

Halina KAMIONKA-MIKUŁA
Politechnika Śląska, Instytut Informatyki

KOMPUTEROWE WSPOMAGANIE NAUCZANIA W ZAKRESIE UKŁADÓW PLD

Streszczenie. Ważnym elementem nauczania, w tym nauczania na odległość z wykorzystaniem sieci komputerowych, są programy edukacyjne. W przypadku nauczania zagadnień syntezy układów cyfrowych oprogramowanie powinno uwzględnić: opis zagadnień teoretycznych, przykłady syntezy, symulację pracy układów i możliwość sprawdzenia nabytej wiedzy. Wymienione funkcje programu dydaktycznego można uwzględnić w wyniku zastosowania różnych rozwiązań. Chcąc umożliwić prezentację syntezy i sprawdzanie wiedzy dla dowolnych danych wprowadzonych przez użytkownika, trzeba zaimplementować odpowiednie algorytmy dla komputerowej syntezy i dla komputerowej kontroli poprawności przebiegu syntezy realizowanej przez użytkownika. W pracy przedstawiono rozwiązania zastosowane w programach: *MatryceLogiczne* [2] *Spld* [3] utworzonych dla nauczania zagadnień programowanych matryc logicznych.

COMPUTER AIDED TEACHING OF PLD-BASED CIRCUITS

Summary. The educational programs are the important element of the teaching, especially remote teaching with the use of computer networks. If the teaching of digital circuits synthesis is the case the software should include: theoretical description, examples of synthesis, simulation of the circuit operation and the verification of new skills. Mentioned functions of the didactic program could be achieved by applying different methods. In order to make possible to present synthesis and to verify the new knowledge for any data chosen by the user, there is a need to implement the proper algorithms for computer performed synthesis and computer control of the synthesis performed by the user. In the paper presents the solution applied to programs *MatryceLogiczne* [2] and *Spld* [3] written for teaching of the PLD based circuits design.

1. Wprowadzenie

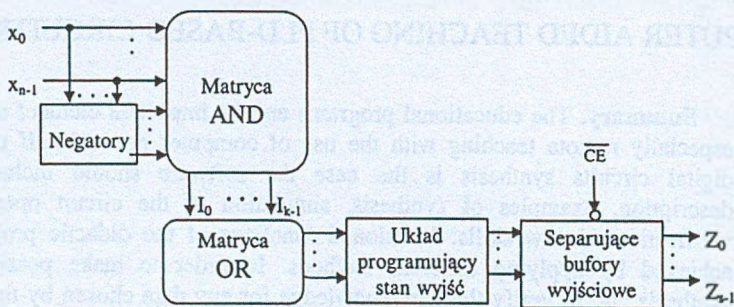
Ważnym elementem nauczania, w tym nauczania na odległość, są programy edukacyjne. Wymagania stawiane programom dydaktycznym są różne i zależą między innymi od tematyki nauczania. W przypadku nauczania zagadnień syntezy układów cyfrowych oprogramowanie to powinno uwzględnić: opis zagadnień teoretycznych, przykłady syntezy, symulacji pracy układu i możliwość sprawdzenia nabytej wiedzy.

W Instytucie Informatyki Politechniki Śląskiej od wielu lat prowadzone są prace, między innymi, w zakresie opracowania metod syntezy i analizy układów cyfrowych oraz algorytmów dla komputerowego wspomaganie nauczania i syntezy układów cyfrowych. Wyniki prac są wykorzystywane między innymi w programach dydaktycznych [4]. W bibliotece programów dydaktycznych są trzy programy [2, 4, 5] dla nauczania w zakresie układów PLD. W pracy przedstawiono rozwiązania zastosowane w dwóch programach: MatryceLogiczne i Spld.

2. Opis struktur układów PLD uwzględnionych w oprogramowaniu

2.1. Układ typu PLA

Układ PLA (Programmable Logic Array) składa się z programowalnej matrycy sum logicznych (OR), programowalnej matrycy iloczynów logicznych (AND) oraz układów ustalających postać funkcji wyjść tak, jak to przedstawia rys. 1.



