

Piotr KAWALEC

POLITECHNIKA WARSZAWSKA, WYDZIAŁ TRANSPORTU
ul. Koszykowa 75, 00-662 Warszawa

Specjalizowane rekonfigurowalne sterowniki poziomu lokalnego w inteligentnych systemach transportowych ITS

Dr hab. inż. Piotr KAWALEC

Ukończył studia na kierunku elektronika, doktoryzował się w specjalności elementy i urządzenia techniki obliczeniowej i systemów sterowania, habilitował się na Politechnice Warszawskiej w specjalności sterowanie ruchem w transporcie. Pracuje na stanowisku profesora nadzwyczajnego na Wydziale Transportu Politechniki Warszawskiej, jego zainteresowania naukowe dotyczą metod modelowania i sterowania ruchem w transporcie z wykorzystaniem specjalizowanych systemów cyfrowych.



e-mail: pka@it.pw.edu.pl

Streszczenie

W artykule przedstawiono możliwość wykorzystania specjalizowanych sterowników ruchu drogowego, zaimplementowanych w układach FPGA, nie tylko na poziomie lokalnym, lecz również w sterowaniu obszarowym, a więc w zakresie właściwych ITS. Na przykładzie rzeczywistego obszaru sterowania, obejmującego kilka skrzyżowań, zaprezentowano proces specyfikacji, syntezy i implementacji takich sterowników w technice SoC. Dla konkretnego układu FPGA przytoczono raporty z wykorzystania zasobów i uzyskanych parametrów czasowych sterownika obszarowego.

Słowa kluczowe: sterowanie ruchem drogowym, sterowniki obszarowe, FPGA, SoC.

Specialized reconfigurable controllers of local level in ITS

Abstract

In contrast to industrial process control systems, the achievements of contemporary electronics of the last several years, such as working out hardware description language HDL, and accessibility of programmable logic devices FPGA have not been used in designing control devices of local level of ITS [4] up to the present moment. The paper presents an original methodology for design of new class local controllers of road traffic in railway traffic control implemented in programmable logic devices. The research results obtained at the Faculty of Transport (Warsaw University of Technology) of specialized prototypes of single-system reconfiguration controllers SoC show that the operating speed of such controllers exceeds the speed of currently used microprocessor controllers by hundreds of thousands times, with the improvement of reliability parameters by hundreds of times. It has been shown that the proposed class of specialized controllers can be used not only on the local level, but they can also be used for area controlling, encompassing several crossroads. The process of specification, synthesis and implementation of such a controller in SoC technology has been presented for a sample area of three neighboring crossroads (Figs. 1 and 2). The example presented shows that specialized reconfigured road traffic controllers implemented in FPGA (Fig. 3) devices can be used not only on the local level, but also on area controlling level, and so in ITS.

Keywords: control traffic, controllers area, FPGA, SoC.

1. Wstęp

W odróżnieniu od sterowania procesami przemysłowymi, w projektowaniu urządzeń sterowania ruchem w transporcie osiągnięcia współczesnej elektroniki takie jak języki opisu sprzętu HDL, oraz dostępność programowalnych struktur logicznych FPGA, są wykorzystywane w ograniczonym zakresie. Z drugiej strony realizacja tych urządzeń w postaci sterowników specjalizowanych, zwłaszcza w technice SoC, może stanowić istotną alternatywę dla stosowanych obecnie rozwiązań.

Uzyskane wyniki badań opracowanych na Wydziale Transportu PW prototypów specjalizowanych, rekonfigurowanych lokalnych

i obiektowych sterowników jednocukłowych SoC, wskazują, że szybkość działania takich sterowników w setki tysięcy razy przekracza szybkość działania stosowanych obecnie sterowników mikroprocesorowych, przy poprawie w setki razy parametrów niezawodnościowych [1, 2, 3].

