Piotr Kawalec
Politechnika Warszawska, Wydział Transportu

Sylwia Sobieszuk-Durka
Urząd m. st. Warszawy, Biuro Drogownictwa i Komunikacji

WSPOMAGANA KOMPUTEROWO SPECYFIKACJA I WERYFIKACJA ALGORYTMÓW STEROWNIA OBSZAROWEGO

Streszczenie: W artykule zaprezentowano mechanizm tworzenia algorytmów sterowania ruchem drogowym w wybranym obszarze miasta z wykorzystaniem wspomagania komputerowego. W specyfikacji tych algorytmów wykorzystano grafy przejść automatów skończonych, w tym grafów hierarchicznych. Opracowane algorytmy adaptacyjnego sterowania obszarowego zostały wyspecyfikowane w edytorze FSM, pozwalającym na weryfikację poprawności działania algorytmu w trybie funkcjonalnej symulacji komputerowej. Przytoczono wyniki specyfikacji i weryfikacji opracowanych algorytmów, oraz ich implementacji w programowalnych strukturach logicznych FPGA.
Słowa kluczowe: ruch drogowy, sterowanie obszarowe, algorytmy sterowania adaptacyjnego, grafy hierarchiczne, układy FPGA.

1. WSTEP

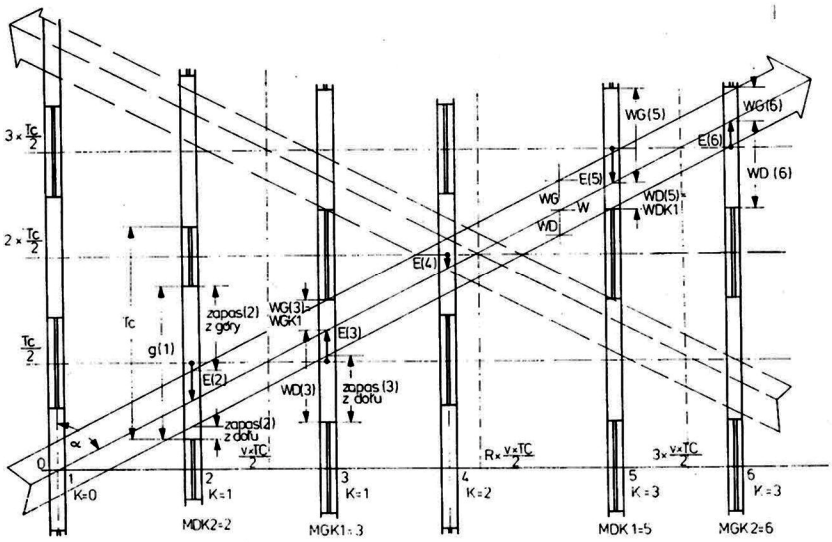
Obserwowany w latach siedemdziesiątych XX wieku rozwój motoryzacji, którego niezwykła i nieprzewidziana intensywność dała się we znaki w latach osiemdziesiątych i dziewięćdziesiątych, oraz jego skutki, zauważalne szczególnie w obszarach miejskich, stały się oczywistym impulsem do poszukiwania nowych, bardziej złożonych metod sterowania ruchem drogowym. Cykliczny program sygnalizacji świetlnej na wybranym skrzyżowaniu, choćby najskrupulatniej przygotowany, w sytuacji sąsiedztwa nierzadko wielu skrzyżowań sterowanych, okazał się rozwiązaniem niewystarczającym. Zwiększająca się liczba skrzyżowań z sygnalizacją świetlną zrodziła potrzebę uzależniania od siebie (koordynowania) sygnalizacji sąsiednich skrzyżowań.

Również w nowych rozwiązaniach sterowników lokalnych i systemów sterowania ruchem drogowym, konieczne stało się uwzględnianie zagadnień sterowania obszarowego. Dotychczasowe wyniki badań poświęconych realizacji sterowników lokalnych w programowalnych strukturach logicznych [1,2] pozwalają stwierdzić, że zasadne jest rozważenie zastosowania takich rozwiązań na poziomie sterowania obszarowego.

Niewielkie wykorzystanie zasobów struktur FPGA przez sterowniki lokalne na pojedynczym skrzyżowaniu, stwarza szansę wykorzystania specjalizowanych sterowników zrealizowanych w FPGA, do sterowania w obszarze ulic. Należy oczekiwać, że odpowiednie algorytmy sterowania obszarowego 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.

