

SPRAWOZDANIE
Z
LABORATORIUM UKŁADÓW LOGICZNYCH

Temat ćwiczenia nr 5: Liczniki synchroniczne i asynchroniczne.

Wykonał:

CEL ĆWICZENIA:

Celem ćwiczenia jest zaznajomienie się z projektowaniem liczników w wersji synchronicznej oraz problemami z tym związanymi.

PRZEBIEG ĆWICZENIA:

Mój licznik będzie posiadał do wyboru cztery długości liczenia: *modulo 3*, *modulo 5*, *modulo 9*, *modulo 12*.

Do utworzenia licznika o S stanach potrzeba k przerzutników ($2^{k-1} \leq S \leq 2^k$). W moim przypadku maksymalnym stanem jest 12, więc mój licznik będzie składał się z 4 przerzutników typu D.

Do kodowania stanów użyłem kodu dwójkowego.

Aby zmieniać pojemność liczenia, zbuduję specjalny układ dekodujący, którego zadaniem będzie zmiana długości liczenia w zależności od wartości sygnałów sterujących.. Wprowadzę dwa sygnały sterujące a i b , które będą „odpowiedzialne” za długość liczenia.

Tabela 1 przedstawia zmianę długości liczenia w zależności od wartości sygnałów sterujących:

sygnał a	sygnał b	dł. liczenia
0	0	modulo 3
0	1	modulo 5
1	0	modulo 9
1	1	modulo 12

Tabela 1

Na poszczególne wejścia dekodera oprócz sygnałów sterujących, będą także podawane sygnały z wyjść liczników ($Q_0 - Q_3$). Wyjście dekodera będzie połączone z asynchronicznym wejściem zerowania (RESET) każdego z przerzutników.

- Projektowanie układu dekodującego:

Cały układ dekodujący będzie składał się z czterech mniejszych układów (czterech bramek NAND). Każdy poszczególne układ (bramka NAND) będzie odpowiedzialny za odpowiednią długość liczenia. Ich wyjścia ($y_0 - y_3$) będą połączone bramką NAND, a wyjście tej bramki połączone do wejścia RESET każdego z liczników.

Normalnie każdy poszczególne układ powinien być bramką AND, a wszystkie wyjścia połączone bramką OR. Skorzystałem jednak z praw algebry Boole'a i zamiast bramek AND i OR zastosowałem bramki NAND. Ta metoda znacznie obniży koszt realizacji układu, a także zmniejszy ilość wykorzystanych układów TTL.

Pracę licznika ilustrują tablica Karnaugh, którymi się posłużyłem szukając odpowiednich kombinacji wyjść licznika i sygnałów sterujących. W miejscu (w stanie) gdzie jest 1, następuje wyzerowanie licznika. Znak '-' oznacza, że nie jest istotne, jaki stan logiczny pojawi się na wyjściu przy tej kombinacji. Ta kombinacja na pewno nie nastąpi, ponieważ licznik wcześniej zostanie wyzerowany.

a) modulo 3

$$a = 0$$

$$b = 0$$

W przypadku licznika modulo 3, licznik zacznie pracę od stanu 0, następnie przejdzie kolejno w stan 1 i 2, a w stanie 3 na wejście RESET zostanie podany wysoki sygnał ('1'), licznik zostanie wyzerowany i zacznie pracę od stanu 0.

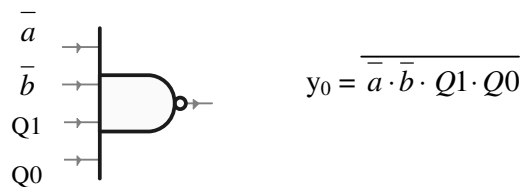
Analogicznie jak w przypadku pracy modulo 3, jest w pozostałych przypadkach.

Tablicę rozpatruję tylko dla przypadku, gdy $a = 0$ i $b = 0$

Q2Q1Q0	000	001	011	010	110	111	101	100
000	0	0	1	0	-	-	-	-
001	-	-	-	-	-	-	-	-

$$y_0 = \bar{a} \cdot \bar{b} \cdot Q1 \cdot Q0$$

Po podaniu powyższych sygnałów na wejścia bramki NAND, na wyjściu otrzymamy:



b) modulo 5

$$a = 0$$

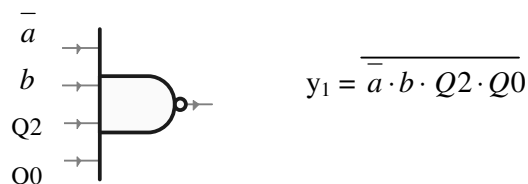
$$b = 1$$

Tablicę rozpatruję tylko dla przypadku, gdy $a = 0$ i $b = 1$

Q2Q1Q0	000	001	011	010	110	111	101	100
010	0	0	0	0	-	-	1	0
011	-	-	-	-	-	-	-	-

$$y_1 = \bar{a} \cdot b \cdot Q2 \cdot Q0$$

Po podaniu powyższych sygnałów na wejścia bramki NAND, na wyjściu otrzymamy:



c) modulo 9

a = 1

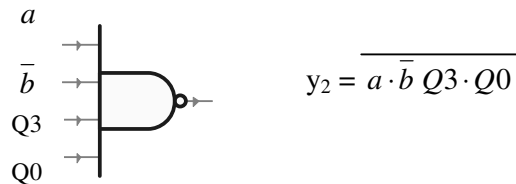
b = 0

Tablicę rozpatruję tylko dla przypadku, gdy a = 1 i b = 0

Q2Q1Q0 abQ3	000	001	011	010	110	111	101	100
100	0	0	0	0	0	0	0	0
101	0	1	-	-	-	-	-	-