Realizacja sterowników specjalizowanych dla pojedynczych skrzyżowań ulic, wykazała, że dobór konkretnej struktury reprogramowalnej determinowany jest liczbą niezbędnych wyprowadzeń IO ogólnego przeznaczenia, przy znikomym wykorzystaniu zasobów sprzętowych struktury [4]. Stwarza to szansę rozszerzenia wykorzystania specjalizowanych sterowników zrealizowanych w FPGA w technice SoC, nie tylko do sterowania na pojedynczych skrzyżowaniach, lecz również do sterowania w systemie obejmującym grupę skrzyżowań. Należy oczekiwać, że odpowiednie algorytmy sterowania będące połączeniem sterowania adaptacyjnego z pewnego rodzaju optymalizacją sieci, opierającą się na kryterium strat czasu, oraz długości kolejek, pozwolą na poprawę warunków ruchu w sterowanym obszarze miasta.

Takie sterowanie, nazywane sterowaniem obszarowym, stanowi zasadniczą część inteligentnych systemów transportowych (ITS) [4]. W tym sensie sterowniki realizujące sterowanie na kilku skrzyżowaniach mogą być określone jako sterowniki poziomu lokalnego w ITS.

2. Metody sterowania obszarowego

Sterowanie obszarowe, początkowo realizowane w postaci kordynacji liniowej, polegającej na maksymalizacji szerokości wiązek koordynacyjnych wzdłuż ciągu skrzyżowań (tzw. „zielonej fali”), musiało zostać częściowo wyparte bądź przynajmniej uzupełnione kryteriami dotyczącymi lokalnych i globalnych strat czasu, liczby zatrzymań, czy długości kolejek, a niejednokrotnie jeszcze dopełnione koniecznością uprzywilejowania środków transportu zbiorowego.

Nowe potrzeby umożliwiły rozwój kolejnych metod, w zasadzie już systemów sterowania, od których oczekuje się, że funkcja celu łączyć będzie w sobie odpowiednio zdefiniowaną kombinację wielu kryteriów (nie pomijając przy tym faktu umiejscowienia skrzyżowań na przecięciu zazwyczaj co najmniej dwóch ciągów), dając tym samym najefektywniejsze rezultaty.

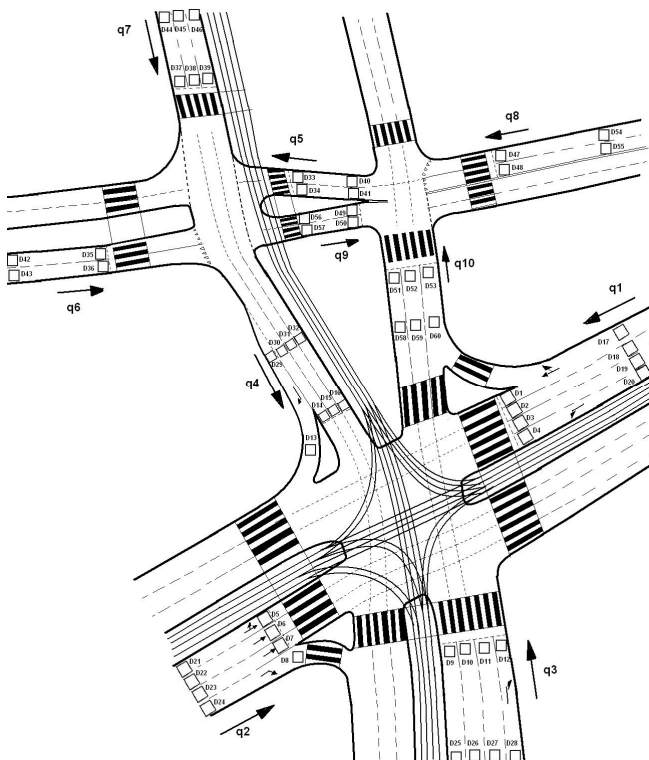
Systemy sterowania obszarowego mogą realizować sterowanie ruchem przez selekcję planów sygnalizacji lub przez ich generowanie na bieżąco, opierając się na historycznych (SIGOP, pierwsze wersje TRANSYT), prognozowanych (zmodyfikowany SIGOP, CYRANO, SCATS) lub całkowicie aktualnych (SCOOT, RHODES, MOTION, UTOPIA – SPOT) danych o ruchu [6].

