

Agnieszka WĘGRZYN, Marek WĘGRZYN
UNIwersytet Zielonogórski, Instytut Informatyki i Elektroniki

Wybrane tekstowe formaty specyfikacji sieci Petriego opisujące algorytmy sterowania

Dr inż. Agnieszka WĘGRZYN

Adiunkt w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego. Zainteresowania naukowe obejmują: modelowanie i weryfikację układów sterowania opisanych sieciami Petriego, formalne metody specyfikacji algorytmów sterowania, bazy danych oraz aplikacje internetowe.



e-mail: a.wegrzyn@iie.uz.zgora.pl

Dr inż. Marek WĘGRZYN

Adiunkt w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego. Pełni funkcję kierownika Zakładu Inżynierii Komputerowej. Zainteresowania badawcze obejmują zagadnienia projektowania systemów cyfrowych, ze szczególnym uwzględnieniem logiki programowalnej i języków opisu sprzętu (VHDL i Verilog).



e-mail: M.Wegrzyn@iie.uz.zgora.pl

Streszczenie

Układy sterowania mogą być modelowane z wykorzystaniem sieci Petriego, gdyż w łatwy sposób można przedstawić za ich pomocą współbieżność. Sieci Petriego mogą być reprezentowane w sposób graficzny (graf dwudzielny) lub tekstowy. Istnieje wiele formatów tekstowych opisu sieci Petriego. Rozwój technologii przyczynił się do prowadzenia prac nad ujednoliceniem formatów tekstowych z wykorzystaniem języka XML – język PNML. W artykule został opisany rozwój formatów tekstowych specyfikacji sieci Petriego opisujących algorytmy sterowania.

Słowa kluczowe: sieci Petriego, sterowniki logiczne, PNSF2, PNSF3, PNML

Selected textual formats for specification of Petri net described logic controllers

Abstract

Discrete systems can be modeled by Petri nets, because Petri nets in easy way present concurrency. There are two method of specification of Petri net: graphical way (bipartite graph) and textual formats. There are a lot of textual format specifying Petri net. Development of new technologies contributes to standardization of textual formats using XML. In the paper development of textual format specifying Petri nets describing logic controllers, is presented.

Keywords: Petri net, logic controllers, PNSF2, PNSF3, PNML

1. Wstęp

W 1962 roku Carl Adam Petri swoją rozprawą doktorską zapoczątkował rozwój nowej teorii umożliwiającej badanie zjawisk współbieżnych [10]. Nosząca imię swojego twórcy teoria sieci Petriego stanowi obecnie obszerną i szybko rozwijającą się dziedzinę nauki [9].

Sieci Petriego są grafem dwudzielnym o dwóch rodzajach wierzchołków, zwanych miejscami i tranzycjami. Z wykorzystaniem sieci można w przejrzysty oraz prosty sposób przedstawiać zjawiska zachodzące współbieżnie [9].

Pomysł wykorzystania sieci Petriego w celu modelowania układów cyfrowych, kontrolerów zwłaszcza kontrolerów logicznych, pojawił się pod koniec lat siedemdziesiątych. Aparat formalny sieci Petriego w połączeniu z metodami logiki formalnej umożliwia weryfikację i syntezę reprogramowanego sterownika logicznego metodą systematyczną [1]. W odróżnieniu od klasycznego automatu sekwencyjnego, automat współbieżny znajduje się równocześnie w jednym lub kilku stanach wewnętrznych. Maksymalne zbiory równocześnie występujących stanów lokalnych definiują stany globalne automatu. Dowolny podzbiór równocześnie występujących stanów lokalnych nazywany jest stanem częściowym. W automatach współbieżnych rozpatruje się zamiast globalnych funkcji przejść lokalne relacje, wiążące ze sobą wewnętrzne stany częściowe, aktualne i następne, oraz odpowiednie stany wejść współbieżnych wyjść automatu. Interpretowana sieć Petriego jest obrazową formą przedstawienia automatu współbieżnego [1].

