

LAB4: Układy arytmetyczne

I. CEL ĆWICZENIA

Celem ćwiczenia jest utrwalenie wiedzy dotyczącej budowy i działania układów arytmetycznych oraz zapoznanie z metodyką projektowania takich układów. Podczas ćwiczenia dokonuje się implementacji układowej z użyciem wybranych cyfrowych układów scalonych zawierających bramki, przerzutniki oraz bloki sumatorów.

II. WPROWADZENIE

SUMATORY

Układy kombinacyjne, których stany wyjściowe odpowiadają wartościom sumy arytmetycznej stanów wejściowych traktowanych jako liczby w zapisie wagowym. Liczby mogą być binarne albo w kodzie BCD.

SUBTRAKTORY

Układy kombinacyjne, których stany wyjściowe odpowiadają wartościom odejmowania arytmetycznego stanów wejściowych traktowanych jako liczby w zapisie wagowym. Liczby mogą być zapisane z bitem znaku lub bez, zazwyczaj występują w kodzie binarnym przy czym wartości ujemne są kodowane dopełnieniowo.

KOMPARATORY CYFROWE

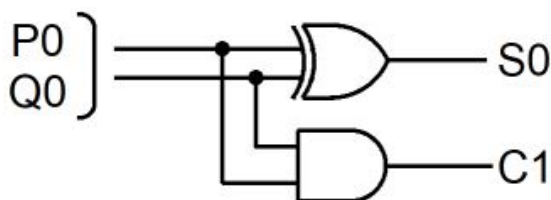
Układy kombinacyjne, których sygnały wyjściowe przyjmują stan aktywny dla spełnionej relacji porównania wejść traktowanych jako liczby, np.: $A=B$, $A \neq B$, $A > B$, $A < B$, $A \geq B$, $A \leq B$.

GENERATORY (NIE-)PARZYSTOŚCI

Układy kombinacyjne, których sygnał wyjściowy przyjmuje stan aktywny dla parzystej lub nieparzystej liczby stanów w słowie sygnału wejściowego. Zazwyczaj stanowi bit uzupełnienia do parzystej liczby stanów w słowie wejściowym.

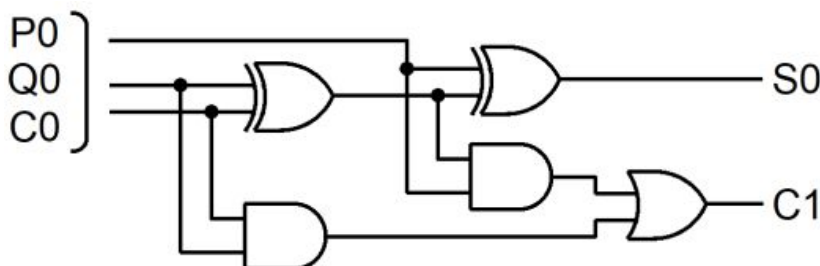
PÓLSUMATOR JEDNOBITOWY

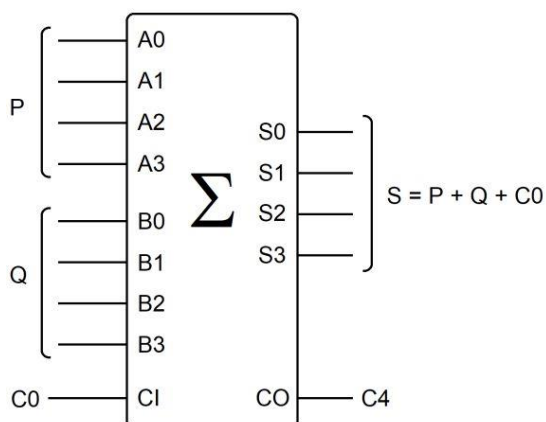
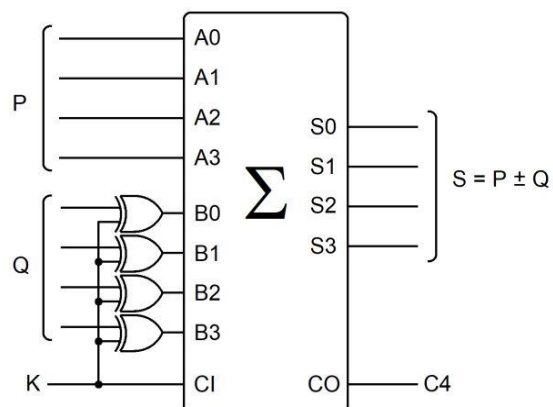
Schemat na bramkach logicznych:



SUMATOR PEŁNY JEDNOBITOWY

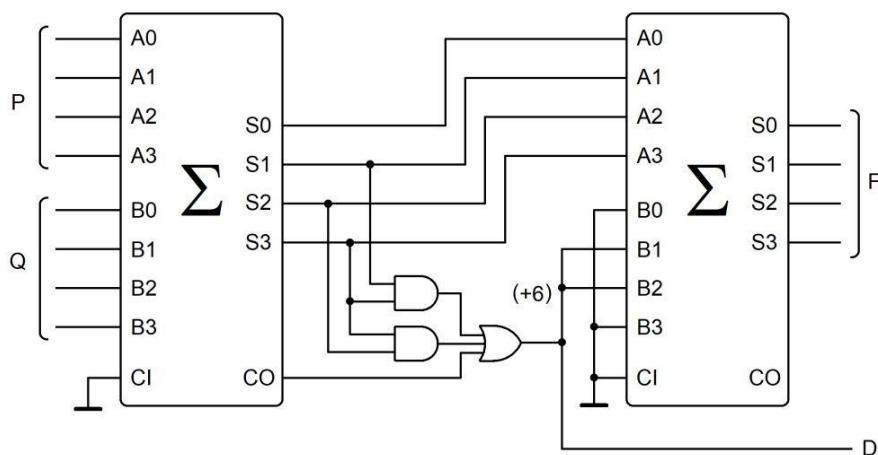
Schemat na bramkach logicznych:



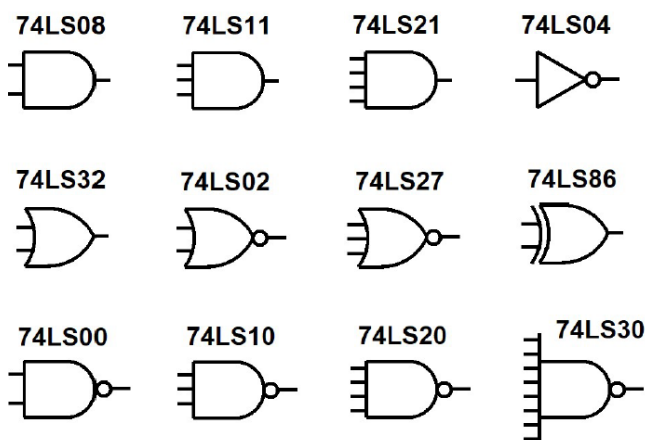
SCALONY SUMATOR
CZTEROBITOWY 7483Symbol graficzny
i oznaczenia wyprowadzeń:SUMATOR / SUBTRAKTOR CZTEROBITOWY
W KODZIE U2Schemat z użyciem scalonego sumatora
czterobitowego 7483 oraz bramek logicznych:

SUMATOR BCD

Schemat z użyciem dwóch scalonych sumatorów czterobitowych 7483 oraz bramek logicznych:



OZNACZENIA ELEMENTÓW LOGICZNYCH W ŚRODOWISKU MULTISIM



III. REALIZACJA ĆWICZENIA LABORATORYJNEGO

STANOWISKO KOMPUTEROWE

Ćwiczenie wykonywane jest na stanowiskach komputerowych z zainstalowanym systemem operacyjnym Windows i środowiskiem projektowo-symulacyjnym *Multisim 14.2* firmy *National Instruments*.

WYKONANIE ĆWICZENIA

Pierwsza część ćwiczenia to projekt wstępny, w którym symulowany jest prosty układ arytmetyczny. W drugiej części wykonywane są indywidualne zadania projektowe, przedstawione przez prowadzącego ćwiczenie.

UWAGA! Wszystkie zadania projektowe należy zapisywać w następującej lokalizacji:

C:\Labuprog\UC\Lab4\

Przed przystąpieniem do ćwiczenia należy utworzyć stosowny folder.