Rys. 1. Schemat blokowy układu PLA

Fig. 1. Block diagram of Programmable Logic Array

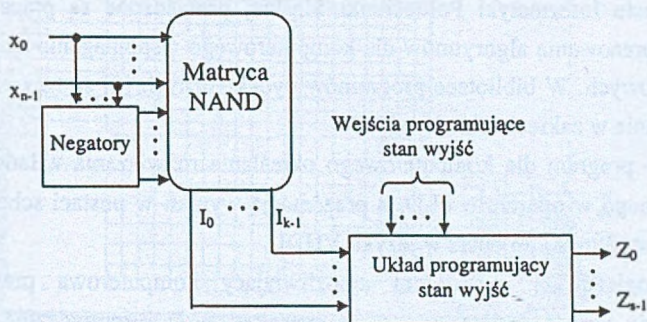
Obecnie są dostępne układy programowalne PLA uwzględniające wiele udogodnień dla użytkownika. Między innymi umożliwiają one dodatkowo programowanie części wyprowadzeń, aby umożliwić zadeklarowanie ich jako wejścia lub wyjścia i programowanie połączeń wyjść układu z wejściami matrycy AND.

Układ PLA pozwala na realizację funkcji logicznej zadanej w postaci alternatywnej (sumy) bądź koniunkcyjnej (iloczynu). Synteza układu w oparciu o PLA sprowadza się do określenia minimalnych wyrażeń opisujących układ w postaci alternatywnej lub postaci koniunkcyjnej.

Dla postaci alternatywnej matryca AND umożliwia realizację iloczynów jako składników sumy, a matryca OR realizację sumy odpowiednich iloczynów potrzebnych do realizacji funkcji. Wyjścia układu należy wtedy zaprogramować jako proste. W przypadku gdy realizacja funkcji przeciwnej do F jest prostsza, można zrealizować funkcję \bar{F} na wyjściach z matrycy OR, a następnie zaprogramować wyjścia jako zanegowane.

2.2. Układ typu PGA

Układ PGA (Programmable Gate Array) składa się z programowalnej matrycy zanegowanych iloczynów logicznych (NAND) oraz układów ustalających postać funkcji wyjść tak, jak to przedstawia rys.2. Algorytm syntezy układu kombinacyjnego przy użyciu układu PGA jest opisany w rozdziale 14 literatury [1].



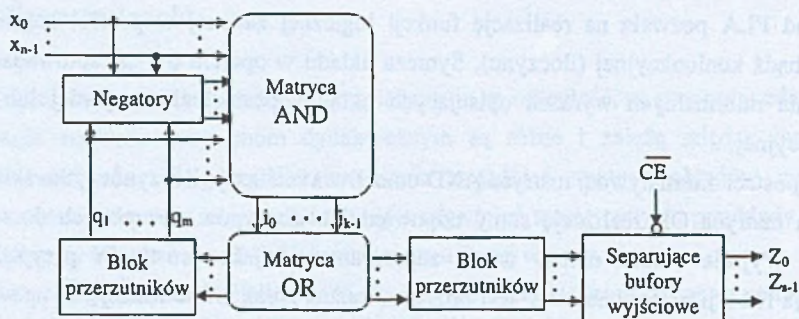
Rys. 2. Schemat blokowy układu PGA

Fig. 2. Block diagram of Programmable Gate Array

2.3. Układ typu PLS

Układ PLS (Programmable Logic Sequencer) w stosunku do układu PLA posiada dodatkowo przerzutniki synchroniczne lub asynchroniczne (rys. 3). Synteza¹ układu realizowanego w oparciu o strukturę PLS polega na określeniu wyrażeń o postaci alternatywnej dla wejść przerzutników sr jedną z metod, np. opisanych w rozdziale 9.5 [1].

¹ Metody określania funkcji wejść przerzutników sr opisane są w rozdz. 9.5 [1] i uwzględnione w programach: Fpls.zip Tkl-synt oraz PLD-TKL.zip dostępnych w witrynie internetowej: <http://zmitac.iinf.polsl.gliwice.pl>.