Wymienione systemy sterowania obszarowego, będące systemami nowej generacji, opierają się na realizacji sterowania poprzez generację planów sygnalizacji i dostosowaniu parametrów algorytmów sterowania do aktualnie występującej sytuacji ruchowej (informacje o ruchu są zbierane na bieżąco przez detektory ruchu i odpowiednio przetwarzane). Nie podlegają one żadnym ograniczeniom czasu uaktualniania parametrów sterowania i charakteryzują się znaczną decentralizacją. Parametrami wyznaczanymi w procesie obliczania mogą być: split, offset, cykl, momenty rozpoczęcia poszczególnych grup bądź faz ruchu na skrzyżowaniach objętych sterowaniem [7].

Algorytmy sterowania dla systemów sterowania obszarowego, stanowią tajemnicę firm oferujących poszczególne systemy i są ściśle chronione. Dlatego też algorytmy sterowania obszarowego zostaną w uproszczony sposób pokazane dla prostego, rzeczywistego obszaru sterowania, obejmującego trzy sąsiednie skrzyżowania, nietworzące struktury liniowej.

3. Algorytm sterowania obszarowego

Dla rzeczywistego obszaru miasta, o założonej organizacji ruchu (rys. 1) i rzeczywistych natężeń ruchu zastosowano, przy tworzeniu algorytmu sterowania, metodę optymalizacji zbliżoną do metody TRANZYT, ze skokową (co 10 minut) procedurą optymalizacji, bazującą na kryterium kombinowanym, którego głównym elementem jest minimalizacja strat czasu, oraz jednokowej dla wszystkich skrzyżowań długości cyklu [7].



Rys. 1. Schemat sieci drogowej obszaru sterowanego
Fig. 1. Diagram of the road network control area

Na podstawie zebranych danych, z umieszczonych na skrzyżowaniach detektorów D, algorytm sterowania obszarowego ustala optymalną długość cyklu i dostosowuje długości faz poszczególnych skrzyżowań, zgodnie z występującymi współczynnikami obciążenia dla grup miarodajnych.

Załączenie sygnalizacji kolorowej, jest możliwe po spełnieniu tzw. warunku natężenia Q. Przyjęto, że warunek ten jest spełniony wówczas, gdy na którymkolwiek wlocie sieci wystąpi natężenie większe niż 200 poj/h.

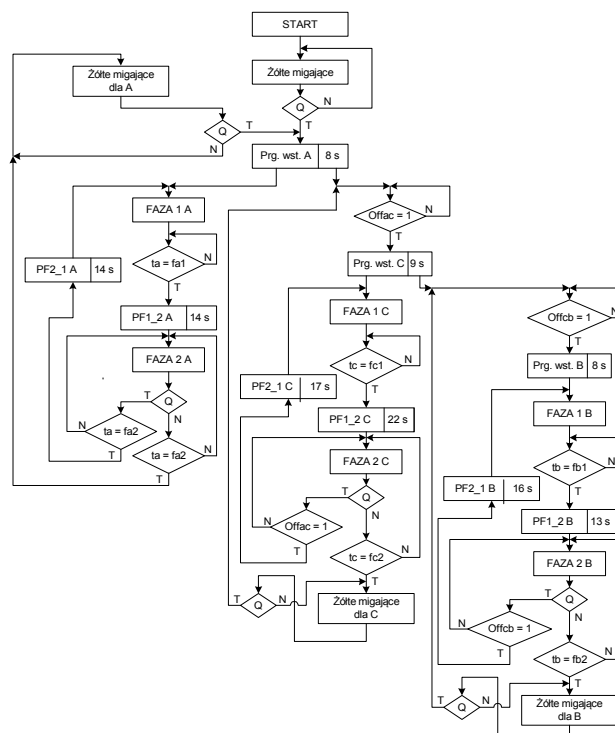
Po spełnieniu warunku Q, na skrzyżowaniach kolejno zaczynają być realizowane tzw. programy wstępne, poprzedzające wejście w fazę pierwszą – koordynowaną. Odstęp czasowy, pomiędzy początkami poszczególnych programów wstępnych jest zależny od przyjętych wartości offsetów.