$$y_2 = a \cdot \bar{b} \cdot Q_3 \cdot Q_0$$

Po podaniu powyższych sygnałów na wejścia bramki NAND, na wyjściu otrzymamy:



d) modulo 12

a = 1

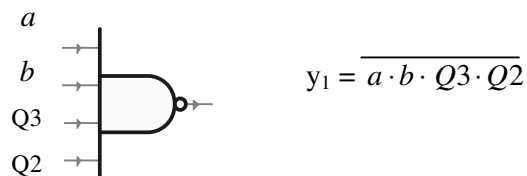
b = 1

Tablicę rozpatruję tylko dla przypadku, gdy a = 1 i b = 1

Q2Q1Q0 abQ3	000	001	011	010	110	111	101	100
110	0	0	0	0	0	0	0	0
111	0	0	0	0	-	-	1	-

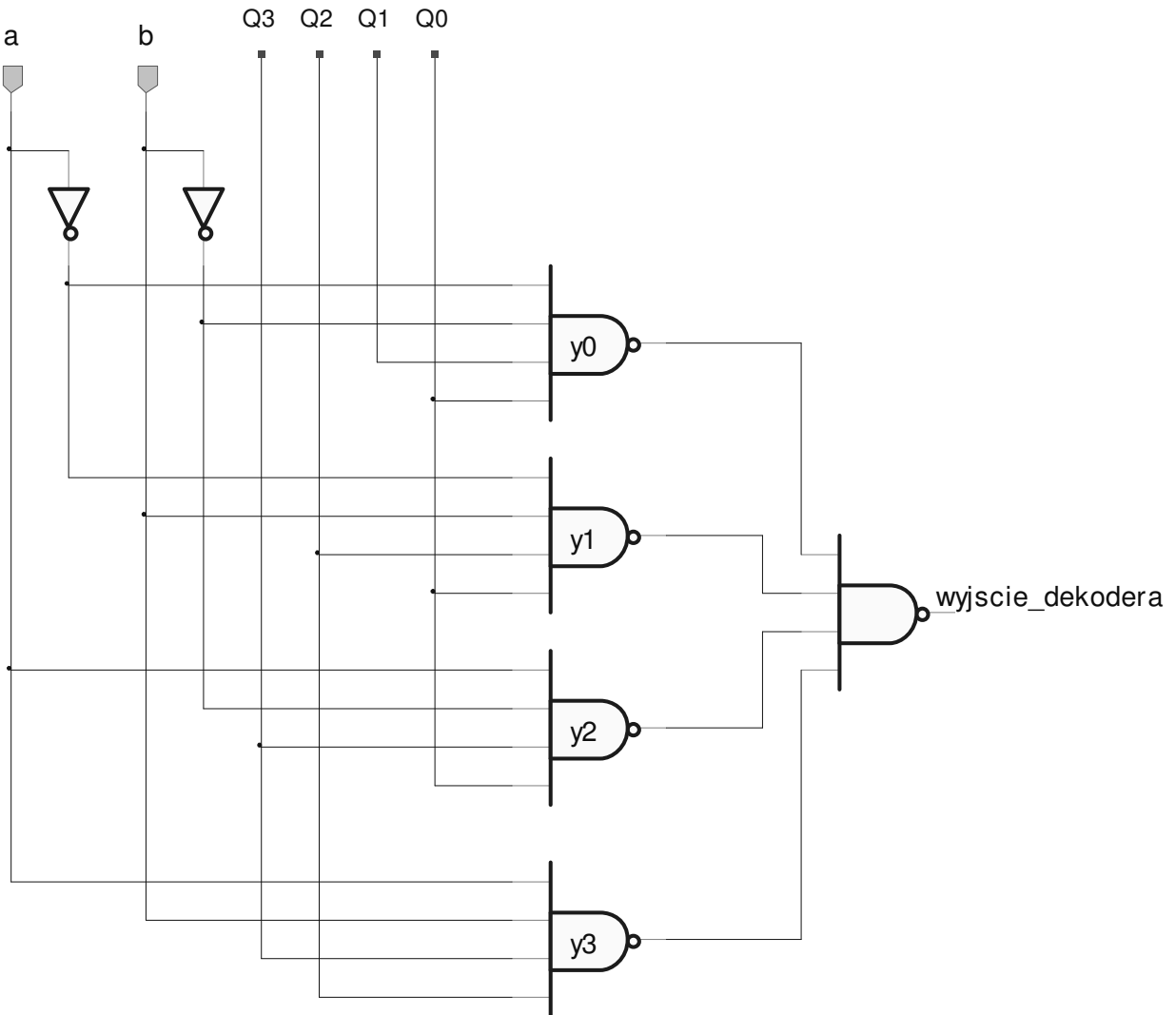
$$y_3 = a \cdot b \cdot Q_3 \cdot Q_2$$

Po podaniu powyższych sygnałów na wejścia bramki NAND, na wyjściu otrzymamy:



Poszczególne wyjścia $y_0 - y_3$ układów należy połączyć bramką NAND i wyjście tej bramki połączyć z wejściem RESET każdego z przerzutników.

Schemat połączeń dekodera w programie AHDL (Rysunek 1)



Rysunek 1

- Projektowanie licznika synchronicznego.

