



Piotr KAWALEC, Marcin RŻYSKO

KOMPUTEROWO WSPOMAGANA SPECYFIKACJA FUNKCJI ZALEŻNOŚCIOWYCH URZĄDZEŃ SRK W JĘZYKACH OPISU SPRZĘTU

Streszczenie

W artykule omówiona została koncepcja wykorzystania języków opisu sprzętu do specyfikacji funkcji zależnościowych urządzeń sterowania ruchem kolejowym. Przegląd stosowanych rozwiązań pokazuje, że nie istnieje metoda opisu zapewniająca jednolitą platformę specyfikacji dyskretnych układów sterowania w kolejnictwie. Obecnie wykorzystywane metody nie oferują również możliwości statycznej i dynamicznej weryfikacji wykonanego układu. Proces projektowania przy użyciu proponowanej metody zaprezentowany został na przykładzie specyfikacji algorytmu działania pojedynczego modułu dla geograficznego systemu zależnościowego, tworzonego w języku VHDL. Zastosowanie symulatorów logicznych w procesie weryfikacji projektu pozwoliło na wykonanie symulacji potwierdzających poprawność działania modelu. Na podstawie tak wykonanej specyfikacji pokazano zalety wykorzystania wspomaganie komputerowego w postaci wszechstronnego pakietu przeznaczonych do tworzenia w językach HDL.

WSTĘP

Urządzenia sterowania ruchem kolejowym, a w szczególności systemy zależnościowe, spełniać muszą rygorystyczne warunki w zakresie bezpieczeństwa i niezawodności. Każdy błąd w ich funkcjonowaniu może potencjalnie prowadzić do zagrożenia życia ludzkiego i wysokich strat materialnych. Źródłem błędów, oprócz uszkodzeń sprzętowych, może być niewłaściwie wykonana specyfikacja. Dlatego wraz z rosnącą niezawodnością elementów, z których budowane są urządzenia, zwiększać musi się także pewność metod projektowania.

W przypadku urządzeń zabezpieczenia ruchu pociągów stosowanych jest wiele metod opisu, zależnie od wybranej techniki realizacji. W ostatnich latach najczęściej zabudowywane są urządzenia komputerowe. Obecna tendencja do obejmowania coraz większych obszarów sterowaniem z jednej nastawni (tzw. Lokalne Centra Sterowania) powoduje, iż znacząco wzrasta ilość funkcji realizowanych przez poszczególne systemy srk. Konieczność współpracy wielu typów urządzeń (systemy zależnościowe, blokady liniowe, systemy przekazywania informacji o pociągu, układy diagnostyki taboru) w obrębie jednego obszaru sterowania powoduje, że niezbędne staje się opracowanie jednolitej i spójnej platformy projektowej i opisowej dla różnych rodzajów urządzeń i systemów sterowania ruchem kolejowym.

Artykuł przedstawia koncepcję wykorzystania do stworzenia takiej platformy języków opisu sprzętu wraz z dedykowanymi narzędziami wspomaganie komputerowego.

Udowodniona została przydatność wykorzystania wspomaganie komputerowego do szczegółowej analizy działania urządzeń przekaźnikowych [8]. Poruszany jest również w pracach naukowych temat zastosowania w technice srk układów programowalnych, tworzonych przy użyciu języków opisu sprzętu [3]. W niniejszym artykule przedstawiony zostanie wspomagany komputerowo proces specyfikowania i weryfikowania modelu urządzeń zależnościowych.

1. METODY OPISU URZĄDZEŃ SRK

W przypadku pierwszych urządzeń zabezpieczenia ruchu kolejowego zasadniczymi metodami zapisu projektu były opisy słowne i rysunki konstrukcyjne. Dla urządzeń przekaźnikowych powszechną formą opisu jest schemat elektryczny. Taki schemat odzwierciedla fizyczne połączenia poszczególnych elementów układu, jest więc dogodną formą do montażu. Jako metoda projektowania, opiera się wyłącznie na doświadczeniu projektanta. Możliwa jest tylko statyczna analiza układu, która jednak w przypadku złożonych układów staje się trudna i długotrwała, nie dając wciąż gwarancji, że urządzenie w ogóle uda się uruchomić.

Graficzne metody opisu, takie jak schematy blokowe algorytmu lub grafy przejść (maszyny stanów) stosowane są często przy projektowaniu urządzeń komputerowych, stanowią jednak tylko pośredni etap syntezy. Bez narzędzi wspomaganie komputerowego nie jest bowiem możliwe automatyczne przekształcenie tak stworzonego modelu w program gotowy do zaimplementowania w urządzeniu. Takie narzędzia są już co prawda wykorzystywane do pojedynczych rozwiązań, jednak ich zastosowanie ograniczone jest do konkretnego urządzenia, nie mogą więc stanowić uniwersalnej platformy projektowej.

Prowadzone są prace badawcze ukierunkowane na formalne metody specyfikacji urządzeń sterowania ruchem kolejowym [1, 5, 6, 7, 9]. Proponowane były specjalne języki programowania dedykowane wyłącznie projektowaniu takich układów [4]. Podejście formalne w istocie minimalizuje ryzyko przeoczenia błędu w procesie tworzenia specyfikacji. Należy jednak zwrócić uwagę na to, aby wykorzystywana metoda była nieustannie wykorzystywana i rozwijana przez wiele branż automatyki, co zapewnia stały dostęp do najnowszych zdobyczy techniki w zakresie sprzętowym. Dlatego do stworzenia uniwersalnej platformy formalnego opisu urządzeń srk zaproponowane zostały języki opisu sprzętu.