Przy tych założeniach został opracowany algorytm sterowania obszarowego w postaci graficznego schematu algorytmu (rys. 2), opisującego sterowanie dla poszczególnych skrzyżowań obszaru.

Realizację opracowanego algorytmu sterowania przeprowadzono, przyjmując, że sterownik pracuje w trybie on-line. Sterowanie jest realizowane przez drzewiastą strukturę dwuwarstwową: jedną warstwę tworzą trzy sterowniki A, B, C, obsługujące poszczególne skrzyżowania, drugą zaś stanowi sterownik nadrzędny.

Na pierwszym poziomie zbierane i przetwarzane są informacje o ruchu oraz, realizowane są wszystkie funkcje sterowania bezpośredniego. Natomiast optymalizacja funkcji sterowania przeprowadzana jest w sterowniku nadrzędnym. Oba poziomy sterowania implementowane są w postaci jednego specjalizowanego sterownika.

Opracowana metoda optymalizacji sterowania w obszarze sterowania polega głównie na doborzeniu optymalnej długości cyklu – wspólnej dla wszystkich skrzyżowań, oraz na korygowaniu ustalonej długości faz uwzględniającymi długości kolejek. Można stwierdzić, że w zastosowanej metodzie optymalizacji przyjęte zostało kryterium kombinowane, łączące minimalizację strat czasu i minimalizację długości kolejek.



Rys. 2. Schemat algorytmu sterowania obszarowego
Fig. 2. Algorithm of the control area

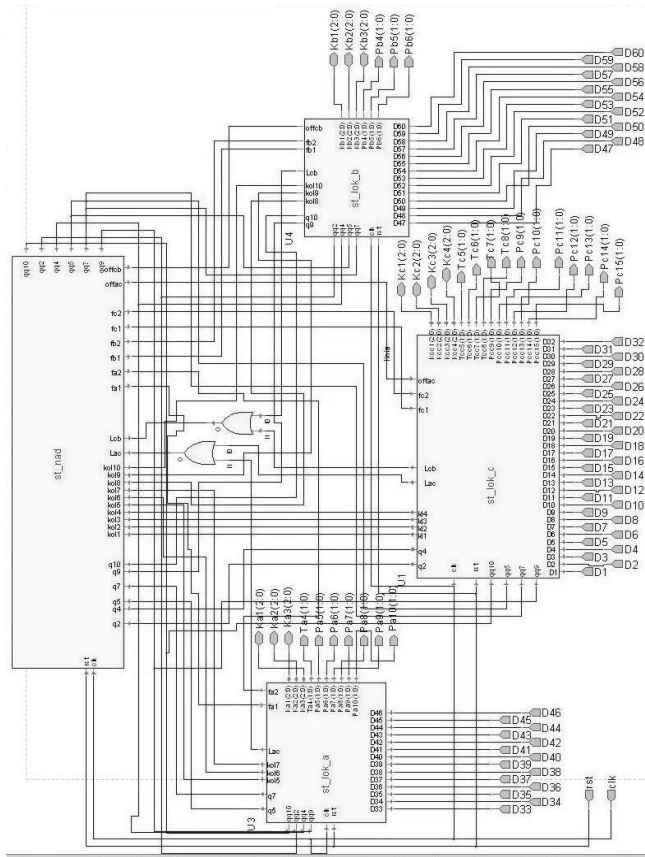
Zastosowana metodyka projektowania pozwala na realizację systemu sterowania obszarowego w postaci specjalizowanego sterownika zaimplementowanego w jednej strukturze FPGA.