Tablica stanów (Tabela2)

Stan	Stan bieżący				Stan następny				Wejścia przerzutników			
	Q3	Q2	Q1	Q0	Q3'	Q2'	Q1'	Q0'	D3	D2	D1	D0
0	0	0	0	0	0	0	0	1	0	0	0	1
1	0	0	0	1	0	0	1	0	0	0	1	0
2	0	0	1	0	0	0	1	1	0	0	1	1
3	0	0	1	1	0	1	0	0	0	1	0	0
4	0	1	0	0	0	1	0	1	0	1	0	1
5	0	1	0	1	0	1	1	0	0	1	1	0
6	0	1	1	0	0	1	1	1	0	1	1	1
7	0	1	1	1	1	0	0	0	1	0	0	0
8	1	0	0	0	1	0	0	1	1	0	0	1
9	1	0	0	1	1	0	1	0	1	0	1	0
10	1	0	1	0	1	0	1	1	1	0	1	1
11	1	0	1	1	1	1	0	0	1	1	0	0
12	1	1	0	0	1	1	0	1	1	1	0	1
13	1	1	0	1	1	1	1	0	1	1	1	0
14	1	1	1	0	1	1	1	1	1	1	1	1
15	1	1	1	1	0	0	0	0	0	0	0	0

Tabela 2

Pracę licznika ilustruje tablica Karnaugh, którą się posłużyłem szukając odpowiednich kombinacji wyjść licznika podawanych na poszczególne przerzutniki.

a) przerzutnik D3

Q1Q0 Q3Q2	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	1	1	0	1
10	1	1	1	1

$$D3 = \overline{Q3} \cdot \overline{Q2} + \overline{Q3} \cdot \overline{Q1} + \overline{Q3} \cdot \overline{Q0} + \overline{Q3} \cdot Q2 \cdot Q1 \cdot Q0 = \overline{Q3} \cdot (\overline{Q2} + \overline{Q1} + \overline{Q0}) + \overline{Q3} \cdot Q2 \cdot Q1 \cdot Q0 = (\overline{Q2} \cdot \overline{Q1} \cdot \overline{Q0}) \cdot \overline{Q3} + Q2 \cdot Q1 \cdot Q0 \cdot \overline{Q3} = \overline{Q2} \cdot \overline{Q1} \cdot \overline{Q0} \oplus \overline{Q3}$$

b) przerzutnik D2

Q1Q0 Q3Q2	00	01	11	10
00	0	0	1	0
01	1	1	0	1
11	1	1	0	1
10	0	0	1	0

$$D2 = Q2 \cdot \overline{Q1} + Q2 \cdot \overline{Q0} + \overline{Q2} \cdot Q1 \cdot Q0 = Q2 \cdot (\overline{Q1} + \overline{Q0}) + \overline{Q2} \cdot (Q1 \cdot Q0) = Q2(\overline{Q1} \cdot \overline{Q0}) + \overline{Q2}(Q1 \cdot Q0) = Q2 \oplus Q1 \cdot Q0$$

c) przerzutnik D1

Q1Q0 Q3Q2	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

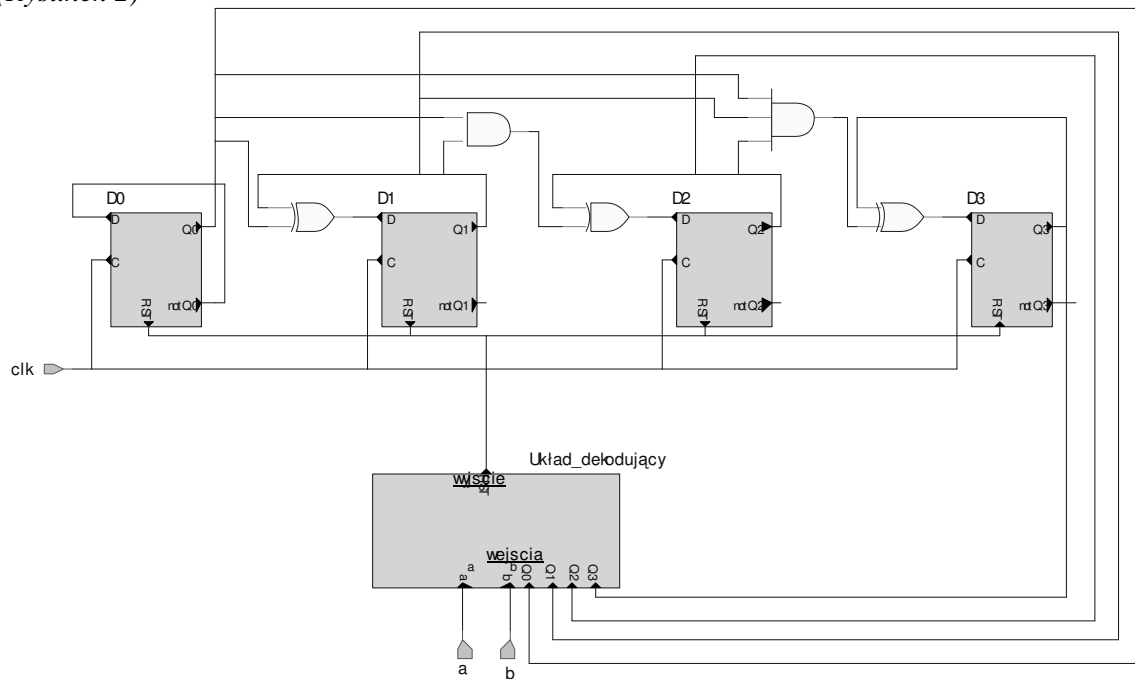
$$D1 = \overline{Q1} \cdot Q0 + Q1 \cdot \overline{Q0} = Q1 \oplus Q0$$

d) przerzutnik D0

Q1Q0 Q3Q2	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	1	0	0	1
10	1	0	0	1

$$D0 = \overline{Q0}$$

- Schemat licznika synchronicznego z przeniesieniem równoległym o programowalnej długości (Rysunek 2)