Miejsca sieci reprezentują stany lokalne sterownika, tranzycje ich zmiany. Wszystkie miejsca oznakowane w tym samym czasie, definiują stan globalny sterownika. Znaczniki wyznaczają, które stany sterownika są aktywne w danym czasie. Oznakowanie początkowe reprezentuje stan początkowy układu. Każda tranzycja typu rozwidlenie, tzn. tranzycja posiadająca więcej niż jedno miejsce wyjściowe, jest początkowym punktem dla równoległych procesów, natomiast każda tranzycja typu złączenie, tzn. tranzycja posiadająca więcej niż jedno miejsce wejściowe, synchronizuje procesy, które łączą się w tym punkcie [7].

Modele sieci Petriego odgrywają ważną rolę w projektowaniu informatycznych eksploatacji systemów informatycznych oraz w planowaniu i sterowaniu przepływem produkcji [1, 7]. Badania prowadzone w dziedzinie projektowania układów cyfrowych potwierdziły przydatność sieci Petriego do analizy, weryfikacji i syntezy tego typu układów [4, 11, 12].

W trakcie badań nad możliwościami modelowania i analizy układów współbieżnych z wykorzystaniem sieci Petriego powstało wiele tekstowych formatów opisu sieci - Petri Net Specification Format (PNSF) [7]. Podobną postać regułową wykorzystano w języku CONPAR [3]. Oba formaty posłużyły następnie do opracowania formatu rozszerzonego o możliwość opisu również kolorowanych sieci Petriego (PNSF2) [13]. Powstanie nowej technologii zapisu dokumentów (XML) dało możliwość automatycznej transformacji pomiędzy różnymi formami opisu sieci Petriego. Łącząc zalety wcześniejszych formatów, opracowany został format zapisu sieci Petriego z wykorzystaniem języka XML – PNSF3.

Prowadzone są również prace nad ujednoliceniem zapisu sieci Petriego. Powstały w ich wyniku format Petri Net Markup Language - PNML został przygotowany do standaryzacji [5, 8, 16]. Jednakże format ten specyfikuje jedynie część elementów sieci Petriego. W przypadku opisu układów sterowania niezbędne jest dodanie dodatkowych znaczników. Dlatego przygotowywany jest format specyfikacji sieci Petriego opisujących układy współbieżne, oparty na formacie PNML.

W kolejnych rozdziałach zostanie opisany rozwój formatów tekstowych specyfikacji sieci Petriego modelujących sterowniki logiczne.

2. Formaty tekstowe specyfikacji sieci Petriego

Opisując sieci Petriego w sposób tekstowy można wykorzystywać język regułowy. Opis regułowy w języku sekwentów Gentzena [1] stał się podstawą uproszczonej, tekstowej reprezentacji interpretowanych sieci Petriego. Dzięki temu istnieje możliwość formalnej weryfikacji i optymalizacji z wykorzystaniem aparatu logiki matematycznej. Format został nazywany PNSF [6]. Format PNSF opracowany przez Kozłowskiego posiada znaczne podobieństwo do reguł produkcji, stosowanych do reprezentacji wiedzy i będących stwierdzeniami o postaci typu *if ... then ...*. Rozszerzeniem formatu PNSF był CONPAR [3]. Różnicą między formatem CONPAR a PNSF było wprowadzenie elementów hierarchii.

W uzupełnionym formacie, nazwanym PNSF2, zorientowanym na formalną specyfikację sieci hierarchicznych, wprowadza

dzono możliwość definiowania modułów (*.macroplace*, *.macrotransition*). Inną zasadniczą różnicą między nową wersją, a jej poprzednikami polega na możliwości przedstawiania w PNSF2 interpretowanych, kolorowanych sieci P/T [2]. Kolorowana i interpretowana P/T sieć jest użytecznym podzbiorem sieci CPN [4]. Kolorowana P/T sieć została zaadoptowana do potrzeb syntezy sterowników logicznych typu ASLC (ang. Application Specific Logic Controller) [13]. Jest ona też efektywnym narzędziem specyfikacji i syntezy układów cyfrowych.