2. METODY STEROWANIA OBSZAROWEGO

Pierwsze z nowych metod, polegające na liniowej koordynacji sygnalizacji świetlnej (np. MAXBAND, SIGPROG, czy polskie: KORD, KORS), służące do zwiększania efektywności cyklicznego sterowania sygnalizacją na ciągach komunikacyjnych charakteryzowała wspólna cecha: jednakowy główny cel optymalizacji, tj. maksymalizacja sumy szerokości wiązek koordynacyjnych. Sposób optymalizacji, właściwy dla każdej z metod, sprowadza się tu zwykle do doboru (w pewnym narzuconym zakresie swobody) parametrów sterowania, takich, jak: szerokości wiązek koordynacyjnych, wartości offsetów, długość cyklu, splity lub długości sygnałów zielonych, przy wcześniej zdefiniowanych wielkościach charakteryzujących rozpatrywany ciąg (np. liczba skrzyżowań, odległości między nimi, proponowane czasy otwarcia dla kierunków koordynowanych, prędkość – średnia lub jej zakres od minimum do maksimum, proponowana długość cyklu lub jego zakres, czas przejazdu, itp.). Wynikiem prowadzonych wspomnianymi metodami obliczeń, oprócz konkretnych wartości liczbowych, jest charakterystyczny wykres koordynacji, ilustrujący podstawowe jej parametry (rys. 1).



Rysunek 1. Wykres liniowej koordynacji sygnalizacji wraz z oznaczeniami podstawowych parametrów

Przedstawiony na rys. 1 wykres jest wynikiem prowadzonych jeszcze w latach 70. przez biuro BPBK „Stolica" prac nad algorytmem liczenia koordynacji ulicznej sygnalizacji świetlnej (pisanego wówczas w języku FORTRAN 1900). Jest to ilustracja problemu koordynacji w prostokątnym układzie współrzędnych, gdzie na osi czasu (t) są odkładane odcinki odpowiadające założonej długości cyklu sygnalizacyjnego c, jednakowego dla wszystkich skrzyżowań rozpatrywanego ciągu. Dla każdego ze skrzyżowań wydzielono tzw. *czas otwarcia wlotu* (tu interpretowany jako sygnał zielony wraz z czasem międzyzielonym) g_i oraz *czas zamknięcia wlotu*. Na osi drogi (l) są umieszczone punkty odpowiadające kolejnym, ponumerowanym skrzyżowaniom. Pierwsze skrzyżowanie jest umieszczone w początku układu współrzędnych. Odległość skrzyżowania i-tego od skrzyżowania pierwszego jest oznaczona l_i. l_1 jest więc równa 0. W powyższy układ prostokątny wrysowano prostą ruchu w kierunku A o równaniu (1).

$$t = 1/v^A * l \tag{1}$$

gdzie:

v^A – jest prędkością średnią pojazdu poruszającego się w kierunku A,

$1/v^A$ – odpowiada współczynnikowi kierunkowemu tej prostej.

Fazy zielone na każdym skrzyżowaniu usytuowano tak, by przechodziła przez nie prosta ruchu, pamiętając o tym, by warunek ten zachodził dla obu kierunków ruchu. Konfiguracja faz zielonych cyklu na skrzyżowaniach powinna być taka, by jak najwięcej pojazdów mogło przejechać przez cały ciąg bez zatrzymania. Miarą tych pojazdów może być czas upływający pomiędzy momentem przejazdu przez dowolny punkt ciągu pierwszego i ostatniego pojazdu z grupy pojazdów jadących bez zatrzymania w poszczególnych kierunkach. Jest to nic innego jak wspominana wcześniej szerokość wiązek koordynacyjnych [3].

Metody koordynacji sygnalizacji świetlnej ulegały nieustannym modyfikacjom, za równo ze względu na zwiększające się możliwości obliczeniowe, sprzętowe, jak i samo

zmieniające się podejście do zagadnień sterowania ruchem. Niewątpliwie wiele z nowych rozwiązań powstało za sprawą rozwijających się równolegle: z jednej strony – metod sterowania adaptacyjnego, dających nowe możliwości za równo na lokalnym, jak i wyższym poziomie sterowania, z drugiej zaś - zrozumiałych tendencji sieciowego traktowania problemu synchronizacji sygnałów na sterowanych skrzyżowaniach.

Nasycenie układów komunikacyjnych i zauważalne problemy z przepustowością wlotów skrzyżowań wpłynęły znacząco na poszerzenie zbioru kryteriów optymalizacji. Popularne dotąd pojęcie „zielonej fali" związane z maksymalizacją szerokości wiązek koordynacyjnych wzdłuż ciągu skrzyżowań 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 optymalnie najefektywniejsze rezultaty.

Do bardziej znanych metod sterowania obszarowego zaliczyć można m.in.: PASSER (Progression Analysis and Signal System Evaluation Routine), SIGOP (SIGnal OPtimization, CYRANO (CYcle – free Responsive Algorithm for Network Optimization), TRANSYT (TRAffic Network StudY Tool), SCOOT (Split, Cycle and Offset Optimizer Technique), SIDRA (Signalized and Unsignalized Intersection Design and Research Aid), SYNCHRO, SCATS (Sydney Coordinated Adaptive Traffic System), RHODES (Real-time Hierarchical Optimized Distributed and Effective System), MOTION (Method for Optimization of Traffic signals in On-line controlled Network), UTOPIA (Urban Traffic Optimisation by Integrated Automation) [4,5,6].