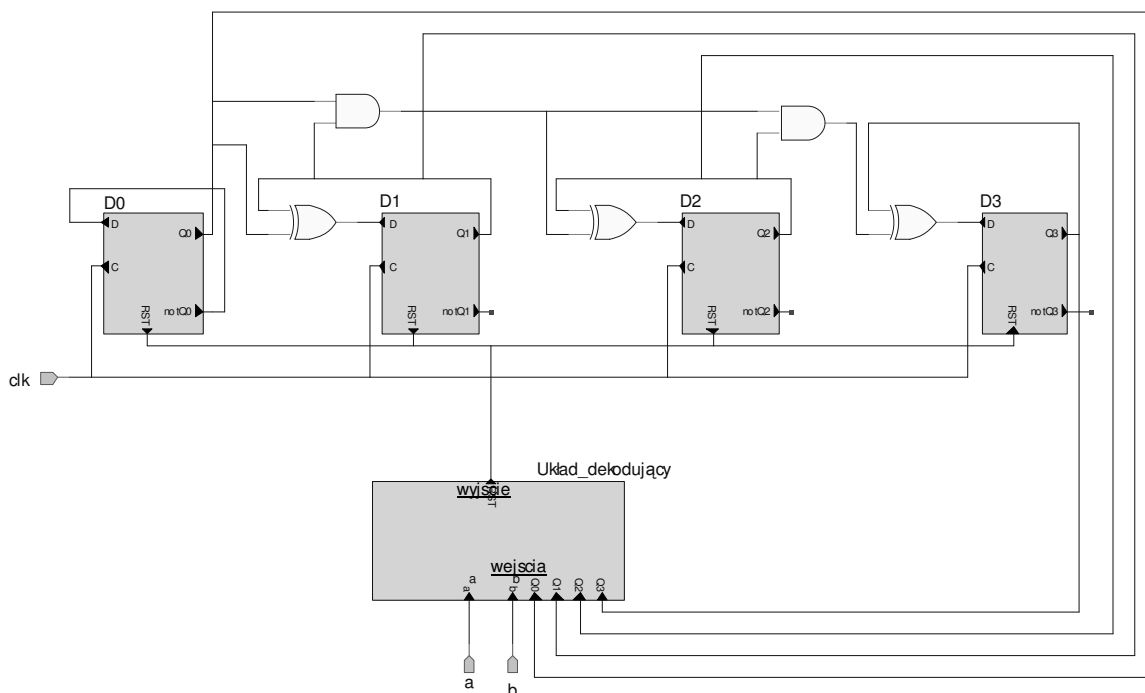


Rysunek 2

Maksymalna częstotliwość liczenia powyższego układu określona jest czasem trwania przerzutu jednego przerzutnika, powiększona o czas opóźnienia na bramce podłączonej do wejścia informacyjnego przerzutnika.

$$f_{\max} = 1/(t_{\text{przerzutnika}} + t_{\text{bramki AND}}) = \frac{1}{40\text{ns} + 15\text{ns}} = 18,2 \text{ MHz}$$

- Schemat licznika synchronicznego z przeniesieniem szeregowym o programowalnej długości (Rysunek 3)



Rysunek 3

$$f_{\max} = 1/(t_{\text{przerzutnika}} + 2t_{\text{bramki AND}}) = \frac{1}{40\text{ns} + 2 \cdot 15\text{ns}} = 14,3 \text{ MHz}$$

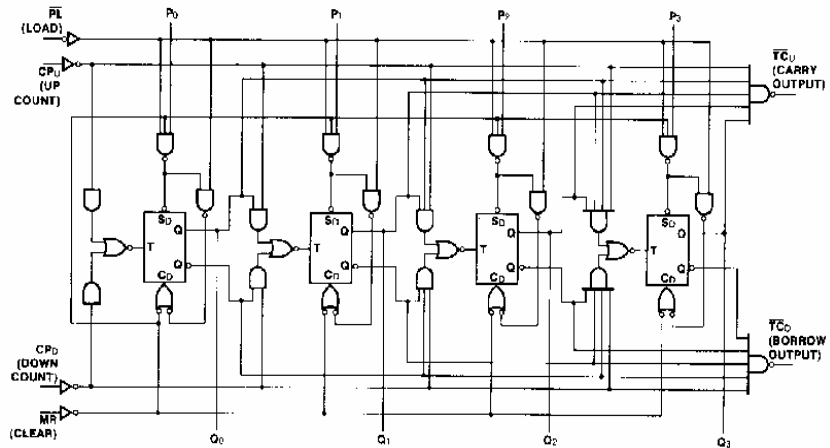
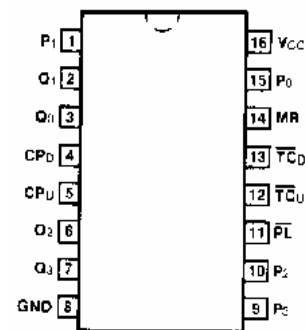
Z powyższych obliczeń wynika, że licznik z przeniesieniem szeregowym działa wolniej od licznika z przeniesieniem równoległym. Wynika to z zamiany iloczynów wielowejsiowych stosowanych w przeniesieniu równoległym na połączone kaskadowe iloczyny dwuwejsiowe.