OPRACOWANIE WYNIKÓW

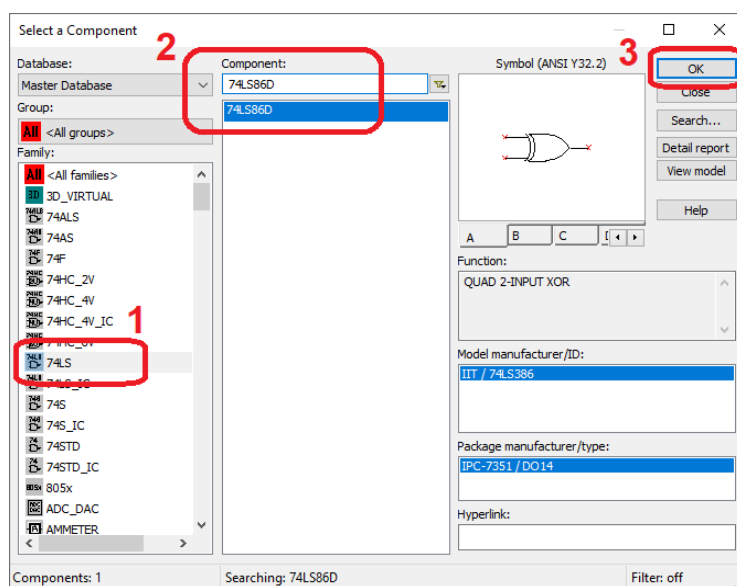
1. Sprawozdanie z ćwiczenia laboratoryjnego powinno zawierać zwięzłe opisy postawionych zadań projektowych wraz z rozwiązaniami, w szczególności z tablicami przejść i schematami zaprojektowanych układów.
2. Do sprawozdania należy dołączyć zrzuty ekranów z wynikami przeprowadzonych symulacji.
3. W sprawozdaniu należy również zawrzeć rozwiązania dodatkowych problemów projektowych postawionych przez prowadzącego ćwiczenie.

ZALICZENIE ĆWICZENIA

1. Zaliczenie kolokwium wstępnego oraz poprawne wykonanie zadań laboratoryjnych postawionych przez osobę prowadzącą ćwiczenie.
2. Złożenie sprawozdania, zawierającego zwięzły opis wykonanych zadań, wnioski i poprawne odpowiedzi na postawione pytania.

PROJEKT WSTĘPNY

W ramach projektu wstępnego dokonamy symulacji układu półsumatora jednobitowego. Po uruchomieniu środowiska *Multisim* domyślnie utworzonemu projektowi (**Design1**) nadajemy nową nazwę, np. **samouczek**. Następnie z paska menu wybieramy **Place→Component...** (Ctrl+W). Otwiera się okienko **Select a Component** (rys.1), w którym w polu **Family**: wybieramy rodzinę cyfrowych układów scalonych **74LS**, a w polu **Component**: wprowadzamy symbol układu scalonego **74LS86D** (cztery bramki dwuwejściowe XOR). Naciskamy **OK**, aby zamknąć okienko komponentów.



Rys. 1 Okienko Select a Component z zaznaczonym układem scalonym 74LS86D

Domyślnie wracamy do okna schematu, w którym pozostaje aktywny pasek wyboru elementów zawartych w układzie scalonym **74LS86D**. Wybieramy bramkę **A**. Pojawia się symbol graficzny bramki. Poprzez jednokrotne kliknięcie lewym klawiszem myszki umieszczamy symbol na schemacie. Ponownie pojawia się pasek wyboru elementów układu **74LS86D**. Naciskamy **Cancel**, ponieważ tylko jedna bramka XOR jest potrzebna.

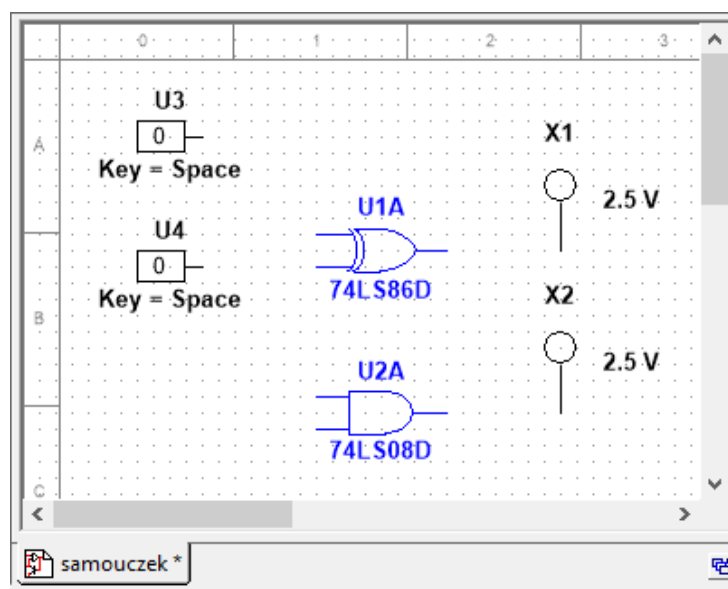
Symulator domyślnie utrzymuje otwarte okienko **Select a Component**, zachęcając do dodania kolejnych elementów. Zatem pozostawiamy wybraną rodzinę **74LS**, a w polu **Component**: wprowadzamy **74LS08D** (cztery bramki dwuwęściowe AND) i naciskamy **OK**. Umieszczamy na schemacie symbol bramki AND pod symbolem bramki XOR.