2. JĘZYKI OPISU SPRZĘTU

Wykorzystanie języków opisu sprzętu (HDL – *Hardware Description Languages*) stanowi dziś światowy standard w projektowaniu układów elektronicznych, w szczególności w branżach wymagających od urządzeń wysokiej niezawodności działania takich jak lotnictwo czy wojsko. Języki HDL pozwalają na tworzenie formalnego opisu działania układu, będącego zarówno dokumentacją projektu, jak i podstawą do implementacji oraz prototypowania. Popularność języków opisu sprzętu sprawia, że posiadają one wiele dedykowanych narzędzi CAD (komputerowego wspomaganie prac projektowych). Pakiety oprogramowania projektowego upraszczają proces tworzenia kodu, a także umożliwiają automatyczną weryfikację oraz symulację działania opracowanego modelu, co stanowi podstawowy argument przemawiający za ich wykorzystaniem.

Dokonując wyboru języka należy kierować się dostępnością wszechstronnych i kompletnych narzędzi CAD, które umożliwiają wykonanie każdego etapu projektu (specyfikacja, weryfikacja, synteza, implementacja). Takie warunki mogą zapewnić tylko języki mające status światowego standardu, takie jak język *VHDL*, który znalazł zastosowanie w modelowaniu złożonych układów cyfrowych w wielu branżach automatyki [12]. Proces komputerowo wspomaganie specyfikacji funkcji urządzeń srk zaprezentowany zostanie na

przykładzie specyfikacji i weryfikacji przykładowego algorytmu działania stacyjnych urządzeń sterowania ruchem kolejowym w tym języku.

3. KONCEPCJA MODUŁOWEGO SYSTEMU STACYJNYCH URZĄDZEŃ SRK

W przypadku pierwszych scentralizowanych urządzeń sterowania ruchem kolejowym stosowany był podział funkcjonalny. To znaczy, że dany element systemu odpowiadał za właściwą sobie funkcję w całym okręgu nastawczym. Skrzynia zależności odpowiadała za wykluczenie sprzecznych przebiegów i kontrolę położenia zwrotnic, aparat blokowy za utwierdzenie przebiegu i obsługę blokad. Podobnie we wczesnych urządzeniach przekaźnikowych praca opierała się na dedykowanych obwodach utwierdzenia, zwolnienia czy sygnałowych. Trudności związane z wdrażaniem takich urządzeń wynikały z konieczności indywidualnego projektowania elementów i obwodów dla danej stacji. Na podstawie albumów schematów konkretnego systemu (przedstawiającego jedynie wybrane, charakterystyczne przypadki zastosowania) projektant opracowywał schematy sieci przekaźnikowych dostosowane do zadanego układu torowego. Taka metoda podziału wydłużała proces projektowania, jak również wprowadzała zbyt dużą dowolność rozwiązań.

Trudności te wymusiły stworzenie nowej metodyki, w której zrezygnowano z funkcjonalnego podziału urządzeń stacyjnych, a skupiono się na podziale geograficznym, tworząc bloki logiczne odpowiadające urządzeniom w terenie. Już w systemie *PB* (lata 60-te) zastosowano metodę projektowania charakterystyczną dla urządzeń zblokowanych (każdemu elementowi w terenie odpowiada schemat blankietowy), ograniczała się ona jednak jedynie do etapu projektu, w warstwie wykonawczej były to urządzenia o strukturze właściwie niezblokowanej [10]. Dopiero urządzenia takie jak *IZH-111* wykonane były jako zblokowane zarówno w części projektowej jak i wykonawczej. System ten składał się z uniwersalnych modułów semaforów, zwrotnic, tarcz manewrowych i innych typowych elementów. Wystarczyło połączyć je kablami geograficznymi zgodnie z układem torowym stacji oraz zaprogramować wtykami programowymi, aby uzyskać kompletny projekt [11]. Wadą tego systemu była jednak realizacja w technice przekaźnikowej, powodująca duże koszty wdrożenia i utrzymania, oraz wysokie zapotrzebowanie na powierzchnię pomieszczeń w nastawni.

Filozofia tworzenia urządzeń stacyjnych z dekompozycją systemu na pojedyncze moduły w układzie geograficznym pozostaje wiodącym trendem również w obecnie produkowanych systemach komputerowych [2]. Pozwala ona uprościć zarówno proces projektowania urządzeń dla konkretnego układu torowego, jak i przebudowy urządzeń oraz ich walidacji [5]. Dlatego też przedstawiony dalej projekt systemu opiera się właśnie na takim podejściu.

4. PROCES PROJEKTOWANIA W ŚRODOWISKU JĘZYKA OPISU SPRZĘTU

Pierwszym krokiem przy projektowaniu układu sterowania jest określenie funkcji, jakie musi pełnić oraz sposobu, w jaki funkcje te mają być realizowane.