Język PNSF2 umożliwia łatwą specyfikację układu cyfrowego z wyjściami Moore'a i Mealy'ego (*.MooreOutputs* oraz *.MealyOutputs*), zarówno typu rejestrowego, jak i kombinacyjnego. W formacie PNSF2 rozróżnia się wyjścia kombinacyjne (*.comb_outputs*) i wyjścia rejestrowe (*.reg_outputs*). W opisie automatu współbieżnego podaje się listy wszystkich sygnałów wejściowych (*.inputs*), sygnałów wyjściowych (*.outputs*), miejsc sieci (*.places*), tranzycji (*.transitions*) oraz miejsc w oznakowaniu początkowym (*.marking*) (rys. 1).

```
.clock CLOCK
.inputs x1 x2 x3 x4 x5 x6 x7
.outputs y1 y2 y3 y4 y5 y6 y7 y8

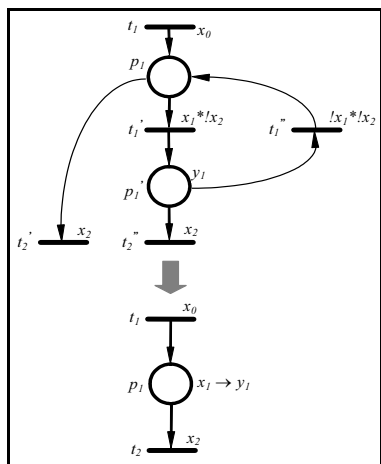
.part makrosiec4
.places d1=mp1(,) d2
.transitions t1 t7

.net
t1: d2 |- d1;
t7: d1 |- d2;
.marking d1
.endp

.macroplace mp1 (,)
interface d3 d4, d3 d4
.endm
...
.end
```

Rys. 1. Przykład specyfikacji sieci Petriego PNSF2
Fig. 1. Example of Petri net specification (PNSF2)

Rozszerzając język PNSF wyraźnie zadeklarowano wyjścia typu rejestrowego, które bardzo często zapisywane są równocześnie ze zmiennymi stanu, tym samym zboczem sygnału zegarowego. Z punktu widzenia struktury realizowanego zadania, rejestr wyjściowy traktowany jest jako zewnętrzny w stosunku do kontrolera. W formacie PNSF wszystkie wyjścia musiały być jednego typu, tzn. kombinacyjne albo rejestrowe. Szczególnie przy realizacji sterowników logicznych tego rodzaju ograniczenia są nie do przyjęcia. Wymagałoby to przekształcania standardowej formy zapisu (np. IEC-1131 [17]) na inną równoważną postać, często wbrew zamierzeniom projektanta. Dzięki deklaracji wyjść rejestrowych i kombinacyjnych, wyjścia typu Mealy'ego można deklarować zarówno z opisem tranzycji, jak i w oddzielnym module deklaracji (*.MealyOutputs*). W wielu przypadkach ułatwia to specyfikację projektu. Taki przypadek prezentowany jest na rys. 2.



Rys. 2. Przykład zbliżonych funkcjonalnie fragmentów sieci Petriego możliwych do specyfikacji w PNSF i PNSF2
Fig. 2. Example of functional approximately a part of Petri net specification in PNSF and PNSF2

PNSF2 dopuszcza możliwość bezpośredniej komunikacji dowolnego miejsca z wybranym miejscem z innego makromiejsca. Komunikacja taka odbywa się wyłącznie poprzez zewnętrzne luki zezwalające lub zabraniające. Zapisywane jest to poprzez podanie warunków zdefiniowanych w deklaracji globalnej. Takie rozwiązanie nie jest jednak zalecane, gdyż traci się wtedy zalety podejścia hierarchicznego do analizy sieci.