Do ustawiania stanów logicznych na wejściach półsumatora jednobitowego zastosujemy przyciski. W tym celu w polu **Group**: z rozwijanego menu wybieramy **Sources**. Następnie w polu **Family**: klikamy **DIGITAL_SOURCES**. Pojawiają się komponenty, wybieramy **INTERACTIVE_DIGITAL_CONSTANT**. Na schemacie umieszczamy jeden taki element po lewej stronie bramek. Nazwą elementu jest **U3**. Umieszczamy pod symbolem elementu **U3** jeszcze drugi element, który przyjmuje nazwę **U4**. W dalszej części zamiast **elementów U3 i U4** jest mowa o **przyciskach U3 i U4**.

Do obserwacji stanów logicznych na wyjściach budowanego układu służą wskaźniki stanów logicznych. Wybieramy **Indicators** w polu **Group** i klikamy **PROBE** w polu **Family**. Następnie wybieramy jeden z poniższych komponentów:

- **PROBE_DIG_BLUE** – wskaźnik w kolorze niebieskim,
- **PROBE_DIG_GREEN** – wskaźnik w kolorze zielonym,
- **PROBE_DIG_ORANGE** – wskaźnik w kolorze pomarańczowym,
- **PROBE_DIG_RED** – wskaźnik w kolorze czerwonym,
- **PROBE_DIG_YELLOW** – wskaźnik w kolorze żółtym.

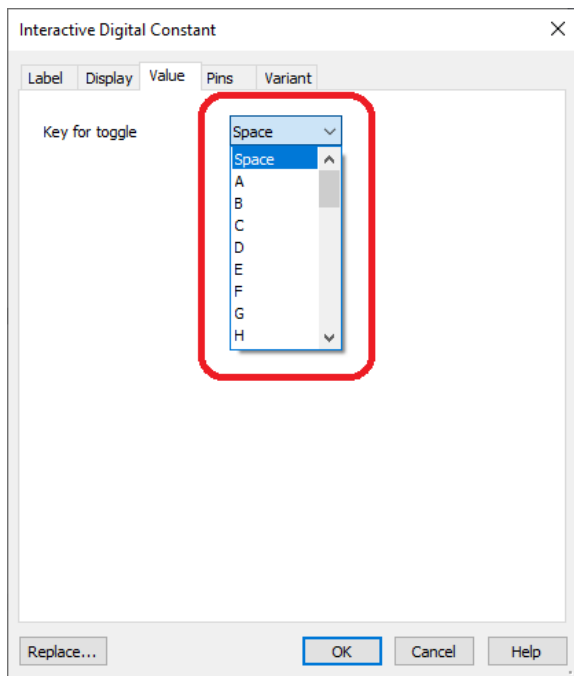
Na schemacie z prawej strony bramek umieszczamy symbol wskaźnika. Symulator przyznaje wskaźnikowi nazwę **X1**. Pod symbolem wskaźnika **X1** umieszczamy jeszcze drugi taki element lub umieszczamy symbol kolejnego wskaźnika. Nazwa wskaźnika to teraz **X2**. Zamykamy okienko z komponentami poprzez **Close**. Wszystkie elementy powinny być rozmieszczone jak na rysunku 2.



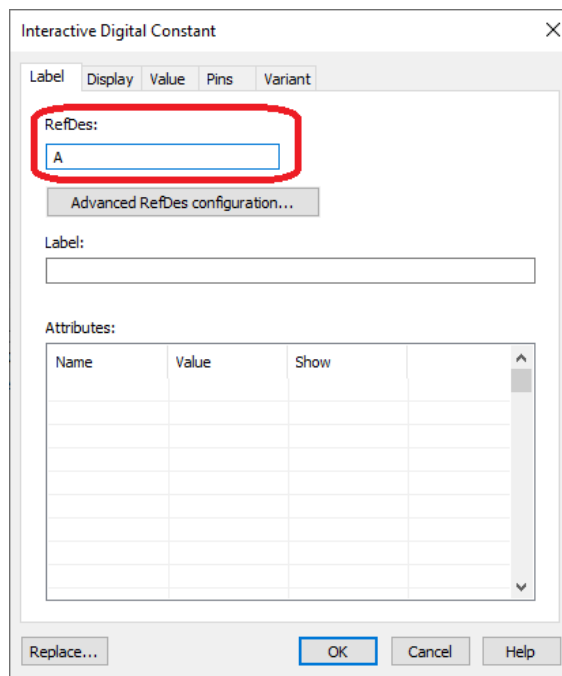
Rys. 2 Schemat z rozmieszczonymi bramkami XOR i AND, przyciskami U3 i U4, oraz wskaźnikami stanów logicznych X1 i X2