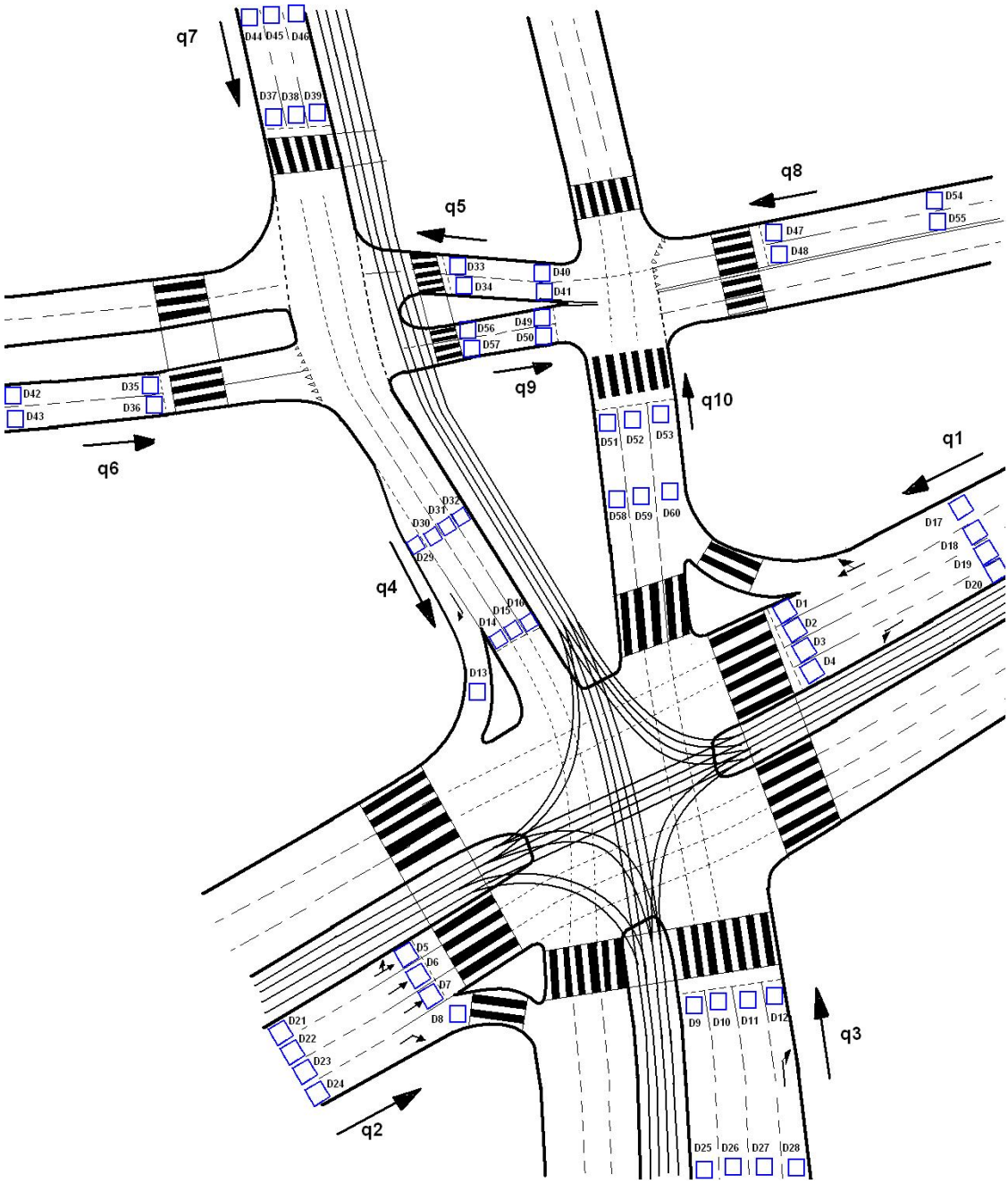
Systemy sterowania mogą realizować sterowanie ruchem przez selekcję planów sygnalizacji lub przez ich generowanie na bieżąco, opierając się na historycznych (SIGOP, pierwsze wersje TRANSYT), prognozowanych (zmodyfikowany SIGOP, CYRANO, SCATS) lub całkowicie aktualnych (SCOOT, RHODES, MOTION, UTOPIA –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

Ostatnie z wymienionych, jako systemy nowej generacji, opierają się na realizacji sterowania poprzez generację planów sygnalizacji i dostosowaniu parametrów algorytmów sterowania do aktualnie występującej sytuacji ruchowej (informacje o ruchu zbierane na bieżąco przez detektory ruchu, 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.