4. Budowa sterownika specjalizowanego

Sterownik został wyspecyfikowany w języku VHDL z wykorzystaniem edytorów tekstowych, grafów przejść automatów skończonych, oraz schematów blokowych. Specyfikację sterownika obszarowego przeprowadzono w edytorze BDE, począwszy od jego podziału na bloki sterowników pierwszego poziomu, oraz na blok sterownika nadrzędnego (rys. 3). Każdy z bloków został zdekomponowany na mniejsze bloki funkcjonalne. Takie podejście upraszcza proces specyfikacji, bowiem rozbija złożone układy na prostsze bloki realizujące wyodrębnione funkcje poszczególnych poziomów sterowania.

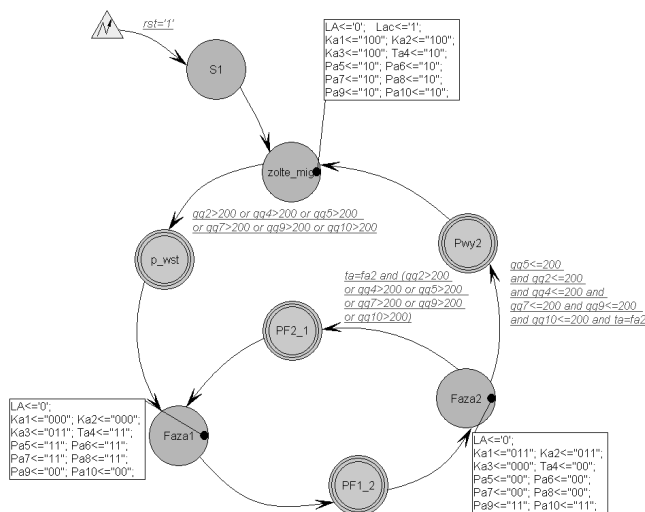
Proces dekompozycji na prostsze bloki prowadzi się do momentu uzyskania modułów, które mogą być specyfikowane w innych edytorach. Jeśli moduł odpowiada układowi kombinacyjnemu, specyfikację prowadzi się bezpośrednio w kodzie języka VHDL.

Moduły odpowiadające synchronicznym układom sekwencyjnym, specyfikowane były w edytorze grafów przejść automatów skończonych, z możliwością zastosowania grafów hierarchicznych. Przykładową specyfikację bloku sterownika A, z wykorzystaniem grafów hierarchicznych, przedstawiono na rys. 4. Pod każdym z wierzchołków hierarchicznych znajduje się podgraf realizujący określony algorytm. Na przykład pod wierzchołkiem hierarchicznym PF1_2 znajduje się podgraf algorytmu przejścia międzyfazowego z fazy 1 do fazy 2 (rys. 5). W analogiczny sposób wyspecyfikowano pozostałe bloki i moduły.



Rys. 3. Schemat blokowy sterownika obszarowego
Fig. 3. Block diagram of the controller area