Projekt układu w programie AHDL z wykorzystaniem bibliotek TTL

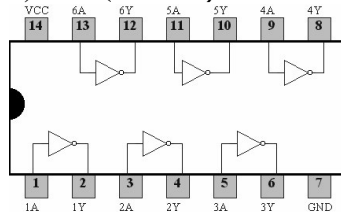
Do budowy układu wykorzystam jeden układ 74193, jeden układ 7404 i 3 układy 7420. Układ 74193 pełnił rolę licznika synchronicznego, układ 7404 był wykorzystywany do zanegowania sygnałów sterujących a i b. Układ 7420 potrzebny był do sterowania długości liczenia tzn. każda bramka NAND była poszczególnym dekoderym. U układach 7420 (U3 i U4) na każde wejście bramki NAND podawane były sygnały sterujące a, b i odpowiednie wyjścia licznika. Układ 7420 (U5), w którym wykorzystałem tylko jedną bramkę NAND, sumował sygnały wyjściowe każdego z dekoderym (*modulo 3, modulo 5, modulo 9, modulo 12*), a wyjście tej bramki podłączone było do wejścia RESET układu 74193.

Przekroje układów TTL:

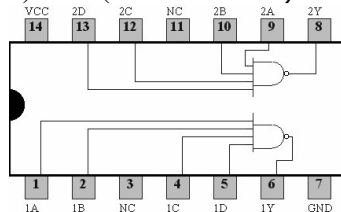
a) 74193



b) 7404 (6x NOT)



c) 7420 (2x NAND 4we)



Do budowy tego układu wykorzystam także trzy magistrale: Q , $ster$, wy .

Do magistrali Q podłączone będą wyjścia (Q_0 - Q_3) licznika, gdzie:

- $Q(0)$ oznacza Q_0 ,
- $Q(1)$ oznacza Q_1 ,
- $Q(2)$ oznacza Q_2 ,
- $Q(3)$ oznacza Q_3 ,
- $Q(4)$ oznacza Q_4 ,

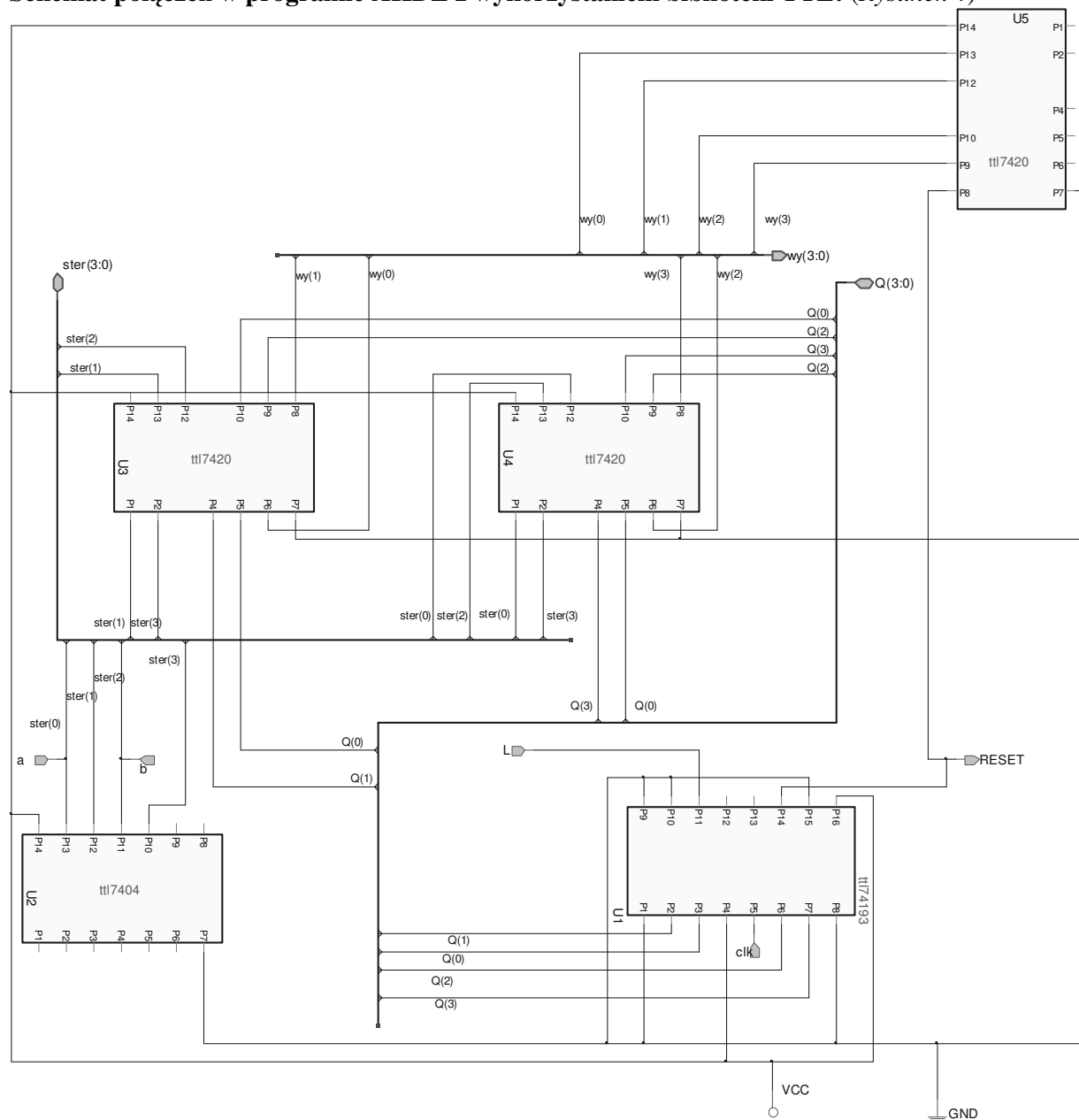
Do magistrali *ster* podłączone będą sygnały sterujące *a* i *b* oraz ich negacje, gdzie:

- *ster(0)* oznacza *a*,
- *ster(1)* oznacza \bar{a} ,
- *ster(2)* oznacza *b*,
- *ster(3)* oznacza \bar{b} ,