3. BUDOWA ALGORYTMU STEROWANIA OBSZAROWEGO

Dla rzeczywistego obszaru miasta, o założonej organizacji ruchu (rys. 2) i rzeczywistych natężeń ruchu zastosowano, przy tworzeniu algorytmu sterowania, metodę

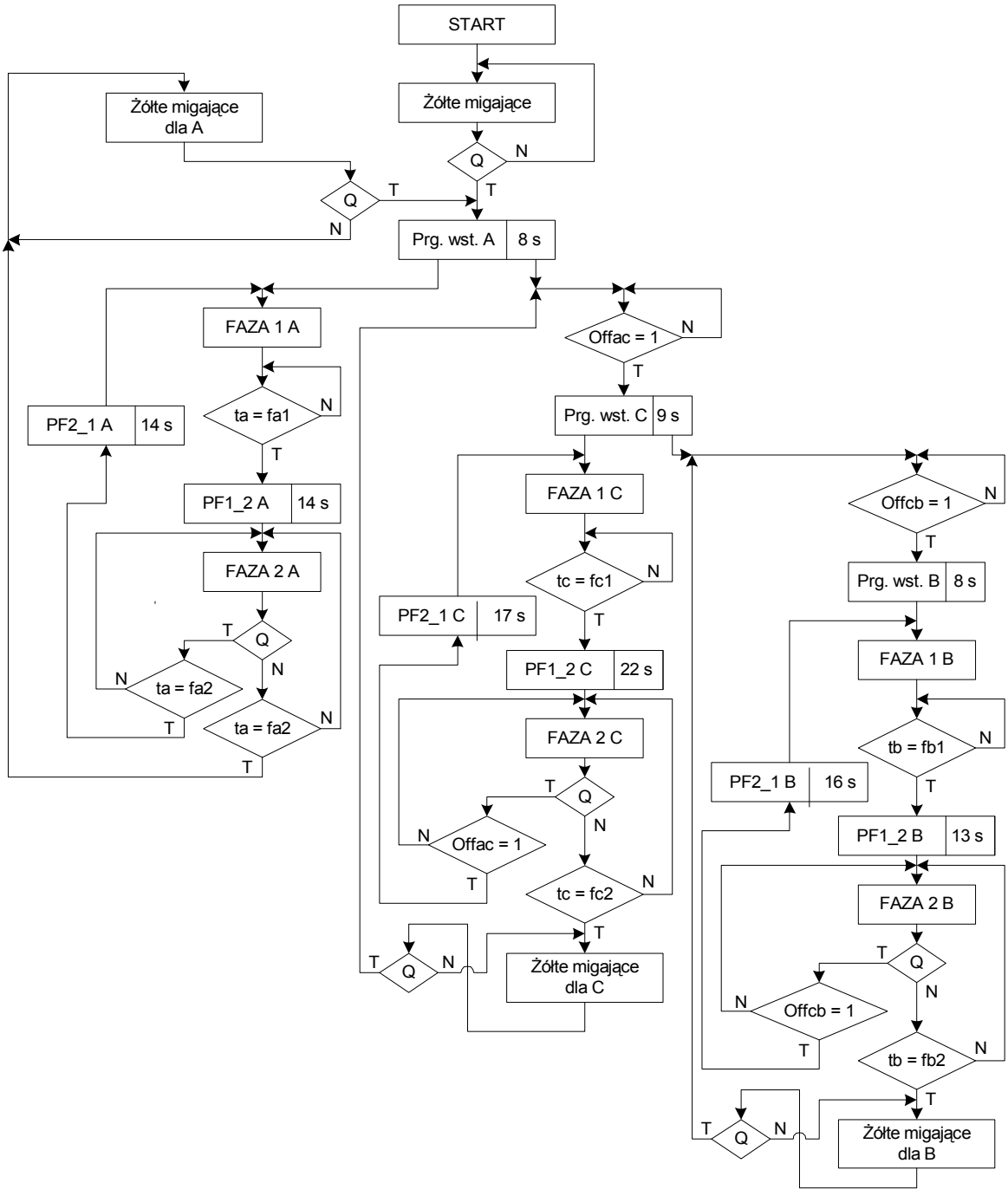
optymalizacji zbliżoną do metody TRANZYT, ze skokową (co 10 minut) procedurą optymalizacji, bazującą na kryterium kombinowanym, którego głównym elementem jest minimalizacja strat czasu, oraz jednakowej dla wszystkich skrzyżowań długości cyklu [7].



Rys. 2. Schemat sieci drogowej z naniesionymi detektorami i natężeniami na wlotach

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.