Przygotowywany model stacyjnych urządzeń sterowania ruchem kolejowym posiada zblokowaną strukturę, w której każdemu z urządzeń zewnętrznych i każdemu elementowi układu torowego odpowiada dedykowany moduł realizujący (we współpracy z sąsiednimi modułami) wszystkie funkcje systemu zależnościowego.

Do takich funkcji należą:

- wybieranie drogi przebiegu,
- nastawianie zwrotnic w drodze przebiegu,
- kontrola poprawności ułożenia drogi i niezajętości odcinków w tej drodze,

- utwierdzenie przebiegu,
- przesyłanie informacji do urządzeń przytorowych (np. semaforów),
- samoczynne zwalnianie przebiegu.

Wymiana informacji pomiędzy modułami odbywa się przy pomocy 16-bitowych sygnałów zgodnych ze standardem IEEE 1164 (*Standard Logic*), przesyłanych równolegle. Ponadto moduły posiadają wejścia i wyjścia przeznaczone do komunikacji z układami sterującymi, wykonawczymi lub urządzeniami kontrolnymi w terenie.

Obsługa systemu opiera się na wejściach adresujących i wejściach poleceń ogólnych. Polecenia ogólne, takie jak wybór rodzaju przebiegu czy anulowanie wybierania, przesyłane są do wszystkich modułów jednocześnie. Aby wybrać przebieg, należy wysłać sygnał na wejście adresujące modułu pierwszego semafora, następnie drugiego, po czym wybrany może być rodzaj przebiegu. Nastawienie drogi przebiegu, kontrola oraz utwierdzenie następują automatycznie, po czym wysyłany jest odpowiedni sygnał do modułu wykonawczego semafora. Informacja o dozwolonej prędkości jazdy kodowana jest w postaci 6-bitowej (po 3 bity dla pierwszej i drugiej drogi jazdy za semaforem), co umożliwi zastosowanie 8 stopni prędkości (0, 40, 50, 60, 80, 100, 130 km/h oraz prędkość maksymalna). Pozwala to na pracę z nowo wdrażanymi układami geometrycznymi głowic stacyjnych, w szczególności zakładając wprowadzenie w przyszłości na sieci PKP PLK wskaźników diodowych wyświetlających dozwoloną prędkość na semaforze.

Proces projektowania przy użyciu proponowanej techniki zaprezentowany zostanie na przykładzie modelu jednego z elementów projektowanego systemu – modułu *SEM* (semafora drogowaskazowego lub wyjazdowego).

4.1. Budowa algorytmu i jego zapis formalny

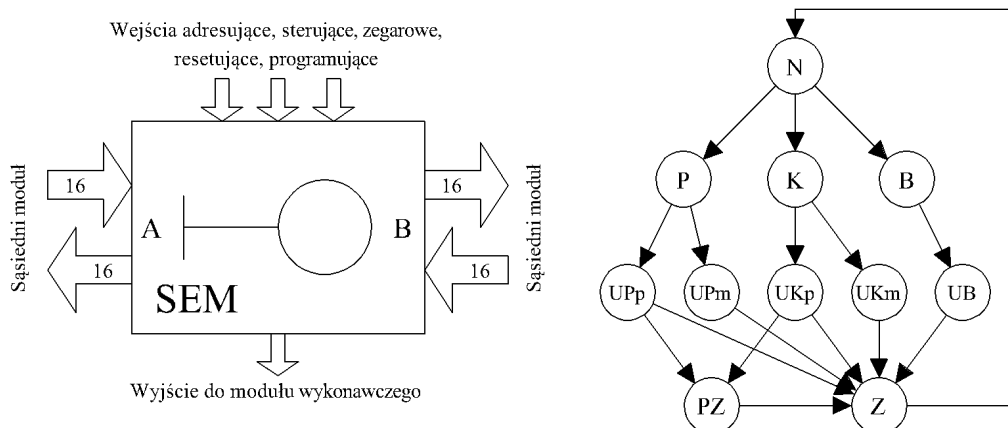
Przed przystąpieniem do konstruowania algorytmu działania modułu określone zostały wymagane dane, które powinien odbierać oraz generować moduł. Struktura wejść i wyjść zilustrowana została na rys. 1. Oprócz portów służących do komunikacji z sąsiednimi modułami zaprojektowano również wejścia dla sygnałów pochodzących z pulpitu nastawczego, wejście programujące (dostosowujące zachowanie modułu do specyficznych rozwiązań w układzie stacji) oraz wyjście do układu wykonawczego semafora.

Na rys. 1 przedstawiono również podstawowe stany, w jakich może znajdować się moduł semafora. Zasadniczy stan, oznaczony symbolem N , to stan neutralny, w którym moduł jest gotowy do pracy i oczekuje na polecenia.

Przewidziano następujące zdarzenia inicjujące:

- a) sygnał z wejścia adresującego – moduł zostaje wybrany jako początkowy (stan P),
- b) sygnał z modułu sąsiedniego od strony wejścia A – moduł może zostać wybrany jako końcowy (stan K),
- c) sygnał z modułu sąsiedniego od strony wejścia B – moduł może pośredniczyć w nastawianiu przebiegu w kierunku przeciwnym do ustawienia semafora (stan B).