3. Formaty zapisu sieci Petriego oparte na języku XML

W ramach prac nad modelowaniem sterowników logicznych opisanych sieciami Petriego powstał format PNSF3 [14, 15]. Format ten w znacznym stopniu oparty jest na formacie PNSF2. Jest on również tekstowym formatem zapisu interpretowanej, synchronicznej, hierarchicznej i kolorowanej sieci Petriego [14, 15]. PNSF3 to zbiór reguł opisujących strukturę sieci Petriego. Reguły mają bezpośrednie odzwierciedlenie w elementach struktury sieci. Z wykorzystaniem formatu PNSF3 opisywany jest algorytm sterowania danego sterownika. Format PNSF3 pozwala na dokładne opisanie kolejnych elementów struktury - miejsc i tranzycji sieci oraz połączeń wewnątrz grafu. Przechowuje także informacje o sygnałach zegarowych, wejściowych i wyjściowych kontrolera i ich rodzaju. Ponadto, pozwala na opisanie warunków realizacji kolejnych tranzycji oraz następstw ich wykonania. Format PNSF3 nie przechowuje jednak informacji o sposobie graficznego rozmieszczenia elementów sieci w obrazie grafu.

Przedstawiony format PNSF3 posiada własny słownik i strukturę znaczników XML, wykorzystywanych do zapisu informacji o sieci. W dokumentacji formatu zdefiniowano szereg ogólnych znaczników grupujących pewne charakterystyczne informacje o modelu. Każdy zdefiniowany znacznik zawiera inne informacje i opisuje inny element grafu. Poszczególne znaczniki posiadają natomiast listę argumentów lub znaczników potomnych, umożliwiających dokładne wyspecyfikowanie informacji o danym elemencie grafu.

Bardzo ważnym zagadnieniem, przy tworzeniu nowego formatu, jest wyznaczenie pewnych zasad i ujednolicenie sposobu zapisu wszystkich tworzonego dokumentów w tym formacie. Wykorzystanie języka XML do zapisu sieci Petriego, znacznie ułatwia proces weryfikacji dokumentu, gdyż istnieje obecnie wiele parserów pozwalających na porównanie zawartości dokumentu z zawartością szablonu wzorcowego. Właśnie, dlatego, w celu poprawnej walidacji tworzonego dokumentu został opracowany zewnętrzny opis DTD definiujący strukturę, składnię i semantykę poszczególnych znaczników. Opis ten, definiuje także kolejność i liczbę powtórzeń znaczników potomnych oraz argumentów danego znacznika. Każdy poprawnie sformułowany dokument formatu PNSF3, powinien być zgodny z opisem DTD. Ogólny schemat poprawnego dokumentu przedstawia rys. 3.

```
<?xml version="1.0" encoding="ISO-8859-2"
standalone="no"?>
<!DOCTYPE PNSF3 SYSTEM "pnsf3.dtd">
<PNSF3>
  <GLOBAL>
    deklaracja sygnałów globalnych
  </GLOBAL>
  <MACRO_PLACE>
    deklaracja makromiejsca
  </MACRO_PLACE>
  <MACRO_TRANSITION>
    deklaracja makrotranzycji
  </MACRO_TRANSITION>
  <PART>
    deklaracja modułu sterownika
  </PART>
</PNSF3>
```

Rys. 3. Ogólny schemat dokumentu w formacie PNSF3
Fig. 3. General schema of PNSF3

Poprawny dokument formatu PNSF3, rozpoczyna się od instrukcji przetwarzania samego języka XML. Instrukcje te, definiują wersję języka, stronę kodową oraz ścieżkę dostępu do szablonu wzorcowego pnsf3.dtd. Po tych instrukcjach musi wystąpić główny znacznik <PNSF3> obejmujący klamrą zawartość całego dokumentu.

Jak przedstawiono na rys. 3 opis sieci Petriego składa się z czterech głównych bloków - znaczników:

<GLOBAL> - znacznik może wystąpić w dokumencie tylko jeden raz. Pozwala na zdefiniowanie sygnałów globalnych modelowanego układu, obowiązujących w zakresie całego opisu sieci Petriego.

<PART> - znacznik musi wystąpić w dokumencie przynajmniej jeden raz. Pozwala na zdefiniowanie pewnego pseudo-niezależnego modułu projektowanego układu. Każdy znacznik <PART> może opisywać cały projektowany układ lub tylko jego fragment. Znacznik ten pozwala na zdefiniowanie lokalnych sygnałów modelowanego układu, obowiązujących w zakresie danego modułu oraz pozwala na opis grafu połączeń sieci.