Rys. 3. Sieć działań sterowania obszarowego

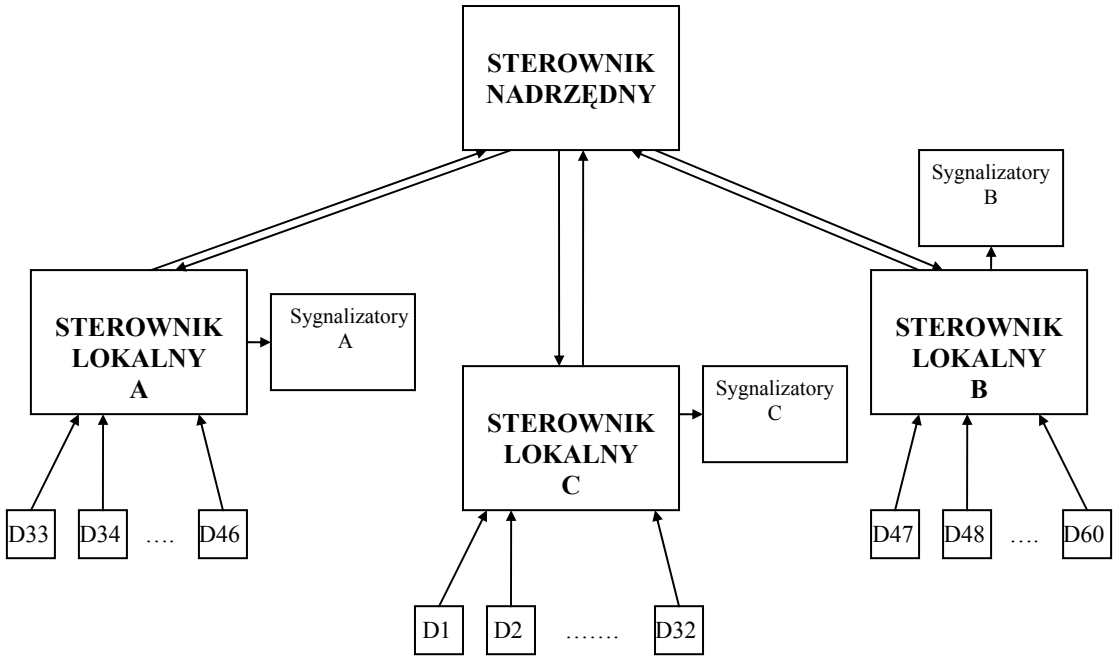
Załączenie sygnalizacji kolorowej, jest możliwe po spełnieniu tzw. warunku natężenia Q. Przyjęto, że warunek ten jest spełniony wówczas, gdy na którymkolwiek wlocie sieci wystąpi natężenie większe niż 200 poj/h.

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.

Dla tych założeń został opracowany algorytm sterowania obszarowego w postaci GSA opisującego sterowanie dla poszczególnych skrzyżowań obszaru (rys.3).