Rys. 3. Schemat blokowy układu PLS

Fig. 3. Block diagram of Programmable Logic Sequencer

3. Opis rozwiązań zastosowanych w programach wspomagających nauczanie w zakresie układów PLD

W Instytucie Informatyki Politechniki Śląskiej prowadzone są prace między innymi w zakresie opracowania algorytmów dla komputerowego wspomagania nauczania i syntezy układów cyfrowych. W bibliotece programów dydaktycznych [6] są trzy programy [2], [3], [4] dla nauczania w zakresie układów PLD:

- ☐ Spld [3] – program dla komputerowego określania rozwiązania układu sekwencyjnego realizowanego w oparciu o PLD, z prezentacją wyniku w postaci schematu logicznego wybranej struktury i jej opisu w języku VHDL,
- ☐ MatryceLogiczne [2] - program umożliwiający komputerową prezentację etapów rozwiązania układu realizowanego w oparciu o PLD i sprawdzanie wiedzy poprzez samodzielne rozwiązywanie zadań przez użytkownika z komputerową kontrolą przebiegu rozwiązania i możliwością tworzenia nowych prezentacji i zadań.
- ☐ Pld [4] - program umożliwiający komputerową prezentację etapów rozwiązania układu realizowanego w oparciu o PLD i rozwiązywania zadań przez użytkownika z kontrolą przebiegu rozwiązania, jednak bez możliwości tworzenia nowych prezentacji i zadań.

Dalej przedstawiono rozwiązania zastosowane w programach: MatryceLogiczne i SPLD.

3.1. Opis oprogramowania Spld

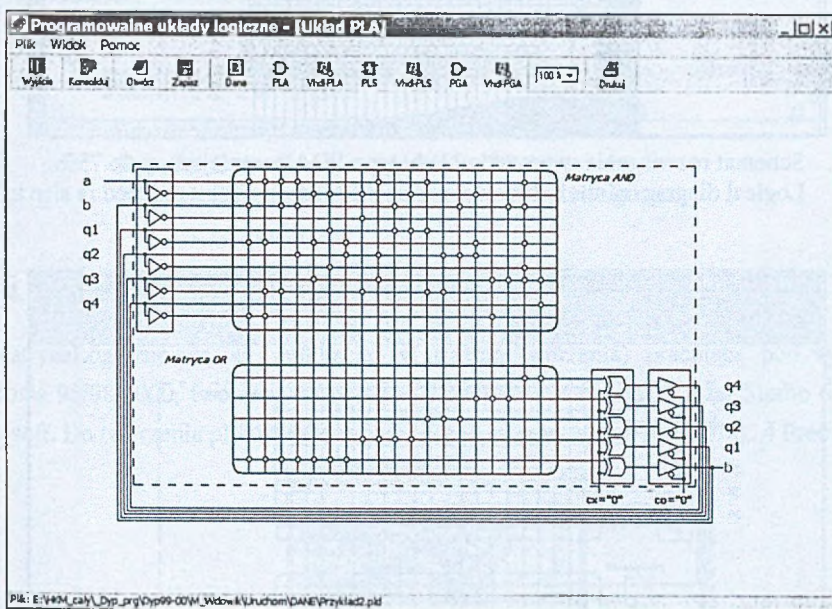
Program *Spld.exe* jest 32-bitową aplikacją pracującą pod systemem operacyjnym Windows 95/98/2000. Do stworzenia programu jako narzędzia programistycznego użyto Delphi 4. Pprogramy: Help & Manual oraz Microsoft Workshop posłużyły do stworzenia pliku pomocy.

Program Spld¹ współpracuje z programami:

- TKL, który dla danych wprowadzonych w postaci wykresu czasowego lub formuły łączy generuje wyniki w postaci kanonicznej i umożliwia prezentację kolejnych etapów rozwiązania w oparciu o Tablicę Kolejności Łączy (TKL),
- Kazakow, który dla danych wypracowanych przez program TKL generuje wyniki w postaci minimalnych wyrażeń logicznych: alternatywnych lub koniunkcyjnych.