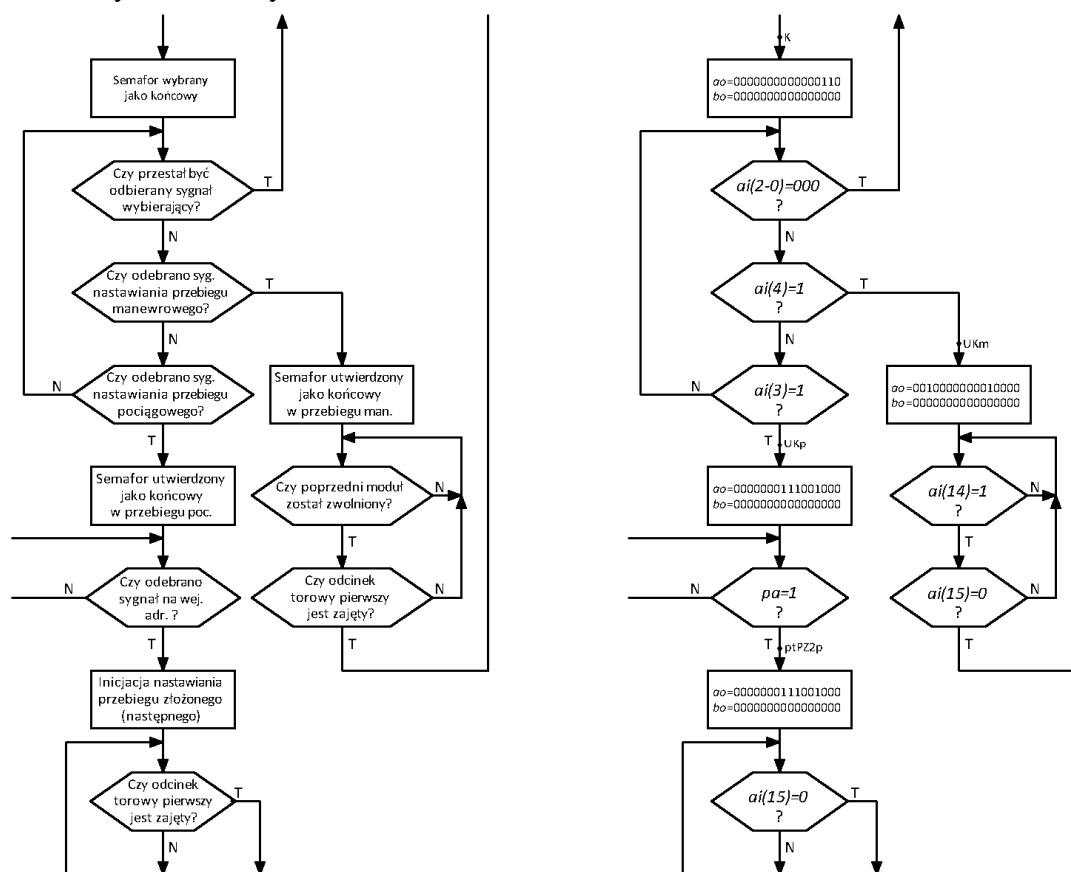
Rodzaj przebiegu (pociągowy lub manewrowy) określany jest odpowiednim poleceniem ogólnym. Wybór jednego z rodzajów prowadzi (po skontrolowaniu drogi) do utwierdzenia przebiegu (stany $U..$). Następnie, jeśli wykryta zostanie odpowiednia sekwencja zajmowania i zwalniania odcinków torowych oraz spełnione będą inne wymagane warunki (zwolnienie poprzedniego modułu, kontrola wyświetlenia sygnału zabraniającego), moduł rozpocznie proces zwalniania przebiegu (stan Z). Może się również zdarzyć, że nastawiony zostanie drugi przebieg (poprzedni lub następny) zawierający ten sam semafor, wtedy moduł przełączy się na pracę jako semafor pośredni w przebiegu złożonym (stan PZ).



Rys. 1. Schemat wejść i wyjść modułu (z lewej strony) i ogólny schemat działania modułu (z prawej strony).

Źródło: Opracowanie własne.

Na podstawie tak zapisanej ogólnej koncepcji stworzony został pełny algorytm działania modułu. Fragment zapisu formalnego tego algorytmu w postaci schematu blokowego przedstawiony został na rys. 2.



Rys. 2. Schemat blokowy algorytmu działania modułu SEM (w opisie słownym oraz po zakodowaniu zmiennych i oznakowaniu stanów).

Źródło: Opracowanie własne.