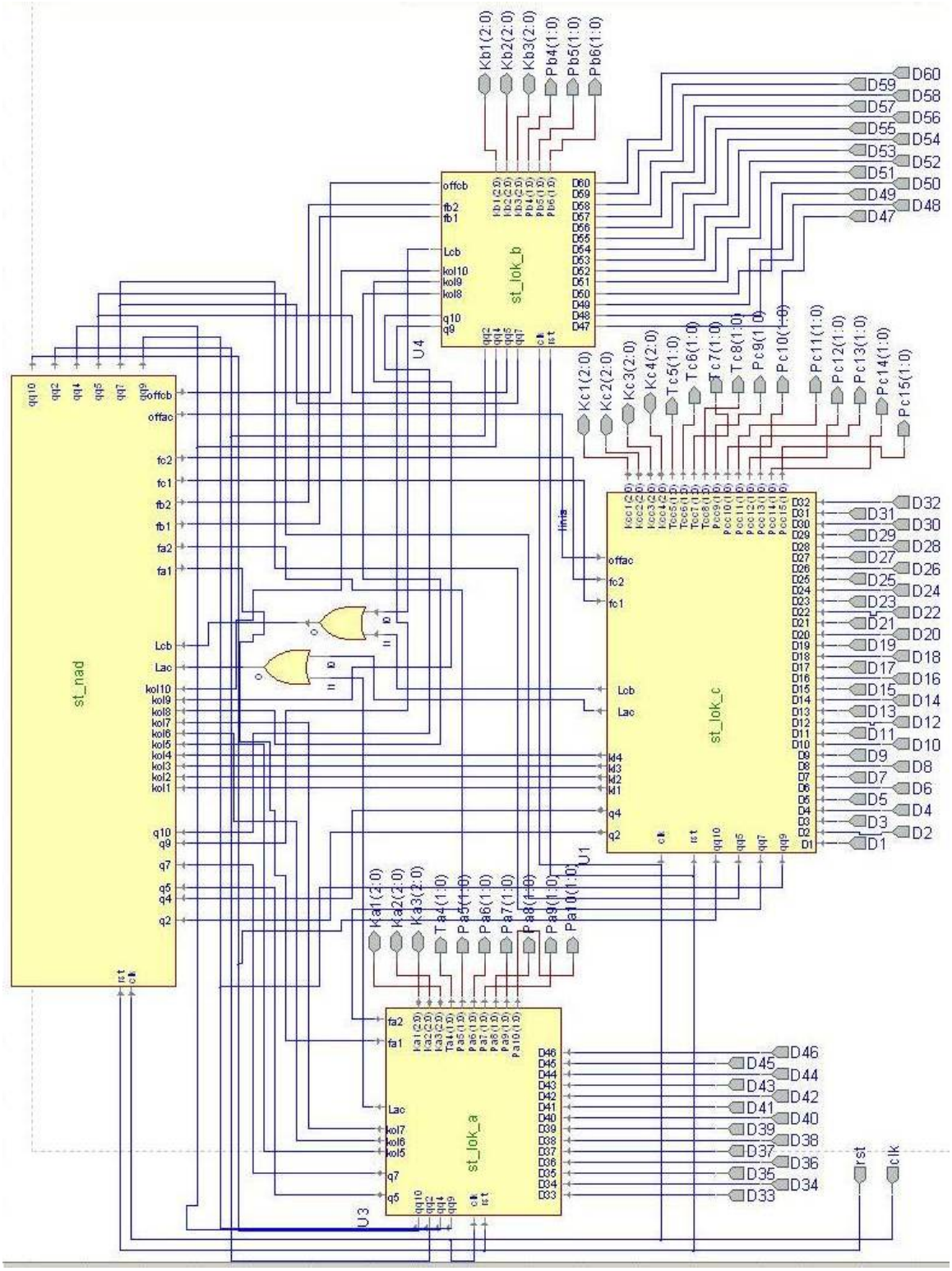
4. SPECYFIKACJA ALGORYTMU STEROWANIA OBSZAROWEGO

Realizacja opracowanego algorytmu sterowania jest możliwa dzięki stworzeniu systemu sterowania obszarowego, obejmującego trzy sterowniki lokalne A, B, C, oraz sterownik nadrzędny (rys. 4). Jest to system pracujący w trybie on-line. Sterowanie jest tu realizowane przy współdziałaniu czterech podstawowych bloków tworzących strukturę drzewiastą. Struktura ta ma charakter dwuwarstwowy: jedną warstwę (lokalną), tworzą trzy sterowniki lokalne przynależne do każdego ze skrzyżowań, drugą zaś stanowi sterownik nadrzędny.



Rys. 4. Struktura systemu sterowania obszarowego

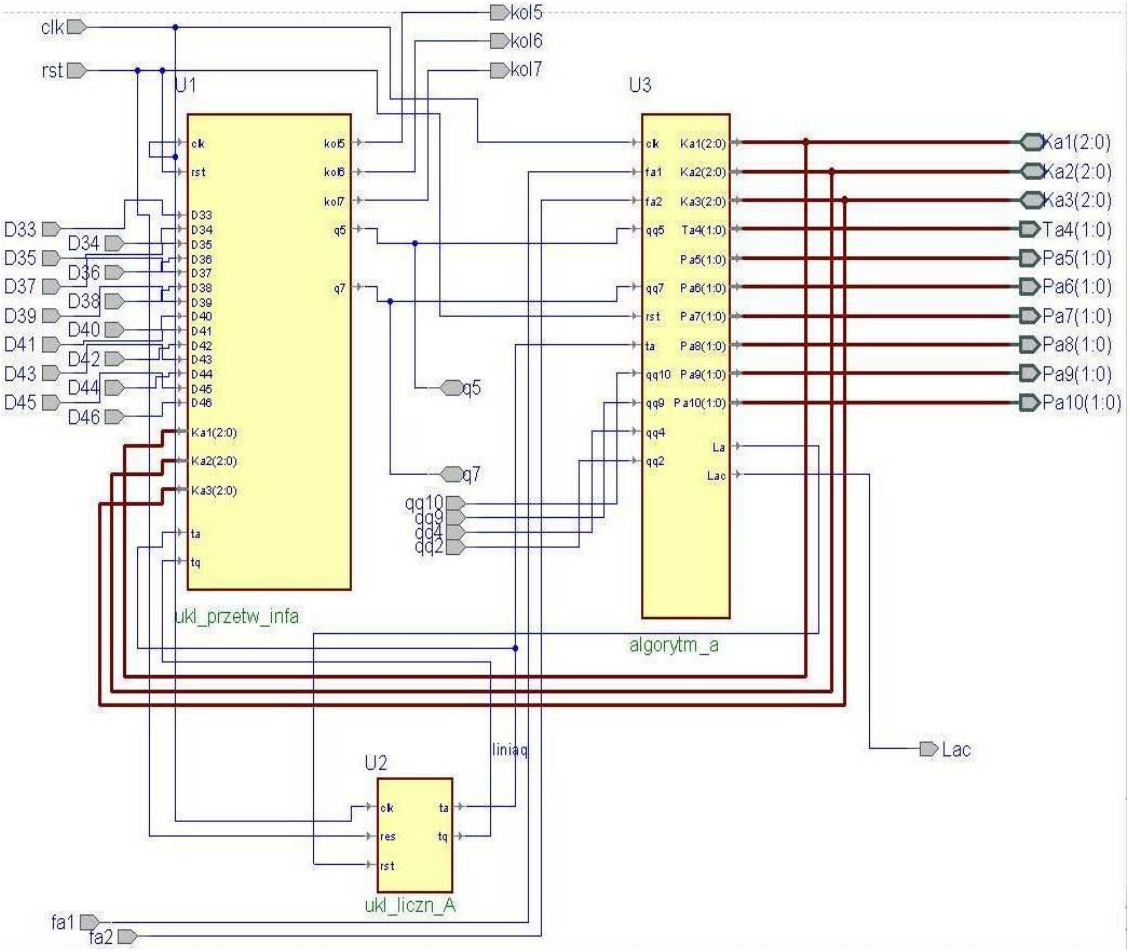
Na poziomie lokalnym zbierane są informacje o ruchu oraz realizowane są funkcje zbierania informacji o ruchu, oraz wszystkie funkcje realizowane przez sterownik lokalny. Natomiast optymalizacja funkcji sterowania realizowana jest w sterowniku nadrzędnym.