W oparciu o skonsolidowane wyniki wygenerowane przez program TKL program Spld generuje rozwiązanie dla asynchronicznych układów sekwencyjnych realizowanych w oparciu o programowane matryce logiczne.

Prezentowane jest rozwiązanie w postaci schematu logicznego trzech typów układów: PLA, PGA i PLS (rys.: 4, 5, 6).

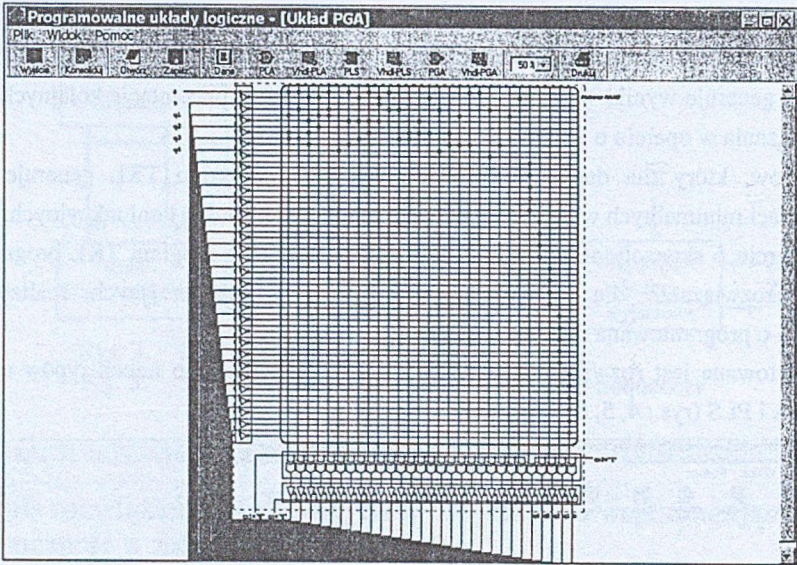


Rys. 4. Schemat rozwiązania w postaci układu typu PLA

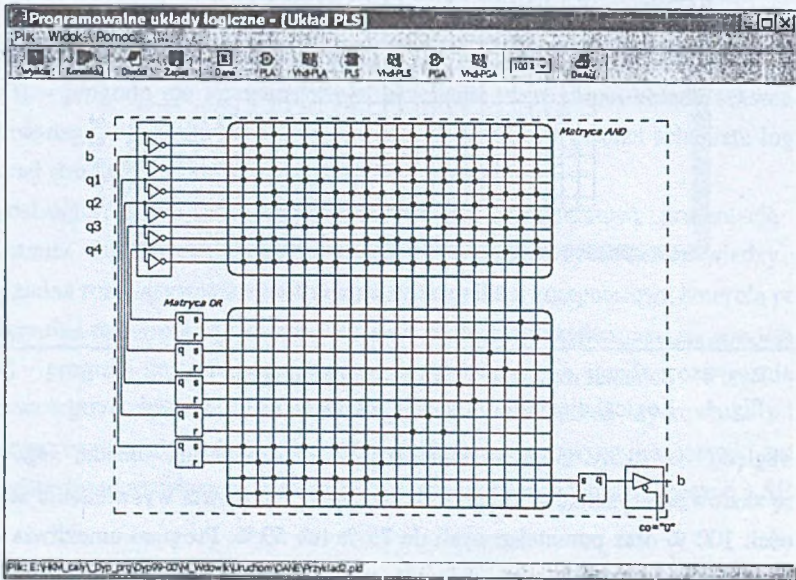
Fig. 4. Logical diagram of the solution given as PLD based circuit

Ze względu na różny stopień skomplikowania rozwiązania układu, jego rozmiar i potrzebę skalowania rysunków, ComboBox: umożliwia wyświetlanie schematów w wielkości: 100 % oraz pomniejszonych do 75 % lub 50 %. Program umożliwia też zapis po pliku rysunku przedstawiającego schemat rozwiązania i wydruk schematu. W programie, oprócz danych wejściowych i schematu rozwiązania, prezentowany jest również opis struktury rozwiązania w języku VHDL służącym do projektowania układów scalonych.