Wskaźnikom stanów logicznych nadajemy nazwy wyjść półsumatora jednobitowego: **SUM** i **CARRY**. Ustawiamy się kursorem na symbol wskaźnika **X1** i lewym klawiszem myszy klikamy dwukrotnie. Otwiera się okienko właściwości, w którym w zakładce **Label** wpisujemy nazwę **SUM**. Naciskamy **OK** i powtarzamy opisane czynności dla wskaźnika **X2**, który nazywamy **CARRY**.

Stany logiczne na wyjściach przycisków **U3** i **U4** są domyślnie przełączane klawiszem **Space** (spacja). Wybierzmy inne klawisze dla tych przycisków. Ustawiamy kursor na symbolu przycisku **U3** i lewym klawiszem myszy klikamy dwukrotnie. Otwiera się okienko **Interactive Digital Constant**, w którym: w zakładce **Value** z rozwijanego menu wybieramy **A** (rys.3). Następnie w zakładce **Label** wprowadzamy nazwę **A** (rys.4). Zamykamy okienko poprzez **OK**. Analogicznie postępujemy dla przycisku **U4**: wybieramy klawisz **B** i nadajemy nazwę **B**.



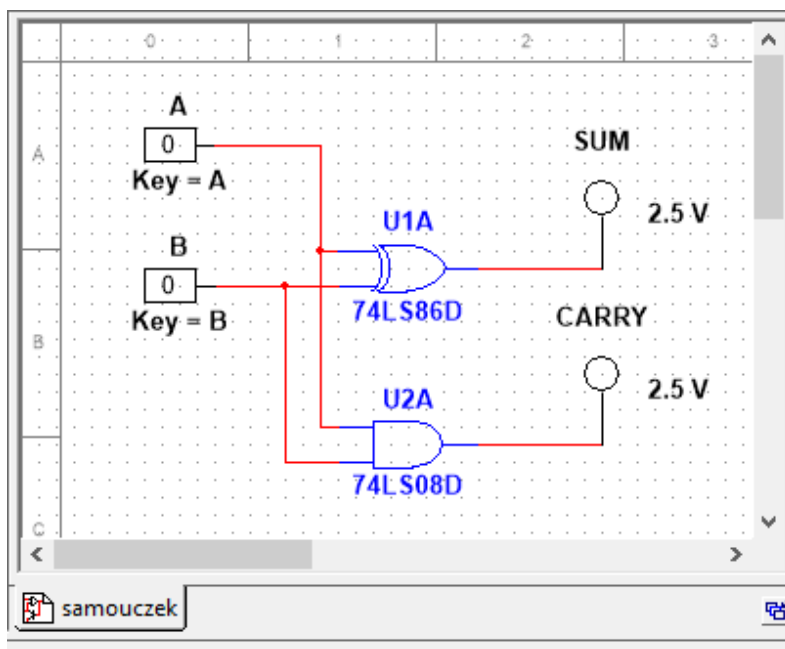
Rys. 3 Zakładka Value w okienku Interactive Digital Constant