<MACRO_PLACE> - znacznik wykorzystywany jest do opisu hierarchicznych sieci Petriego. Występuje tylko wtedy, gdy w opisie nadrzędnego modułu <PART> wystąpi deklaracja makrowęzła. Znacznik ten, stanowi rozwinięcie opisu grafu wewnątrz danego makromiejsca. Znacznik ten pozwala na zdefiniowanie lokalnych sygnałów modelowanego układu, obowiązujących w zakresie danego makrowęzła oraz pozwala na opis grafu połączeń sieci. Wewnętrzne znaczniki potomne specyfikują identyczne parametry jak opisane dla znacznika <PART>. Ponadto dodatkowo za pomocą argumentów tego znacznika można zdefiniować: <MACRO_TRANSITION>.

Znacznik <MACRO_TRANSITION> wykorzystywany jest do opisu hierarchicznych sieci Petriego. Występuje tylko wtedy, gdy w opisie nadrzędnego modułu <PART> wystąpi deklaracja makrowęzła. Znacznik ten, stanowi rozwinięcie opisu grafu wewnątrz danej makrotranzycji. Pozwala on na zdefiniowanie lokalnych sygnałów modelowanego układu, obowiązujących w zakresie danego makrowęzła oraz pozwala na opis grafu połączeń sieci. Wewnętrzne znaczniki potomne specyfikują identyczne parametry jak opisane dla znacznika <PART>.

W formacie PNSF3 pominięte zostały atrybuty graficzne, co pozwala na ręczne przygotowywanie opisu modeli sieci, przy czym analiza składni takiego formatu jest łatwiejsza. Niestety takie podejście ma również wady. W celu przygotowania symulacji sieci w takim formacie należy dokonać konwersji na postać z atrybutami graficznymi.

W literaturze znane są już formaty opisu sieci Petriego w formacie XML [5, 8, 16], większość z nich opisuje atrybuty graficzne każdego elementu sieci. Przykładem takiego systemu jest Design/CPN opracowany na Uniwersytecie w Aarhus (Dania) [4]. Na podstawie istniejących formatów zapisu sieci w języku XML powstał projekt międzynarodowego standardu specyfikacji sieci Petriego w języku XML. Powstały format nosi nazwę PNML. Ponieważ istnieje wiele różnych klas oraz zastosowań sieci Petriego, format ten został on potraktowany jako koncepcja rdzenia dla wielu typów sieci Petriego. Pomysł utworzenia standardu powstał, aby umożliwić wymianę danych pomiędzy różnymi narzędziami wykorzystującymi sieci Petriego.

W formacie PNML przechowywana jest informacja o położeniu poszczególnych elementów sieci, co pozwala w łatwy sposób na transformację do formatu grafiki wektorowej SVG, który również zapisany jest w języku XML. Przedstawienie sieci za pomocą grafiki wektorowej pozwala na symulację sieci.

Aby zastosować standard PNML w modelowaniu sterowników logicznych z wykorzystaniem sieci Petriego należałoby dodać kolejne elementy, np. predykty, wyjścia z układu oraz rodzaj wyjść, itp.

Zalety wykorzystania języka XML w specyfikacji sieci Petriego są następujące:

- łatwość tworzenia i modyfikacji modeli w tym formacie;
- łatwość analizowania składni poprzez ogólnodostępne narzędzia;
- łatwa konwersja formatu do różnych reprezentacji;
- możliwość wymiany danych pomiędzy różnymi systemami;
- możliwość weryfikacji stworzonego modelu w różnych systemach;
- możliwość symulacji zamodelowanej sieci poprzez wykorzystanie formatu SVG.

4. Podsumowanie

Formaty PNSF2 i PNSF3 specyfikują działanie kontrolerów współbieżnych opisanych synchronicznymi interpretowanymi

sieciami Petriego w formie tekstowej. Każde miejsce sieci reprezentuje stan lokalny sterownika. Każda tranzycja reprezentuje zmianę stanu. Wszystkie miejsca oznaczone w danym momencie definiują stan globalny sterownika. Oznakowanie początkowe reprezentuje stan początkowy sterownika.