¹ Program Spld jest dostępny w stronie internetowej <http://zmitac.iinf.polsl.gliwice.pl>



Rys. 5. Schemat rozwiązania w postaci układu typu PGA pomniejszony do 75%
 Fig. 5. Logical diagram of the solution given as PGA based circuit reduced in size to 75%



Rys. 6. Schemat rozwiązania w postaci układu typu PLS
 Fig. 6. Logical diagram of the solution given as PLS based circuit

Rysunek 7 przedstawia okno zawierające fragment opisu w języku VHDL dla struktury układu typu PLA pokazanej na rys. 4.

```

Programowalne układy logiczne - [Dane wyjściowe - Vhdl-PLA]
Plik - Widok - Pomoc
Wydruk Komentuj Odczytaj Zapisać Drukuj
PLA Vhdl-PLA PLS VhdlPLS PGA VhdlPGA 100%1 Drukuj
Opis struktury PLA w języku VHDL
LIBRARY IEEE;
USE IEEE.Std_Logic_1164_A1;

ENTITY aPrzyklad2 IS
    PORT (
        i0 IN STD_ULOGIC;
        i1 IN STD_ULOGIC;
        i2 IN STD_ULOGIC;
        i3 IN STD_ULOGIC;
        i4 IN STD_ULOGIC;
        i5 IN STD_ULOGIC;
        i6 IN STD_ULOGIC;
        o0 OUT STD_ULOGIC;
        o1 OUT STD_ULOGIC;
        o2 OUT STD_ULOGIC;
        o3 OUT STD_ULOGIC;
        o4 OUT STD_ULOGIC;
    );
END aPrzyklad2;

ARCHITECTURE aPrzyklad2 OF aPrzyklad2 IS
    SIGNAL MaAND : STD_ULOGIC_VECTOR(3 DOWNTO 0);
    SIGNAL MaOR : STD_ULOGIC_VECTOR(3 DOWNTO 0);
    SIGNAL MaANDn : STD_ULOGIC_VECTOR(3 DOWNTO 0);
    SIGNAL MaORn : STD_ULOGIC_VECTOR(3 DOWNTO 0);

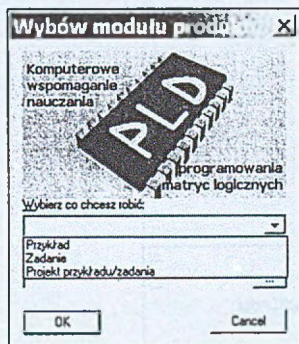
BEGIN
    MaAND(0) <= i0 AND i1 AND NOT i2 AND NOT i3;
    MaAND(1) <= i0 AND NOT i1 AND NOT i2 AND NOT i3;
    MaAND(2) <= i0 AND i1 AND NOT i2;

```

Rys. 7. Opis układu w języku VHDL
Fig. 7. Description of the circuit in VHDL language

3.2. Opis oprogramowania MatryceLogiczne


MatryceLogiczne.exe są aplikacją (w trakcie tworzenia) pracującą pod systemem Windows 95/98/2000, tworzoną za pomocą Visual C++ z pakietu Visual Studio 6.0 firmy Microsoft. Do tworzenia plików w formacie HTML wykorzystano AceHTML 4 Freeware.



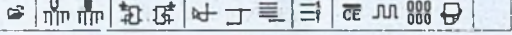
Rys. 8. Okno logowania
Fig. 8. Login window

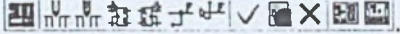
Program umożliwia: prezentację rozwiązań dla danych wprowadzonych przez użytkownika, testowanie działania struktur oraz sprawdzanie wiedzy poprzez umożliwienie budowy struktur układów PLA i PLS i programowanie matryc z kontrolą poprawności.