Rys. 4 Zakładka Label w okienku Interactive Digital Constant

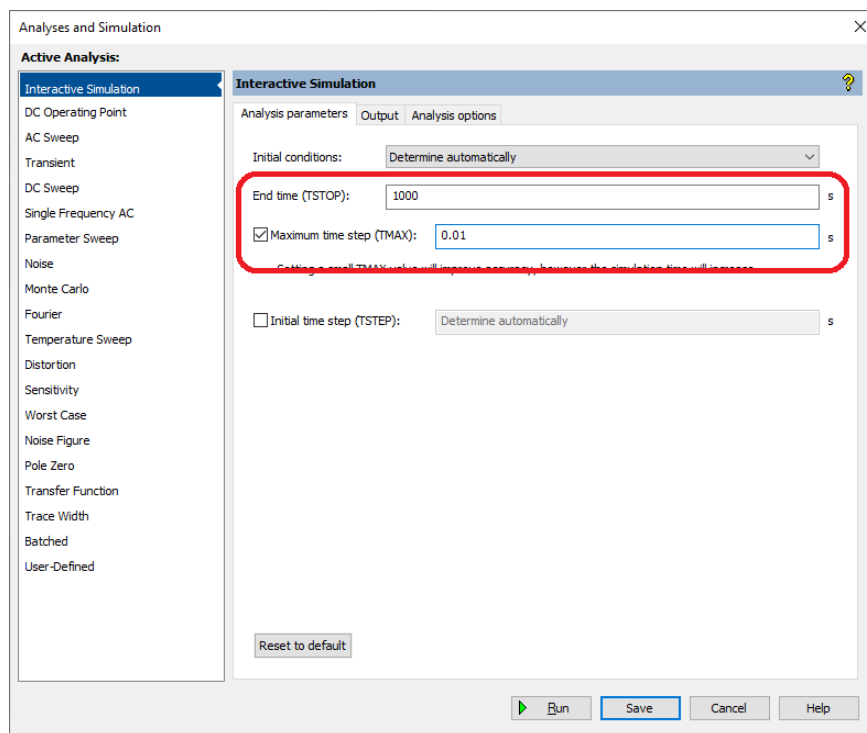
W celu poprowadzenia połączeń należy ustawić kursor na wybranej końcówce oraz kliknąć lewym klawiszem myszki. Następnie ustawić kursor na kolejnej końcówce i zakończyć rysowanie połączenia pojedynczym kliknięciem. Zostanie automatycznie poprowadzone połączenie, ale można modyfikować jego kształt. Usuwanie połączenia dokonujemy poprzez jego zaznaczenie i naciśnięcie **Delete** na klawiaturze. Zaniechanie rysowania to klawisz **Esc**. W przypadku zakończenia rysowania połączenia na innym połączeniu automatycznie wstawiany jest węzeł.

Tworzymy połączenia pomiędzy elementami (rys.5). Zapisujemy projekt **Ctrl+S** i ostatecznie uzyskujemy schemat układu gotowy do symulacji.



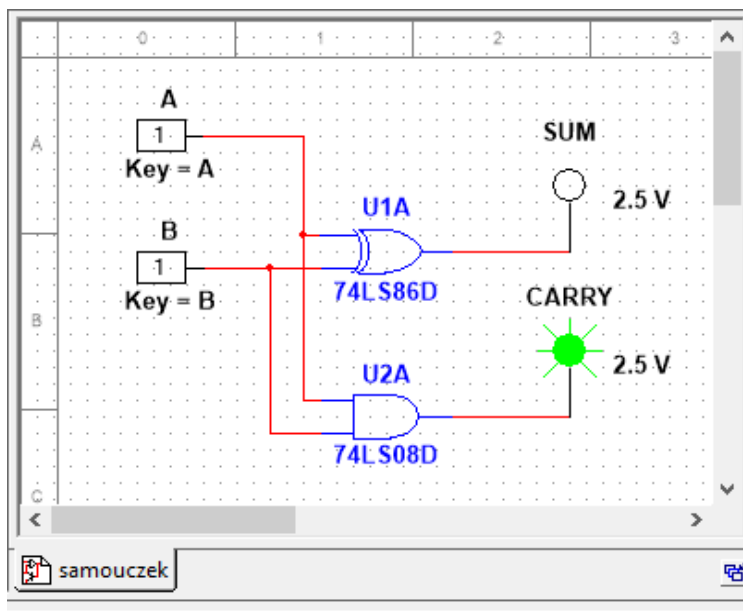
Rys. 5 Kompletny schemat półsumatora jednobitowego

Wymagane jest ustawienie parametrów symulacji przed jej uruchomieniem. W tym celu z menu programu głównego wybieramy **Simulate**→**Analyses and simulation**. Otwiera się okienko **Analyses and Simulation** (rys. 6), w którym wybieramy **Interactive Simulation**. Następnie w polu **End time (TSTOP)** wprowadzamy wartość 1000, zaznaczamy **Maximum time step (TMAX)** i wpisujemy wartość 0.01. Naciskamy **Save**. Takie ustawienia skrócą czas obliczeń.



Rys. 6 Okienko Analyses and Simulation

Uruchamiamy symulację. Wciskamy klawisze **A** i **B**. Obserwujemy stany logiczne na wyjściach **SUM** i **CARRY**, które generowane są w odpowiedzi na stany wprowadzane na wejścia półsumatora jednobitowego (rys.7), zgodnie z tabelą prawdy opisującą zbudowany układ.