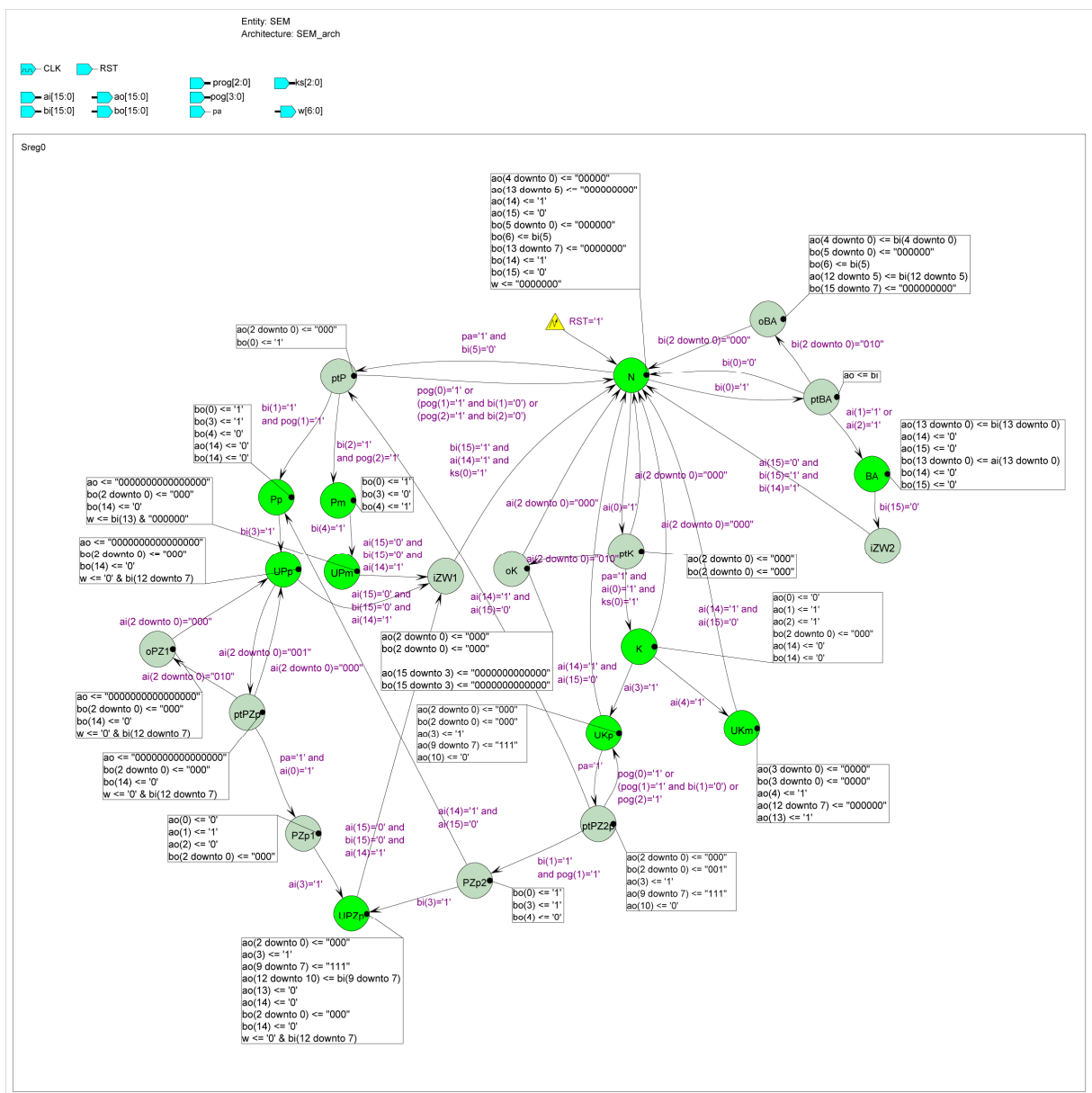
W kolejnym rozdziale opisany zostanie etap specyfikacji skonstruowanego algorytmu w języku opisu sprzętu. Model tworzony jest w języku VHDL przy użyciu pakietu wspomagania komputerowego *Active-HDL* firmy Aldec. Jest to popularny pakiet

oprogramowania do syntezy logicznej w strukturach programowalnych, wyposażony w edytory tekstowe oraz graficzne, jak również w symulatory logiczne [12].

4.2. Specyfikacja algorytmu w języku VHDL

Pakiet *Active-HDL* w zakresie tworzenia kodu języka oferuje trzy edytory. Oprócz edytora tekstowego, pozwalającego na pracę bezpośrednio w języku, dostępne są edytory graficzne. Edytor grafów przejść *FSM* pozwala na specyfikowanie przy pomocy grafów przejść automatu. Edytor schematów blokowych *BDE* pozwala powiązać fragmenty projektu wykonane wszystkimi edytorami.

Do specyfikacji zaprojektowanego algorytmu wykorzystany został edytor *FSM*, najbardziej intuicyjny w przypadku tworzenia układów sekwencyjnych. Wykonany w ten sposób graf przejść automatu widoczny jest na rys. 3.



Rys. 3. Specyfikacja algorytmu w edytorze *FSM*.

Źródło: Opracowanie własne.

Zasadniczą część specyfikacji stanowi odwzorowanie architektury układu. Widoczne są stany, w których może znajdować się moduł (zasadnicze, opisane wcześniej na rys. 1, oraz pomocnicze, konieczne do poprawnej pracy układu), tranzycje wraz z warunkami przejścia (w postaci równań logicznych) oraz przypisanie wartości sygnałom wyjściowym w każdym ze stanów. Ponad obszarem grafu znajduje się odwzorowanie jednostki projektowej, w którym zadeklarowane są wejścia i wyjścia.