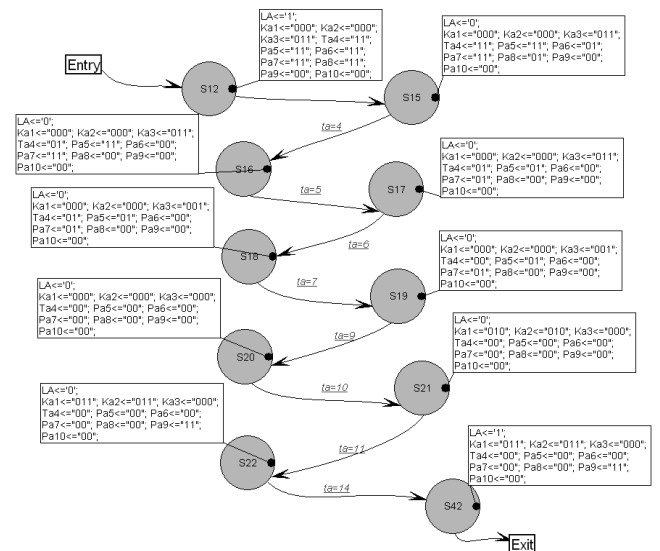
Po skompilowaniu projektu sprawdzono poprawność opracowanych algorytmów oraz poprawność specyfikacji w języku VHDL. Wyniki symulacji można obserwować zarówno w postaci przebiegów poszczególnych sygnałów, jak i bezpośrednio na schemacie blokowym sterownika oraz na grafach sterowania poprzez wyróżnienie stanu, w którym aktualnie znajduje się sterownik obszarowy



Rys. 4. Graf hierarchiczny sterownika A
Fig. 4. Hierarchical graph of the controller A

Po zweryfikowaniu poprawności działania sterownika obszarowego przeprowadzono jego syntezę i implementację w programowalnej strukturze logicznej FPGA Virtex XCV400hg240-4. Raporty z wykorzystania zasobów i parametry czasowe prototypu

sterownika obszarowego wskazują, że zasoby struktury zostały wykorzystane w 75%, natomiast układ może pracować z szybkością przekraczającą 80 MHz.



Rys. 5. Graf przejścia międzyfazowego
Fig. 5. Graf of the interfacial transition

Uzyskane wyniki symulacji, zarówno funkcjonalnej jak i czasowej, potwierdzają zarówno poprawność opracowanego algorytmu, jak i słuszność zastosowanej metody specyfikacji algorytmu sterowania obszarowego.

5. Wnioski

Sterowniki obszarowe ruchu drogowego mogą być realizowane w postaci jednocukłowej w technice SoC, a specjalizowane, rekonfigurowane sterowniki, zaimplementowane w układach FPGA, mogą stanowić poziom lokalny ITS. Taką realizację sterowania obszarowego wyróżnia wysoka szybkość działania oraz dobre parametry niezawodnościowe.

6. Literatura

- [1] Firląg K., Kawalec P.: Realizacja algorytmów sterowania ruchem drogowym w specjalizowanych sterownikach sprzętowych. *Pomiary Automatyka Kontrola*, Wydawnictwo PAK, vol.56, nr 7-2010, Warszawa, 2010, str. 721 – 723.
- [2] Kawalec P., Firląg K.: Reliability analysis of specialized traffic control devices. *Archives of Transport Quarterly*, Polish Academy of Sciences, vol. 19, iss. 1–2, Warsaw, 2007, pp. 75–82.
- [3] Kawalec P., Koliński D.: Synteza i analiza niezawodnościowa urządzeń sterowania ruchem kolejowym w układach FPGA. *Pomiary Automatyka Kontrola*, vol. 53, 7 '2007, Agenda PAK, Warszawa, 2007, s. 57–59.
- [4] Adamski A.: *Inteligentne systemy transportowe*. Wydawnictwa AGH, Kraków, 2003.
- [5] Kawalec P.: *Analiza i synteza specjalizowanych układów modelowania i sterowania ruchem w transporcie*. Oficyna Wydawnicza Politechniki Warszawskiej, Prace Naukowe – Transport, zeszyt 68, Warszawa, 2009.
- [6] Gaca S., Suchorzewski W., Tracz M.: *Inżynieria ruchu drogowego. Teoria i praktyka*. WKŁ, Warszawa, 2009.
- [7] Kawalec P., Sobieszuk-Durka S.: *Wspomagana komputerowo specyfikacja i weryfikacja algorytmów sterowania obszarowego*. *Logistyka* nr 4/2010, Instytut Logistyki i Magazynowania, Poznań, 2010, (wersja elektroniczna).