Rys. 5. Schemat blokowy sterownika obszarowego

Specyfikację sterownika obszarowego przeprowadzono w edytorze BDE, poczynając od jego podziału na bloki sterowników lokalnych, oraz na blok sterownika nadrzędnego (rys. 5).

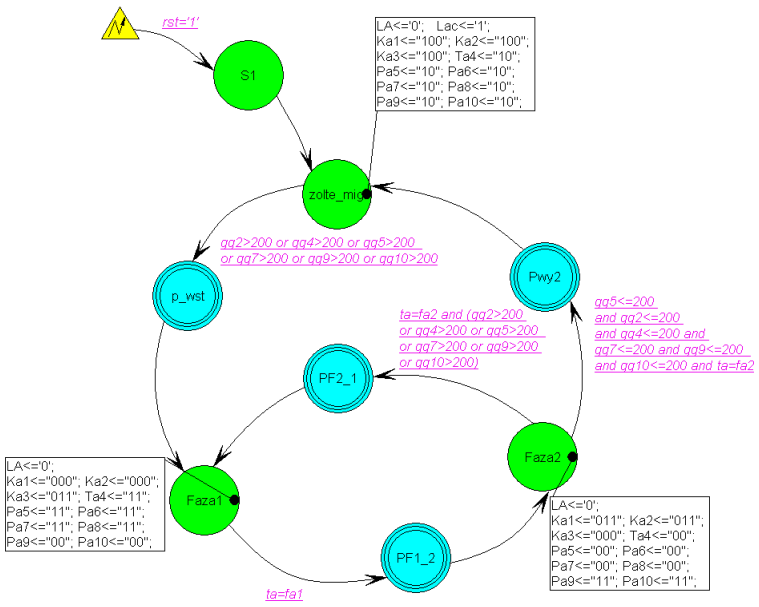
Każdy z bloków został zdekomponowany na mniejsze bloki funkcjonalne. Takie podejście upraszcza proces specyfikacji, bowiem rozbija złożone układy na prostsze bloki realizujące wyodrębnione funkcja poszczególnych poziomów sterowania (rys. 6).