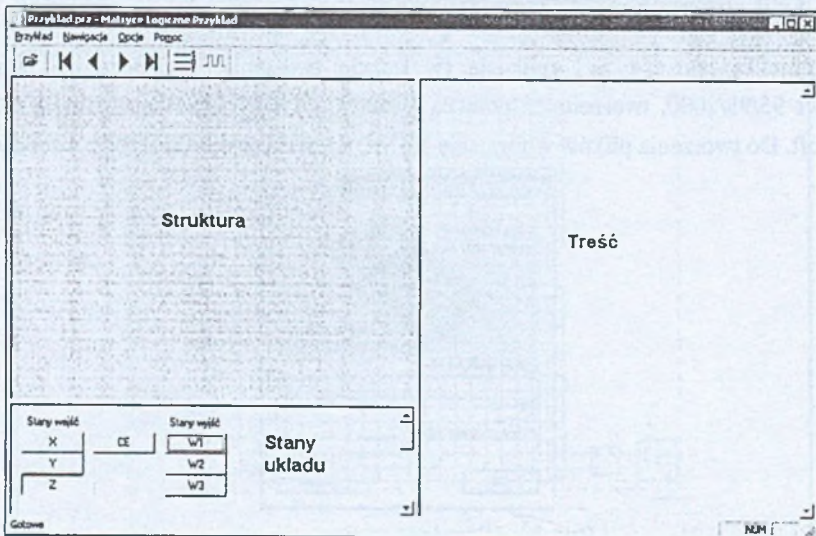
W programie uwzględniono moduły: Przykład, Zadanie, Projekt przykładu/zadania. Wyboru akcji dokonuje się w okienku logowania (rys. 8).

Po wybraniu modułu Przykład okno ma postać widoczną na rys. 9. W części: „Struktura” przedstawia się schemat prezentowanej struktury, „Treść” prezentuje treść tekstową wykładu. Część „Stany układu” zawiera przyciski dla obrazowania stanu sygnałów układu. Stan sygnałów wyjściowych jest wyliczany po każdorazowej zmianie stanu wejść. Prezentację umożliwiają funkcje z menu „Nawigacja” i paska narzędzi: 

Na prezentację składają się slajdy: statyczne ilustrujące zagadnienia teoretyczne oraz interaktywne umożliwiające śledzenie zmian sygnałów wejściowych i wyjściowych (dla układów w pamięć, również sygnałów stanu wewnętrznego) układu.

Moduł Zadanie (rys. 10) umożliwia edycję układu PLD o strukturze zadanej tematem z wykorzystaniem funkcji dostępnych w menu „Edytuj układ” i na pasku narzędzi: . Możliwe jest testowanie budowanego układu za pomocą funkcji dostępnych w opcji „Akcja”.

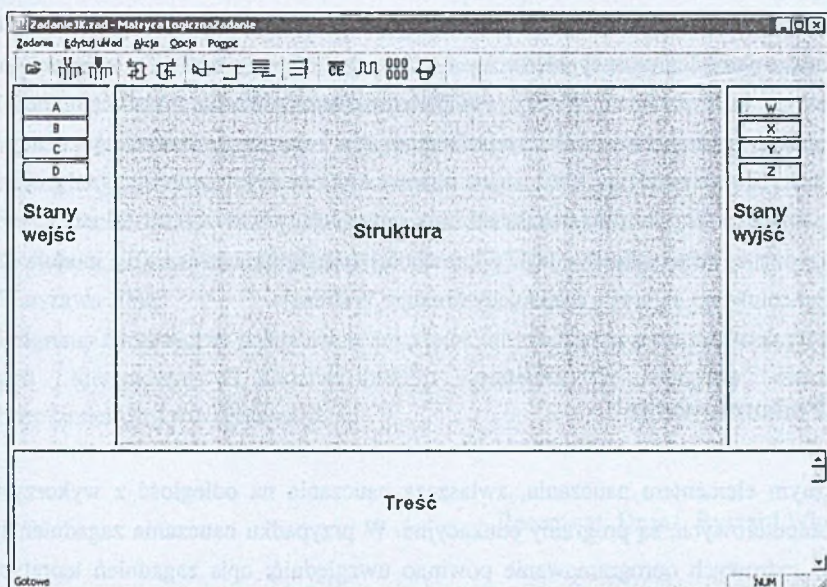
Moduł Projekt (rys. 11) jest narzędziem do tworzenia prezentacji i zadań. Podobnie jak moduł Zadania, udostępnia on funkcje do edycji i testowania struktury układu za pomocą funkcji dostępnych w pasku: .