Do magistrali *wy* podłączone będą wyjścia każdego z dekodерów (*modulo 3*, *modulo 5*, *modulo 9*, *modulo 12*), gdzie:

- *wy(0)* oznacza wyjście dekodera *modulo 3*,
- *wy(1)* oznacza wyjście dekodera *modulo 5*,
- *wy(2)* oznacza wyjście dekodera *modulo 9*,
- *wy(3)* oznacza wyjście dekodera *modulo 12*,

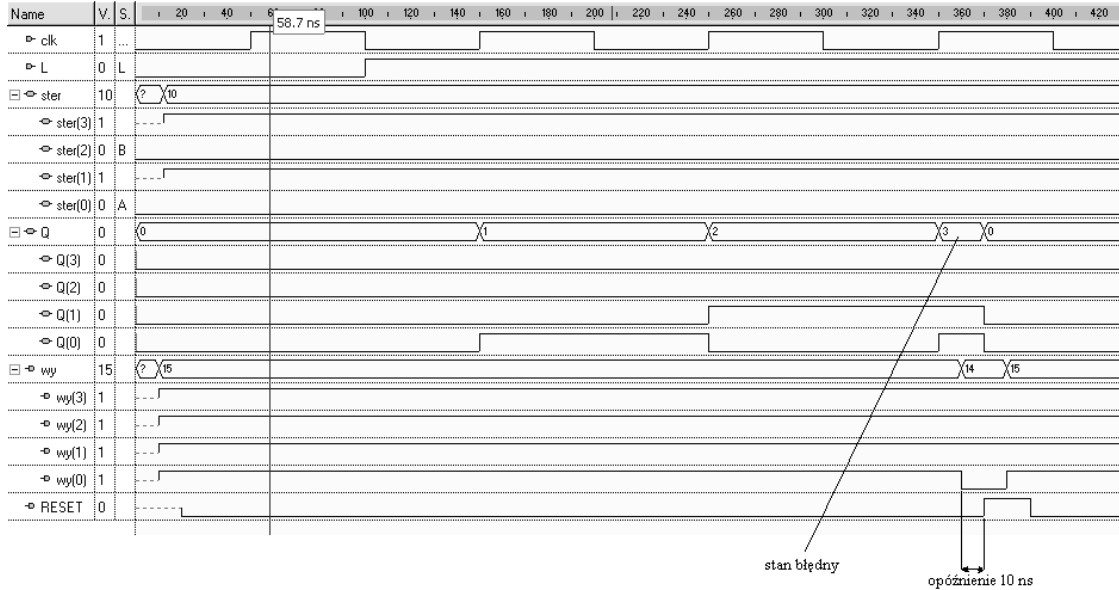
Schemat połączeń w programie AHDL z wykorzystaniem biblioteki TTL: (Rysunek 4)



Rysunek 4

Przebiegi czasowe:

a) modulo 3 (Rysunek 5)



Rysunek 5

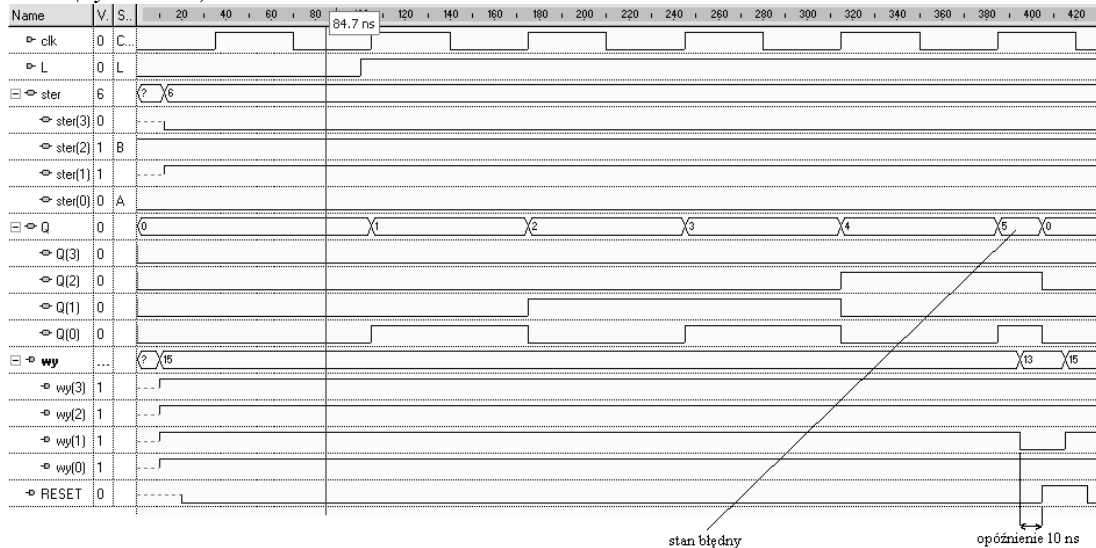
Z powyższego przebiegu widać, że układ reaguje na zbocze opadające sygnału zegarowego.

Na wejścia sterujące *a* i *b* podałem sygnał niski ('0'). Widać, że sygnały *ster(0)* i *ster(2)* mają stan niski, a sygnały *ster(1)* i *ster(3)*, które są negacją sygnałów *a* i *b* mają stan wysoki ('1'). W takim przypadku układ miał liczyć modulo 3 i widać, że właśnie tak pracuje licznik. W stanie 3 na *wy(0)* pojawił się stan niski, który powodował wystąpienie na wejściu RESET stanu wysokiego ('1'), co powodowało wyzerowanie licznika.