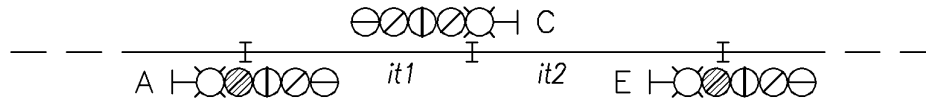
Specyfikacja wykonana w edytorze *FSM* może być wykorzystana na dalszych etapach projektowania: w edytorze tekstowym (kod języka generowany jest automatycznie) lub w edytorze schematów blokowych (jako pojedynczy element). Zaletą wykorzystania wspomaganie komputerowego jest możliwość projektowania układu za pomocą edytorów graficznych, w których praca jest przejrzysta i intuicyjna. Widoczne są zależności hierarchiczne. Wielokrotnie skraca to czas opracowania projektu, pozwalając poświęcić więcej uwagi funkcjonalności układu niż sprawdzaniu poprawności zapisania projektu. Samoczynne zarządzanie składnią języka ogranicza możliwość występowania błędów w kodzie. Pozostałe błędy, wynikające z nieprawidłowego wykonania lub wyspecyfikowania algorytmu, wyeliminować można dzięki symulatorom logicznym.

4.3. Weryfikacja i badania symulacyjne

Możliwość wykonania symulacji działania wyspecyfikowanego układu jest kolejną zaletą zastosowania pakietu wspomaganie komputerowego. Może się to odbywać w tym samym środowisku, co cały proces projektowania, pozwalając łatwo weryfikować poprawność specyfikacji już w trakcie jej tworzenia. Wykorzystanie wbudowanego symulatora logicznego polega na przypisaniu wymuszeń do zmiennych wejściowych i badaniu odpowiedzi układu. Wymuszeniami mogą być między innymi zegary predefiniowane, sygnały losowe, klawisze *Hotkey* oraz zaprogramowane sekwencje sygnałów. Pracę modelu podczas symulacji obserwować można bezpośrednio na grafach i schematach blokowych (graficznie) oraz na wykresie, przedstawiającym zmiany wartości zmiennych podczas symulacji. Pierwsza metoda pozwala weryfikować poprawność sekwencji zmian stanów. Przebiegi czasowe symulacji umożliwiają rejestrowanie stanu każdej zmiennej w dowolnie wybranej chwili.

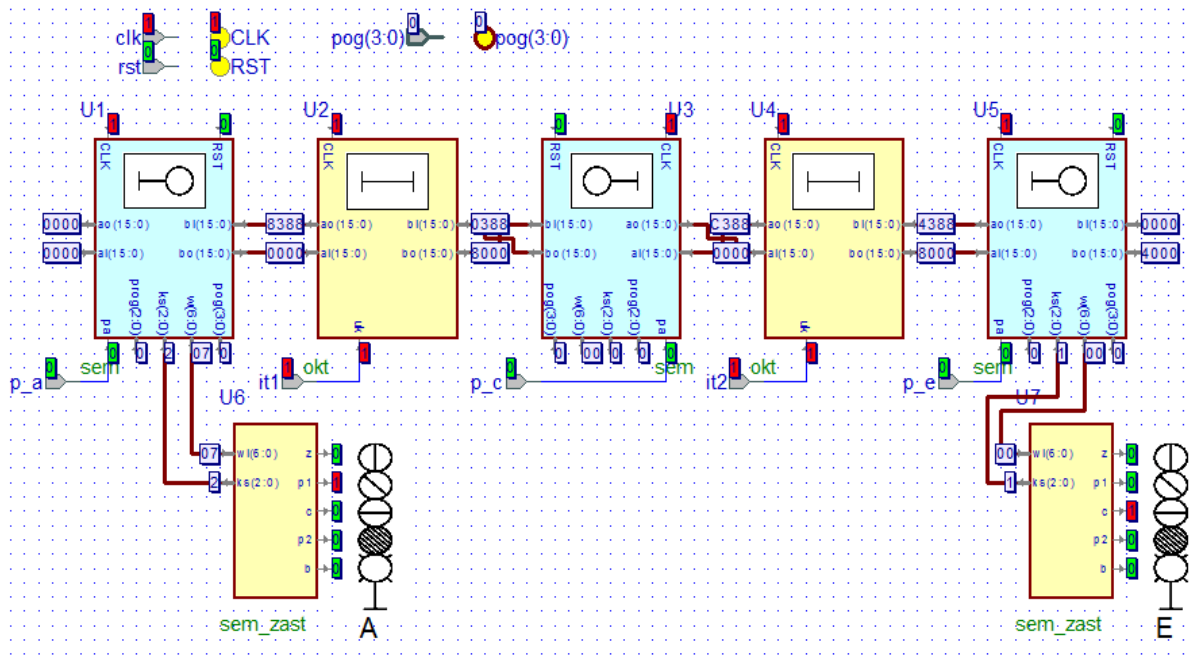
W pierwszej kolejności należy zweryfikować poprawność wykonanej specyfikacji. Weryfikacja pozwala na sprawdzenie, czy przejścia pomiędzy stanami zachodzą przy zgodnych z algorytmem wzbudzeniach, oraz czy wartości zmiennych wyjściowych w każdym ze stanów odpowiadają wartościom założonym przez projektanta. Ten etap projektowania pozwala wyeliminować błędy przypadkowe związane z tworzeniem specyfikacji. Pozytywny wynik weryfikacji modelu stanowi punkt wyjścia do przeprowadzenia symulacji działania założonych funkcji projektowanego modułu.