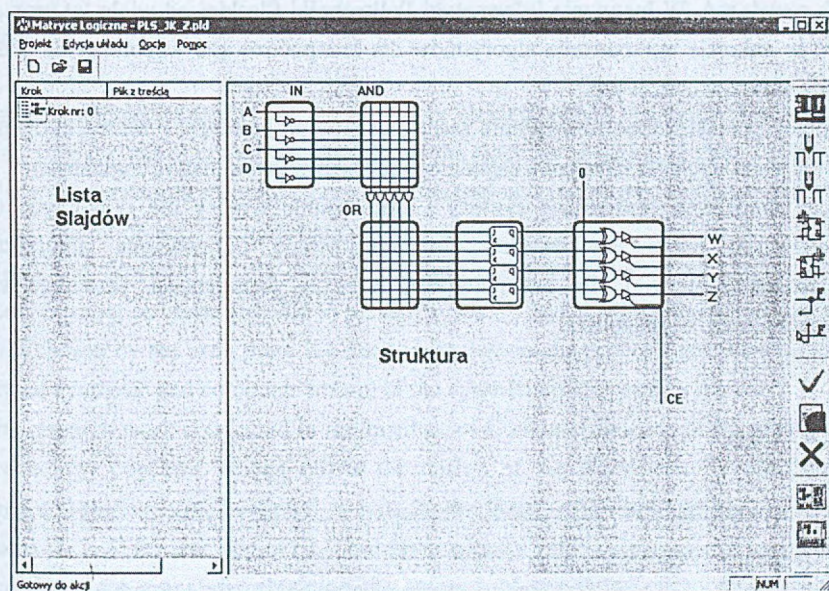
Rys. 9. Okno główne programu w module Przykład

Fig. 9. Main window of the program in module Przykład

W modułach: Zadania i Projekt edycja polega na wyborze: elementów składowych struktury, sposobu ich połączenia oraz stanu zaprogramowania matrycy logicznych.



Rys. 10. Okno główne programu w module Zadania
 Fig. 10. Main window of the program in module Zadania



Rys. 11. Okno główne programu w module Projekt
 Fig. 11. Main window of the program in module Projekt

Do umożliwienia symulacji pracy układu zaimplementowano między innymi algorytmy [2]: realizujące funkcje: PodajWartosc(int numer) - która oblicza stan linii

wyjściowej o indeksie przekazywanym jako parametr numer, `PodajWartosc(int Wiersz)` - która oblicza sumę logiczną sygnałów wychodzących z matrycy AND dla wiersza o indeksie przekazanym jako parametr `Wiersz`, `PodajWartosc(int Kolumna)` - która oblicza iloczyn logiczny linii wyjściowych bloku wejściowego, dla kolumny o indeksie przekazywanym parametrem `Kolumna`, `PodajWyjscie(int numer)` - która przekształca sygnały wejściowe z linii zewnętrznych, pamięci układu lub linii zwracających w zależności od postaci linii: prosta lub zanegowana. Do kontroli poprawności rozwiązania zadania w module `Zadania` zaimplementowano algorytm realizujący funkcję: `Walidacja`.

4. Podsumowanie

Ważnym elementem nauczania, zwłaszcza na odległość z wykorzystaniem sieci komputerowych, są programy edukacyjne. W przypadku nauczania zagadnień syntezy układów cyfrowych oprogramowanie powinno uwzględnić: opis zagadnień teoretycznych, przykłady syntezy, symulację pracy układów i możliwość sprawdzenia nabytej wiedzy. Wymienione funkcje programu dydaktycznego można uwzględnić w wyniku zastosowania różnych rozwiązań. W Instytucie Informatyki Politechniki Śląskiej od wielu lat prowadzone są prace w zakresie opracowania algorytmów dla komputerowego wspomaganie nauczania i syntezy układów cyfrowych.