PNSF2 oraz PNSF3 opisują zarówno zachowanie kontrolera podane w formie synchronicznych interpretowanych sieci Petriego, jak również zawierają pewne szczegółowe informacje na temat implementacji sprzętowej.

Celem standaryzacji formatu specyfikacji sieci Petriego wymiana danych pomiędzy równymi systemami. Powstały format PNML jest jedynie koncepcją rdzenia dla wielu klas sieci. Aby wykorzystywać ten format do modelowania kontrolerów cyfrowych należy rozszerzyć go o dodatkowe znaczniki opisujące poszczególne elementy kontrolera, np. zegar, wejścia i wyjścia do i z układu.

Praca naukowa współfinansowana ze środków Komitetu Badań Naukowych w latach 2004-2006 jako projekt badawczy (grant nr 3 T11C 046 26).

5. Literatura

- [1] M.Adamski, Projektowanie układów cyfrowych systematyczną metodą strukturalną, Monografia Nr 49, Wydawnictwo Wyższej Szkoły Inżynierskiej w Zielonej Górze, Zielona Góra, 1990
- [2] M.Adamski, M.Węgrzyn, "Hierarchically Structured Coloured Petri Net Specification and Validation of Concurrent Controllers", Proceedings of the 39th International Scientific Colloquium, IWK'94, Ilmenau, Niemcy, 27-30.09.1994, Band 1, ss.517-522, 1994
- [3] J.M.Fernandes, M.Adamski, A.Proença, "VHDL Generation from Petri Net Specifications", IEE Proceedings - E, Computer and Digital Techniques, Vol.144, No.2, ss.127-137, March 1997
- [4] K.Jensen, Coloured Petri Nets. Basic Concept, Analysis Methods and Practical Use, Volume 1, Basic Concepts, Springer-Verlag, Berlin, 1992
- [5] E.Kindler, Definition, Implementation and Application of a Standard Interchange Format for Petri Nets, Proceedings of the Workshop on the Definition, Implementation and Application of a Standard Interchange Format for Petri Nets
- [6] T.Kozłowski, Petri-Net-Based CAD Tools for Parallel Controller Synthesis, M.Sc. Thesis, University of Bristol, Electrical and Electronic Engineering Department, Bristol, 1993
- [7] T.Kozłowski, E.L.Dagless, J.M.Saul, M.Adamski, J.Szajna, "Parallel controller synthesis using Petri nets", IEE Proceedings-E, Computers and Digital Techniques, Vol.142, No.4, ss.263-271, July 1995
- [8] R.B.Lyngso, T.Mailund, "Textual Interchange Format for High-Level Petri Nets", Workshop on Practical Use of Coloured Petri Nets and Design, June 1998, Aarhus University, ss.47-64
- [9] T.Murata, "Petri Nets: Properties, Analysis and Applications", Proceedings of the IEEE, Vol.77, No.4, ss.541-580, April 1989
- [10] C.A.Petri, Kommunikation mit automaten, Institut für Instrumentelle Mathematik, Bonn, 1962
- [11] W.Reisig, Sieci Petriego. Wprowadzenie, Wydawnictwa Naukowo-Techniczne, Warszawa, 1988
- [12] P.H.Starke, Sieci Petri - podstawy, zastosowania, teoria, PWN, Warszawa, 1987
- [13] M.Węgrzyn, Hierarchiczna implementacja współbieżnych kontrolerów cyfrowych z wykorzystaniem FPGA, Rozprawa doktorska, Politechnika Warszawska, Warszawa, 1998
- [14] A.Węgrzyn, Symboliczna analiza układów sterowania z wykorzystaniem wybranych metod analizy sieci Petriego, Rozprawa doktorska, Politechnika Warszawska, 2003
- [15] A.Węgrzyn, P.Bubacz, XML application for modelling and simulation of concurrent controllers, The International Workshop on Discrete-Event System Design, DESDes'01, Przytok, 27-29.06.2001 ss.215-221
- [16] Software and Systems Engineering - High-level Petri Nets, International Standard ISO/IEC 15909-2
- [17] International Electrotechnical Commission, International standard IEC 1131-3, Programmable Controllers, Part 3: Programming Languages, Geneva, 1992