Rys. 7 Przykładowe stany logiczne na wyjściach półsumatora jednobitowego

WSKAZÓWKA #1

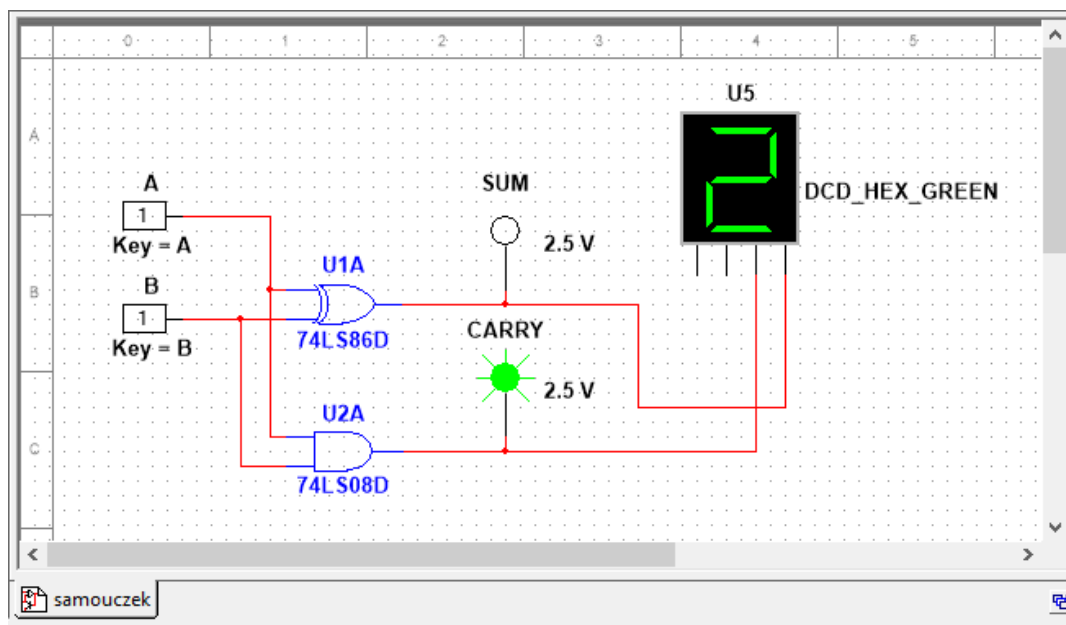
Po utworzeniu nowego projektu należy pamiętać o ustawieniu parametrów symulacji przed jej uruchomieniem.

WSKAZÓWKA #2

W realizowanych projektach można również zastosować wyświetlacze 7-segmentowe. W tym celu otwieramy okienko **Select a Component** (Ctrl+W). Następnie w polu **Group**: z rozwijanego menu wybieramy **Indicators**, a w polu **Family**: zaznaczamy **HEX_DISPLAY**. Wybieramy jeden z dostępnych komponentów:

- **DCD_HEX** – wyświetlacz w kolorze czerwonym,
- **DCD_HEX_BLUE** – wyświetlacz w kolorze niebieskim,
- **DCD_HEX_GREEN** – wyświetlacz w kolorze zielonym,
- **DCD_HEX_ORANGE** – wyświetlacz w kolorze pomarańczowym,
- **DCD_HEX_YELLOW** – wyświetlacz w kolorze żółtym.

Każdy z komponentów zawiera cztery wyprowadzenia, którym przypisano odpowiednie wagi pozycji naturalnego kodu binarnego (NKB) w porządku malejącym od lewej do prawej strony. Stąd lewemu skrajnemu wyprowadzeniu odpowiada najbardziej znaczący bit (MSB), a prawemu skrajnemu wyprowadzeniu przypisuje się bit najmniej znaczący (LSB). Na rysunku 8 przedstawiony został sposób podłączenia wyświetlacza do wyjść półsumatora jednobitowego.



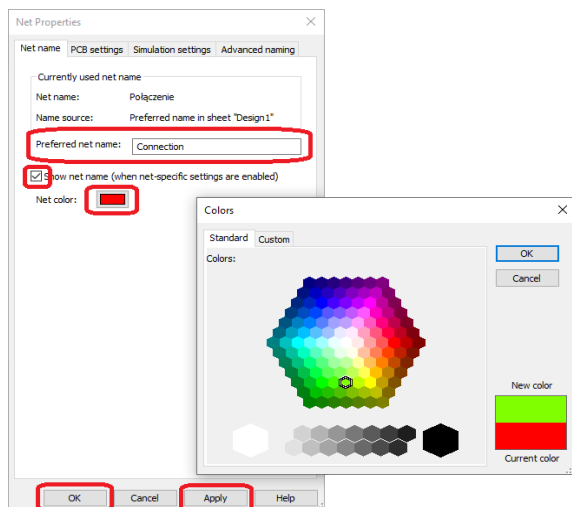
Rys. 8 Schemat półsumatora jednobitowego z wyświetlaczem 7-segmentowym