Jako ilustracja do badań wykonany został odcinek testowy, którego schemat przedstawiono na rys. 4. Składa się on z trzech semaforów (w tym jednego ustawionego w kierunku odwrotnym niż pozostałe) oraz odcinków torowych z kontrolą niezajętości. Dla takiego prostego przypadku zbudowany został w edytorze *BDE* schemat powiązania modułów odpowiadający zadanemu układowi torowemu. Oprócz trzech modułów *SEM* oraz dwóch modułów odcinka kontrolowanego *OKT* znalazły się w nim również moduły symulujące pracę semafora (*sem_zast*). Są to proste układy kombinacyjne, dekodujące (jedynie dla celów wizualizacyjnych) sygnały wychodzące z modułu logicznego semafora. Tak przygotowany model został poddany testom, pozwalając na analizę działania zaprojektowanego algorytmu w symulowanych sekwencjach zdarzeń.



Rys. 4. Schemat odcinka do badań symulacyjnych.

Źródło: Opracowanie własne.



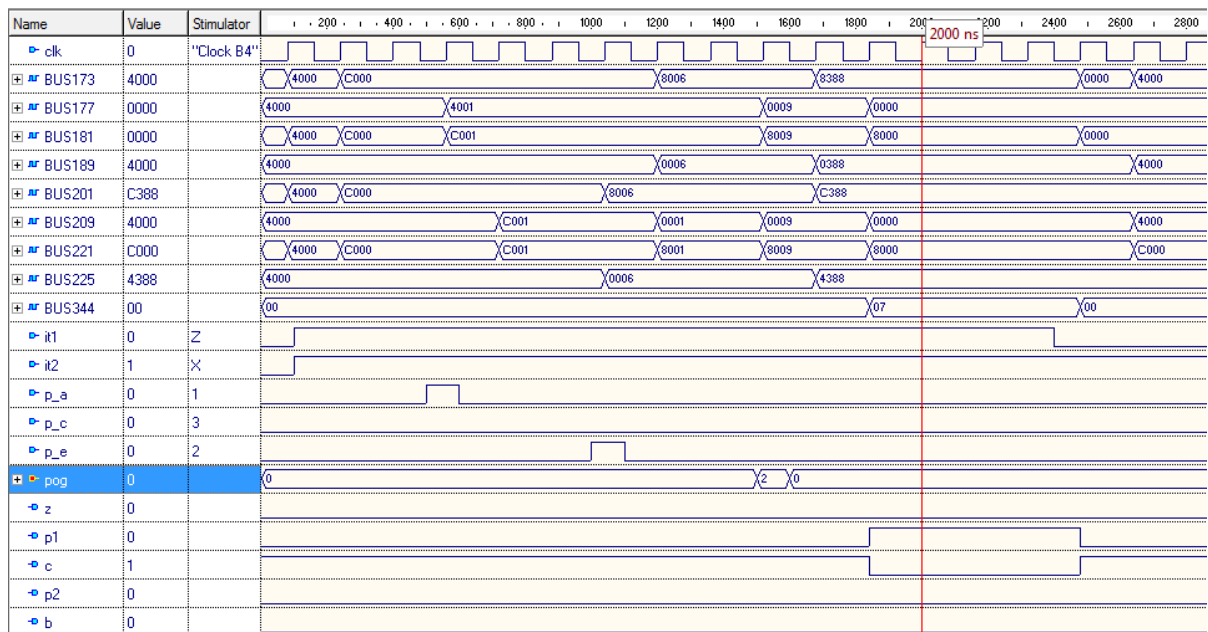
Rys. 5. Schemat blokowy połączeń modułów w edytorze BDE.

Źródło: Opracowanie własne.

Rys. 5 przedstawia schemat odcinka testowego w trakcie symulacji. Widoczne są wartości sygnałów na poszczególnych wejściach, wyjściach i liniach transmisyjnych. Zastosowano opis w postaci heksadecymalnej dla uzyskania większej czytelności. Na wyjściach układów *sem_zast* obserwować można sygnały wyświetlane na semaforach A oraz E (wartość '1' oznacza, że żarówka się świeci).

Jednocześnie wszystkie zdarzenia można obserwować na przebiegach symulacji (rys. 6). Rejestrowane są wartości sygnałów na wejściu zegarowym (*clk*), liniach transmisyjnych (*BUS*), wejściach odcinków kontrolowanych (*it*), wejściach adresujących (*p*), wejściu poleceń ogólnych (*pog*) oraz wyjściach do poszczególnych komór sygnalizatora A (*z*, *p1*, *c*, *p2*, *b*).

W przedstawionym przypadku obserwowana była praca modułów w trakcie nastawiania przebiegu od semafora A do semafora E. Sygnały na wejściach *p_a*, a następnie *p_e* inicjują przygotowanie drogi przebiegu, widoczne są sygnały przesyłane magistralami pomiędzy modułami. Nadanie polecenia wyboru przebiegu pociągowego (zmienna *pog*=2) powoduje utwierdzenie przebiegu oraz wyświetlenie sygnału zezwalającego na semaforze A (widoczne przejście zmiennej *c* w stan '0' oraz zmiennej *p1* w stan '1'). W ostatnim etapie testu następuje zajęcie odcinka za semaforem (zmienna *it1*), co prowadzi do wyświetlenia sygnału zabraniającego.