Na przebiegu czasowym możemy zaobserwować działanie układu. Widać, że na wyjściach licznika jest tylko jeden stan błędny (3), który trwa 10 ns. Jest on związany z asynchroniczną budową wejścia resetującego licznika.

Stan błędny analogicznie pojawiał się na pozostałych przebiegach.

b) modulo 5 (rysunek 6)



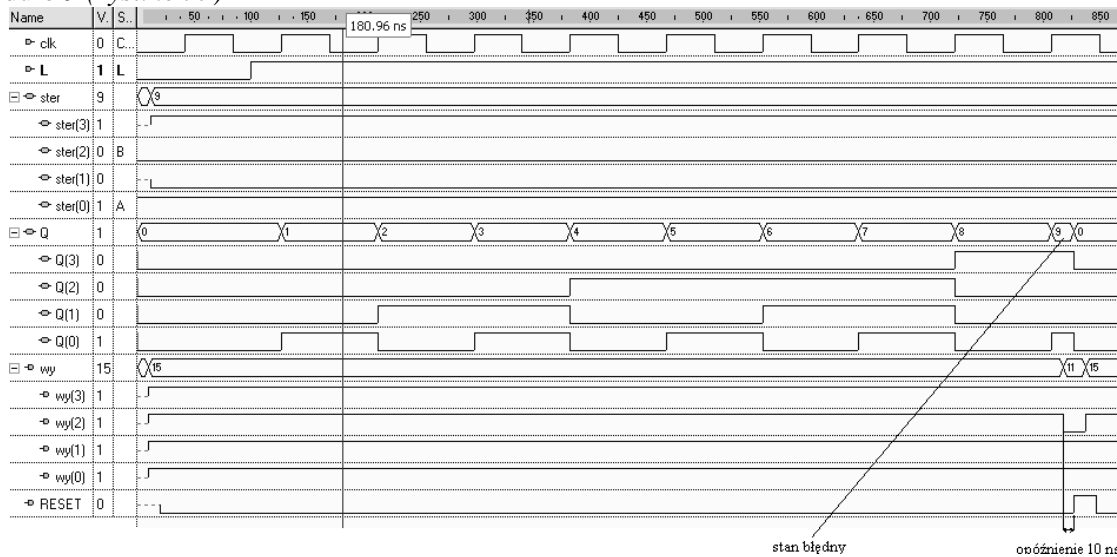
Rysunek 6

Na wejścia sterujące *a* podałem sygnał niski ('0'), a na wejście *b* sygnał wysoki ('1'). Widać, że sygnał *ster(0)* ma stan niski, a sygnał *ster(2)* ma stan wysoki. Sygnały *ster(1)* i *ster(3)*, które są negacją sygnałów *a* i *b* mają odpowiednio stan wysoki ('1') i niski ('0').

W takim przypadku układ miał liczyć *modulo 5* i widać, że właśnie tak pracuje licznik. W stanie 5 na *wy(1)* pojawił się stan niski, który powodował wystąpienie na wejściu RESET stanu wysokiego ('1'), co powodowało wyzerowanie licznika.

Stany błędny zostały przedstawiony, a opóźnienie występuje identycznie jak w przypadku modulo 3.

c) modulo 9 (rysunek 7)

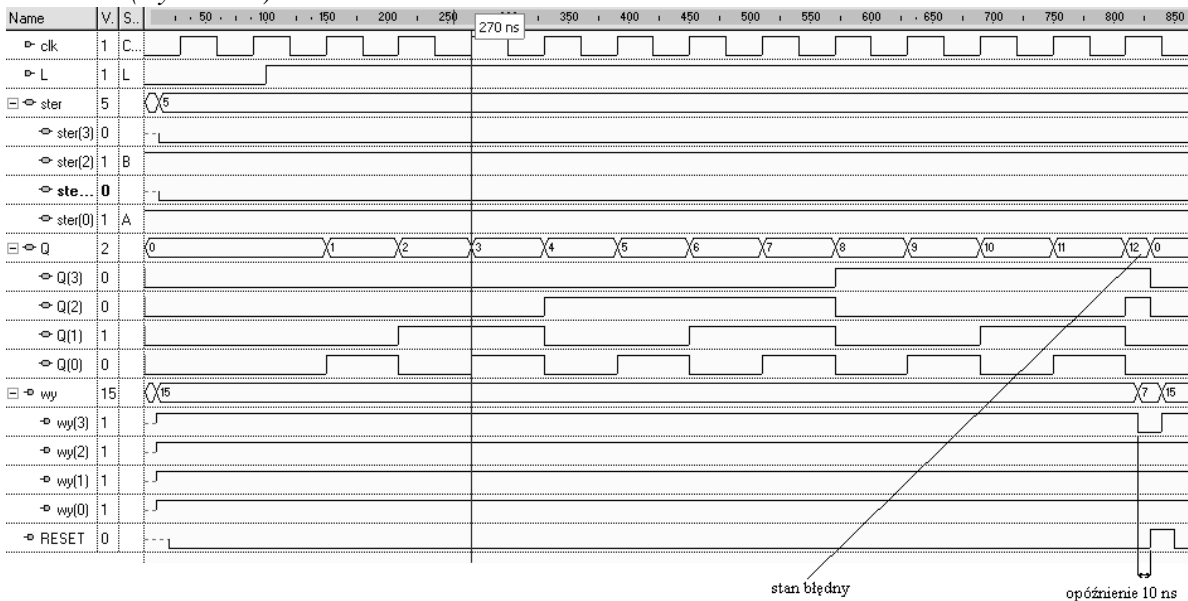


Rysunek 7

Na wejścia sterujące *a* podałem sygnał wysoki ('1'), a na wejście *b* sygnał niski ('0'). Widać, że sygnał *ster(0)* ma stan wysoki, a sygnał *ster(2)* ma stan niski. Sygnały *ster(1)* i *ster(3)*, które są negacją sygnałów *a* i *b* mają odpowiednio stan niski ('0') i wysoki ('1'). W takim przypadku układ miał liczyć *modulo 9* i widać, że właśnie tak pracuje licznik. W stanie 9 na *wy(2)* pojawił się stan niski, który powodował wystąpienie na wejściu RESET stanu wysokiego ('1'), co powodowało wyzerowanie licznika.