WSKAZÓWKA #3

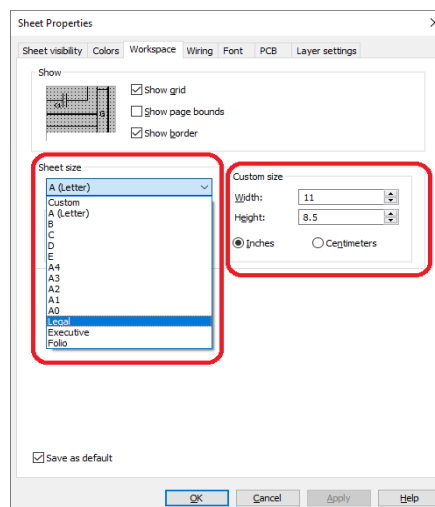
Na schemacie narysowanym połączeniom można nadawać nazwy lub zmieniać ich kolory. Należy wskazać kursorem połączenie i kliknąć dwukrotnie lewym klawiszem myszki. Otwiera się okienko **Net Properties** (rys. 9), w którym w zakładce **Net name**: wprowadzamy nazwę połączenia w polu **Preferred net name** i zaznaczamy **Show net name**. Zmiana koloru odbywa się poprzez naciśnięcie ikonki obok etykiety **Net color** i przybornik koloru, który jest wywołany. Naciskamy **Apply**, aby zatwierdzić zmiany, a następnie zamykamy okienko właściwości poprzez **OK**.

WSKAZÓWKA #4

W sytuacji za małej powierzchni obszaru edytora schematu można zmienić jej wielkość. Należy wskazać kursorem okno schematu i nacisnąć prawy klawisz myszki. Rozwija się menu, w którym wybieramy **Properties** (Ctrl+M). Następnie otwiera się okienko **Sheet Properties** (rys.10). W zakładce **Workspace** rozwijamy menu **Sheet size** i wybieramy odpowiednią wielkość lub ustawiamy wymiary w polu **Custom size** (w calach lub centymetrach).



Rys. 9 Okienko Net properties i przybornik kolorów



Rys. 10 Okienko Sheet Properties

PRZYKŁADOWE ZADANIA PROJEKTOWE

1. Zbudować układ sumatora pełnego jednobitowego oraz sprawdzić jego działanie
2. Zbudować i sprawdzić działanie komparatora cyfrowego, dwubitowego o wejściach A(1:0), B(1:0) oraz wyjściach: A=B, A>B i A<B.
3. Sprawdzić działanie sumatora czterobitowego 7483.
4. Zbudować układ sumatora/subtraktora z użyciem układu 7483. Wykazać prawidłowość jego działania.
5. Zbudować sumator dwóch liczb w kodzie BCD.
6. Zbudować i sprawdzić działanie układu wielobitowego sumatora szeregowego.
7. Zbudować n-bitowy generator parzystości.

IV. ZAGADNIENIA DO OPRACOWANIA PRZED PRZYSTĄPIENIEM DO ĆWICZENIA

1. Operacje arytmetyczne z użyciem kodów dwójkowych (ZM, ZU1, ZU2).
2. Przeznaczenie, budowa oraz równania logiczne: sumatora, półsumatora, subtraktora.
3. Budowa i zasada działania komparatora cyfrowego.
4. Zalety i wady sumatorów: równoległych i szeregowych.
5. Sumatory dziesiętne (BCD) – budowa i przeznaczenie.
6. Zasady łączenia elementów sumatorów (binarnych oraz BCD) w celu uzyskania sumatorów n-bitowych.
7. Sposoby realizacji operacji mnożenia.
8. Zasada działania i budowa układów mnożących.

V. LITERATURA I MATERIAŁY POMOCNICZE

1. Materiały wykładowe.
2. NI Multisim. User Manual, National Instruments Corporation 2009.
3. J. Kalisz: Podstawy elektroniki cyfrowej, WKŁ, Warszawa 2007.
4. J. Tyszer: Układy cyfrowe, Wyd. Politechniki Poznańskiej, 2000.
5. J.F. Wakerly: Digital Design, Principles and Practices, 4th Edition, Pearson/Prentice Hall, 2005.
6. B. Wilkinson: Układy cyfrowe, WKŁ, 2000.
7. A. Skorupski: Podstawy techniki cyfrowej, WKŁ, 2001.
8. J. Tyszer, G. Mrugalski, A. Pogiel, D. Czysty: Technika cyfrowa, Zbiór zadań z rozwiązaniami, BTC, Wyd. 2, Legionowo 2016.