W pracy przedstawiono rozwiązania zastosowane w programach: `i MatryceLogiczne` [2] `Spld` [3] utworzonych dla nauczania zagadnień programowanych matryc logicznych.

Chcąc umożliwić prezentację syntezy i sprawdzanie wiedzy dla dowolnych danych wprowadzonych przez użytkownika, zimplementowano odpowiednie algorytmy dla komputerowej syntezy i dla komputerowej kontroli poprawności przebiegu syntezy realizowanej przez użytkownika.

LITERATURA

1. Kamionka-Mikuła H., Małysiak H., Pochopień B.: Układy cyfrowe – teoria i przykłady. Wydanie III rozszerzone, PJS, Gliwice 2001.
2. Kosowski R.: Komputerowe wspomaganie nauczania w zakresie programowanych matryc logicznych. Praca dyplomowa magisterska, realizowana pod kierunkiem H. Kamionki-Mikuły w Instytucie Informatyki Politechniki Śląskiej, Gliwice 2002.

3. Wdowik M.: Komputerowe wspomaganie syntezy układów sekwencyjnych. Praca dyplomowa magisterska, wykonana pod kierunkiem H. Kamionki-Mikuły w Instytucie Informatyki Politechniki Śląskiej, Gliwice 2000.
4. Roszkowski K.: Komputerowe wspomaganie nauczania w zakresie programowanych modułów logicznych. Praca dyplomowa magisterska, wykonana pod kierunkiem H. Kamionki-Mikuły w Instytucie Informatyki Politechniki Śląskiej, Gliwice 2000.
5. Pieńkos J., Turczyński J.: Układy scalone TTL w systemach cyfrowych. WKiŁ. Warszawa 1986.
6. Programy dydaktyczne wykonane w Instytucie Informatyki w ramach prac dyplomowych pod kierunkiem H. Kamionki-Mikuły dostępne w witrynie internetowej: <http://zmitac.iinf.polsl.gliwice.pl>

Recenzent: Dr inż. Ryszard Winiarczyk

Wpłynęło do Redakcji 18 kwietnia 2002 r.

Abstract

The educational programs are the important element of the teaching, especially remote teaching with the use of computer networks. If the teaching of digital circuits synthesis is the case the software should include: theoretical description, examples of synthesis, simulation of the circuit operation and the verification of new skills.

Mentioned functions of the didactic program could be achieved by applying different methods. In order to make possible to present synthesis and to verify the new knowledge for any data chosen by the user, there is a need to implement the proper algorithms for computer performed synthesis and computer control of the synthesis performed by the user.

The research work conducted in the Institute of Computer Science of Silesian University of Technology concerns among others the design of the algorithms for computer aided teaching and synthesis of the digital circuits. In the library of didactic programs [6] are three programs [2, 3, 4] for teaching of PLD-based circuits:

- Spld [3] – for automatic obtaining the solution of sequential circuit implemented as a PLD-based system with the presentation of the solution as a logical diagram (Fig. 4, 5, 6) of the chosen structure and as the description of it in VHDL language (Fig. 7),
- MatryceLogiczne [2] – giving the possibility of computer presentation of the stages included in the design of the PLD-based circuit (Fig 9) and knowledge verification by

- allowing the solving the task by the user with computer control of the whole process (Fig. 10) and the possibility of generating of new presentations and tasks (Fig. 11),
- Pld [4] - giving the possibility of computer presentation of the stages included in the design of the PLD-based circuit (Fig 9) and knowledge verification by allowing the solving the task by the user with computer control of the whole process, however without the possibility of generating of new presentations and tasks.

In the paper presents the solution applied to programs MatryceLogiczne [2] and Spld [3] written for teaching of the PLD based circuits design.