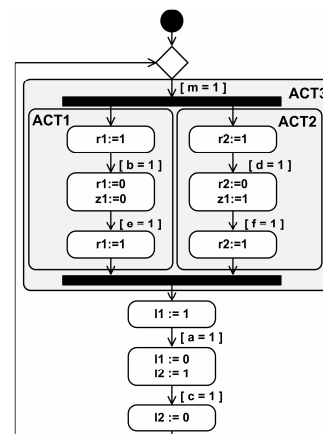
Kolejnym elementem koniecznym do odwzorowania procesów jest przypisanie makromiejscom odpowiadającym im makrotranzycji, które docelowo będą przetransformowane na aktywności sieci Petriego. Sieć Petriego z makrotranzycjami przedstawiona została na rysunku 4. Rozdział przedstawi szczegółowo omawiany proces.



Rysunek 4. Specyfikacja procesu sterowania z wykorzystaniem makrotranzycji
Figure 4. Specification of control process with use of macrotransitions.

5. Transformacja do diagramu aktywności UML

W rozdziale tym przedstawione zostaną zasady transformacji hierarchicznej interpretowanej sieci Petriego sterowania do hierarchicznego diagramu aktywności UML. Diagram aktywności dla omawianego procesu przedstawiony został na rysunku 5. Odwzorowanie bazuje na zasadach transformacji prostych (bez hierarchii) sieci Petriego do diagramów aktywności [5, 6], a dokładnie jest rozszerzeniem reguł transformacji.



Rysunek 5. Diagram aktywności UML dla omawianego procesu sterowania
Figure 5. UML activity diagram for described control process.

6. Wnioski

Specyfikacja behawioralna sterownika logicznego może zostać wykonana z użyciem wielu technik. Wybór techniki specyfikacji zależy od wielu czynników. Realna jest sytuacja, w której będą musiały współpracować dwa zespoły używające na co dzień różnych technik specyfikacji. Przykładowym połączeniem mogą być diagramy aktywności UML i interpretowane sieci Petriego sterowania. Proponowana metoda umożliwia współprojektowanie jednego systemu cyfrowego z jednoczesnym wykorzystaniem obu wymienionych technik. Dodatkową zaletą jest możliwość wykorzystywania hierarchicznych struktur pozwalających na lepsze usystematyzowanie złożonego procesu i podział jego na mniejsze podprocesy. Zastosowanie specyfikacji hierarchicznej jest też wysoce wskazane w przypadku rekonfigurowalnych sterowników logicznych, zwłaszcza jeżeli przewidziana jest częściowa rekonfiguracja. Zastosowanie odwzorowania sieci Petriego na diagramy aktywności umożliwi w takiej sytuacji sprawną wymianę informacji pomiędzy dwoma zespołami lub świadome łączne zastosowanie obu technik specyfikacji przez jeden zespół projektowy w celu efektywnego jednoczesnego wykorzystania zalet obu omawianych technik.

7. Literatura

- [1] M. Adamski, A. Barkalov, M. Wgrzyn (Eds.), Design of Digital Systems and Devices, Lecture Notes in Electrical Engineering, Vol. 79, Springer-Verlag Berlin Heidelberg 2011.
- [2] F. Basile, P. Chiachio, D. Del Grosso, Modelling automation systems by UML and Petri Nets, Proceedings of the 9th International Workshop on Discrete Event Systems, Goteborg, 2008, pp. 308 - 313.
- [3] R. David, H. Alla, "Discrete, Continuous, and Hybrid Petri Nets", Berlin Heidelberg: Springer Verlag, 2010.
- [4] C. Girault, R. Valk, "Petri Nets for Systems Engineering. A Guide to Modeling, Verification, and Applications", Berlin Heidelberg: Springer Verlag, 2003.
- [5] I. Grobelna, M. Grobelny, M. Adamski, "Petri Nets and activity diagrams in logic controller specification - transformation and verification", Mixed Design of Integrated Circuits and Systems, Proceedings of the 17th International Conference, 2010, pp. 607 - 612.
- [6] M. Grobelny, I. Grobelna, Diagramy aktywności języka UML i sieci Petriego w systemach sterowania binarnego od transformacji do weryfikacji, Pomiary Automatyka Kontrola, nr 10, 2010, pp. 1154 - 1158
- [7] Unified Modelling Language website: <http://www.uml.org>