Stany błędny zostały przedstawiony, a opóźnienie występuje identycznie jak w przypadku modulo 3.

d) modulo 12 (Rysunek 8)



Rysunek 8

Na wejścia sterujące *a* i *b* podałem sygnał wysoki ('1'). Widać, że sygnały *ster(0)* i *ster(2)* mają stan wysoki, a sygnały *ster(1)* i *ster(3)*, które są negacją sygnałów *a* i *b* mają stan niski ('0'). W takim przypadku układ miał liczyć *modulo 12* i widać, że właśnie tak pracuje licznik.

W stanie 12 na wy(3) pojawił się stan niski, który powodował wystąpienie na wejściu RESET stanu wysokiego ('1'), co powodowało wyzerowanie licznika.

Stany błędny zostały przedstawiony, a opóźnienie występuje identycznie jak w przypadku modułu 3.

Parametry katalogowe układów:

74193: $f_{\max} = 32 \text{ MHz}$

$t_{p1} = 47 \text{ ns}$ (czas od podania sygnału wejściowego do pojawienia się sygnału na wyjściu)

$t_{p2} = 40 \text{ ns}$ (czas od podania sygnału na wejście zerujące lub wpisujące do pojawienia się sygnału na wyjściu)

7404: $t_p = 15 \text{ ns}$

7420: $t_p = 5 \text{ ns}$

Koszt realizacji.

- (układ **74193**) – 16 zł
- (układ **7404**) – 2,36 zł
- (układ **7420**) – 2,36 zł

$$\text{Koszt} = 16 + 2,36 + 3 \cdot 2,36 = 25,44 \text{ zł}$$

Maksymalna częstotliwość powyższego układu wynosi:

$$f_{\max} = 1/(t_{p1\ 74193} + t_{p2\ 74193} + 2 \cdot t_{p7420} + t_{p7404} + 10 \text{ ns}) = \frac{1}{47\text{ns} + 40\text{ns} + 2 \cdot 5\text{ns} + 15\text{ns} + 10\text{ns}} = \frac{1}{122\text{ns}} = 8,2 \text{ MHz}$$

Wartość 10 ns jest naszym zabezpieczeniem, które stosuje się, by zapobiec błędnym stanom.

Kosz realizacji analogicznego układu licznika asynchronicznego z poprzedniego ćwiczenia wyniosła 16,21 zł.

Maksymalna częstotliwość układu wynosiła:

$$f_{\max} = 1/(t_{p1\ 7493} + t_{p2\ 7493} + 2 \cdot t_{p7420} + t_{p7404} + 10 \text{ ns}) = \frac{1}{135\text{ns} + 40\text{ns} + 2 \cdot 5\text{ns} + 15\text{ns} + 10\text{ns}} = \frac{1}{210\text{ns}} = 4,8 \text{ MHz}$$

Wartość 10 ns jest naszym zabezpieczeniem, które stosuje się, by zapobiec błędnym stanom.

Wnioski:

Liczniki są grupą układów logicznych (sekwencyjnych) służących do liczenia impulsów i pamiętania ich liczby.

W licznikach synchronicznych impulsy wejściowe podawane są wszystkie przerzutniki jednocześnie, przez co informacja na wyjściach przerzutników pojawia się w tej samej chwili czasowej.

W ćwiczeniu tym korzystaliśmy ze scalonego rewersyjnego licznika synchronicznego modułu 74193. Pracuje on w następujących trybach:

- równoległego wpisywania stanu (informacji) startowego (wejścia równoległe danych A, B, C, D);
- równoległego wyprowadzania informacji (Q3, Q2, Q1, Q0);
- dwukierunkowego zliczania impulsów (wejścia T_+ - dla dodawania impulsów i T_- - dla odejmowania impulsów);
- łączenia w zespoły liczników o większej pojemności (przez wyjścia przeniesienia P_+ - do zliczania w przód i pożyczki P_- - do zliczania wstecz).

Po realizacji tego ćwiczenia możemy porównać różnice pomiędzy licznikami synchronicznymi i asynchronicznymi. Widać, że układ z wykorzystaniem licznika synchronicznego może pracować z dwukrotnie większą częstotliwością niż analogiczny układ z wykorzystaniem licznika asynchronicznego. Ponadto w liczniku synchronicznym zaobserwowaliśmy tylko jeden stan błędny, gdzie w układzie z licznikiem asynchronicznym tych stanów było kilka. Związane jest to z tym, że w licznikach asynchronicznych impulsy wejściowe są podawane na jeden przerzutnik, w związku z tym informacja na wyjściach przerzutników pojawia się w różnych chwilach czasowych, co prowadzi do powstawania stanów błędnych. Jediną przewagą licznika asynchronicznego jest jego niższa cena.

Licznik synchroniczny z przeniesieniem równoległym może pracować z większą częstotliwością niż licznik z przeniesieniem szeregowym, przez co jest szybszy. Konstrukcja licznika z przeniesieniem szeregowym umożliwia budowę licznika o dowolnej dużej pojemności, ponieważ wszystkie człony licznika (oprócz pierwszego) są jednakowe.