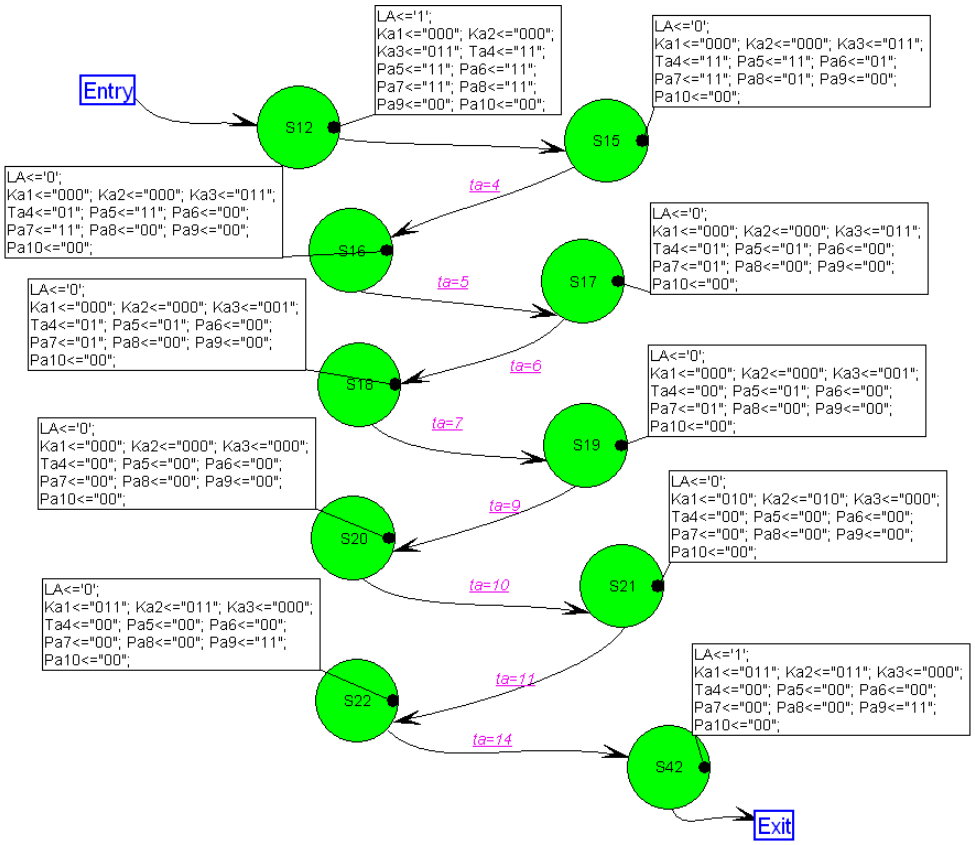
Rys. 6. Schemat blokowy sterownika lokalnego A

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, wykorzystując edytor tekstowy HDE.

Moduły odpowiadające synchronicznym układom sekwencyjnym, specyfikuje się, zgodnie z podana metodyką, w edytorze grafów przejść automatów skończonych FSM, z możliwością zastosowania grafów hierarchicznych. Przykładową specyfikację bloku algorytm_a sterownika lokalnego A, w edytorze FSM, z wykorzystaniem grafów hierarchicznych, przedstawiono na rys. 7.



Rys. 7. Specyfikacja w FSM bloku algorytm_a



Rys. 8. Podgraf przejścia międzyfazowego PF1_2

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. 8). W analogiczny sposób wyspecyfikowano pozostałe bloki i moduły.

5. WNIOSKI

Zastosowanie wspomagania komputerowego pozwala zarówno na graficzną specyfikację algorytmów sterowania obszarowego, jak i na ich weryfikację w postaci symulacji komputerowej. Możliwość dekompozycji złożonych algorytmów sterowania na prostsze bloki, a następnie na graficzne łączenie tych bloków. Ponieważ wspólną platformę tych form opisu algorytmów stanowi język opisu sprzętuVHDL, to poprawnie wyspecyfikowany algorytm sterowania może być poddany automatycznej syntezie, a następnie implementacji i prototypowaniu w programowalnych strukturach logicznych.

Bibliografia

1. Kawalec P.: Analiza i synteza specjalizowanych układów modelowania i sterowania ruchem w transporcie. Oficyna Wydawnicza Politechniki Warszawskiej, Prace Naukowe-Transport , z. 68, Warszawa, 2009.
2. Kawalec P., Firląg K.: Synteza specjalizowanych układów sterowania ruchem drogowym w strukturach FPGA. Pomiary Automatyka Kontrola, 7bis 2006, Agenda Wydawnicza Stowarzyszenia SIMP, Warszawa, s. 8–10.
3. Wrześniowski Z.: Koordynacja sygnalizacji świetlnej. WKŁ, Warszawa 1977.
4. Leśko M., Guzik J.: Sterowanie ruchem drogowym, sterowniki i systemy sterowania i nadzoru ruchu. Wydawnictwo Politechniki Śląskiej, Gliwice 2000.
5. Szatkowski M.: materiały do wykładów z przedmiotu Sterowanie ruchem drogowym, Warszawa, 2005.
6. http://faculty.kfupm.edu.sa
7. Sobieszuk-Durka S: Zastosowanie języków opisu sprzętu do modelowania wybranych algorytmów sterowania ruchem drogowym. Praca magisterska, Wydział Transportu PW, Warszawa, 2006.

COMPUTER SUPPORTED SPECIFICATION AND VERIFICATION OF AREA CONTROL ALGORITHMS

Abstract: The article presents a mechanism of creating algorithms of traffic control in a selected area of the city with the use of computer support. In the specification of these algorithms transition graphs of finite state machines have been used, including hierarchical graphs. Designed algorithms of adaptive area control have been specified in FSM editor, which allows verification of correctness of algorithm operation in the mode of functional computer simulation. Results of specification and verification of the designed algorithms have been quoted, as well as their implementation in programmable logic FPGA structures.

Keywords: traffic control, algorithms of adaptive area control, hierarchical graphs, FPGA devices