Rys. 6. Przebieg symulacji podczas nastawiania przebiegu.

Źródło: Opracowanie własne

Ponieważ symulację można prowadzić równoległe na przebiegach oraz we wszystkich edytorach graficznych, uzyskujemy pełny wgląd w zachowanie się projektowanego układu. W ten sposób, już na etapie tworzenia specyfikacji, projektant może mieć pewność, że efekt końcowy dokładnie odpowiada założeniom. Do etapu tworzenia prototypu dociera więc model zweryfikowany.

4.4. Dalsze etapy projektowania

Pakiet *Active-HDL* umożliwia przeprowadzenie syntezy i implementacji wykonanej i sprawdzonej specyfikacji. Brak konieczności zmiany środowiska programowania w trakcie procesu tworzenia minimalizuje ryzyko wystąpienia dodatkowych błędów. Co więcej, w przypadku wykrycia nieprawidłowości można niezwłocznie poprawić specyfikację i powtórnie dokonać weryfikacji. W zakresie implementacji i prototypowania pakiet ten współpracuje z wieloma współczesnymi układami programowalnymi. Te etapy, po uzupełnieniu modelu o brakujące moduły i funkcje, będą przedmiotem dalszych prac nad projektem.

PODSUMOWANIE

Metoda specyfikacji urządzeń srk zaproponowana w artykule posiada szereg cech, które ułatwiają projektowanie oraz minimalizują ryzyko popełnienia błędu. Proces specyfikacji odbywa się tu z wykorzystaniem wygodnych edytorów, pozwalających w pełni wykorzystać możliwości wspomaganie komputerowego. Zastosowanie symulatorów logicznych pozwala już na etapie projektu przetestować działanie układu przy dowolnie dobranych kombinacjach wymuszeń. Po zbadaniu poprawności wykonania specyfikacji możliwa jest dalsza praca nad modelem, prowadząca do powstania prototypu. Wszystkie te etapy mogą być przeprowadzone w jednym środowisku wspomaganie komputerowego, co sprawia, że zaproponowana metoda opisu spełnia wymagania stawiane poszukiwanej platformie projektowania systemów zabezpieczenia ruchu kolejowego.

COMPUTER AIDED SPECIFICATION OF RAILWAY INTERLOCKING FUNCTIONS USING HARDWARE DESCRIPTION LANGUAGES

Abstract

This paper presents several issues concerning computer aided specification of railway interlocking functions using hardware description languages. The review of currently used methods shows that none of them can provide a uniform design platform for railway applications, capable of static and dynamic verification of the design. The proposed design process is shown on an exemplary specification of a geographical interlocking module algorithm, written in VHDL. With the use of logic simulators it was possible to verify correctness of the description. The article presents various advantages of utilizing both hardware description languages and computer aided design software.

BIBLIOGRAFIA

1. Banci M., Fantechi A., Gnesi S.: *Some Experiences on Formal Specification of Railway Interlocking Systems using Statecharts*. SEFM, Koblenz 2005.
2. Dąbrowa-Bajon M.: *Podstawy sterowania ruchem kolejowym*. OWPW, Warszawa 2007.
3. Dobias R., Kubatova H.: *FPGA Based Design of the Railway's Interlocking Equipments*. EUROMICRO 2004.
4. Fokkink W., Groote J. F., Hollenberg M., van Vlijmen B.: *LARIS 1.0 Language for Railway Interlocking Specifications*. CWI, 2000.
5. Hlavatý T., Přeučil L., Štěpán P.: *Case Study: Formal Methods in Development and Testing of Safety-Critical Systems: Railway Interlocking System*. EUROMICRO 2001.
6. Kanso K., Moller F., Setzer A.: *Automated Verification of Signaling Principles in Railway Interlocking Systems*. Eighth International Workshop of Automated Verification of Critical Systems, Glasgow 2008.
7. Kawalec P., Koliński D.: *Modelowanie interlocking'u z zastosowaniem języka opisu sprzętu*. Logistyka 6/2010.
8. Kawalec P., Koliński D.: *Zastosowanie języka VHDL do badania złożonych sieci zestykowych*. Pomiary Automatyka Kontrola vol. 54, nr 8-2008, Wydawnictwo PAK, Warszawa, 2008, s. 529-531.
9. Kawalec P., Rzyśko M.: *Zastosowanie języków opisu sprzętu do specyfikacji urządzeń srk*. Logistyka 4/2012.
10. Mickiewicz T., Zubkow A.: *Przełącznikowe urządzenia nastawcze typu półblokowego*. WKiŁ, Warszawa 1961.
11. Miksza E., Olendrzyński W., Zubkow A.: *Zblokowany system sterowania ruchem kolejowym na stacjach typu IZH-111*. WKiŁ, Warszawa 1979.
12. Pasierbiński J., Zbysiński P.: *Układy programowalne w praktyce*. WKiŁ, Warszawa 2002.

Autorzy:

prof. nzw. dr hab. inż. Piotr KAWALEC – Politechnika Warszawska
inż. Marcin RZYSKO – Politechnika Warszawska