

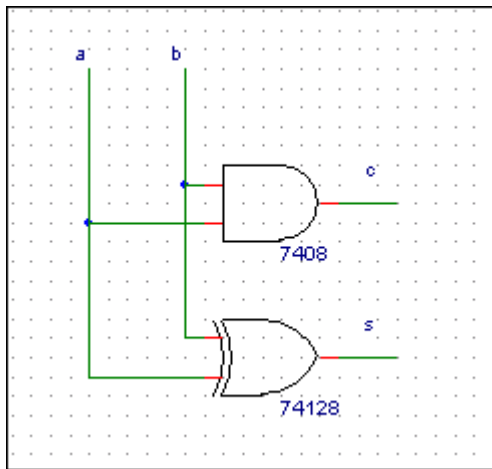
6.1 Układy arytmetyczne

Układy arytmetyczne to układy (kombinacyjne lub sekwencyjne) realizujące dodawanie odejmowanie i mnożenie i dzielenie. Zaczniemy od układów sumatorów.

1. Sumatory

Zaczniemy omawianie sumatorów od *półsumatora*, układu, który dodaje 2 bity bez uwzględnienia przeniesienia z poprzedniej pozycji.

Układ dodający 2 bity powinien mieć 2 wyjścia: wyjście sumy s i wyjście przeniesienia c . Łatwo zauważyć, że układ półsumatora opisują 2 funkcje boolowskie $s = a \oplus b$ oraz $c = a \cdot b$, gdzie s jest sumą a c przeniesieniem. Układ realizujący półsumator pokazany jest na rys. 20.

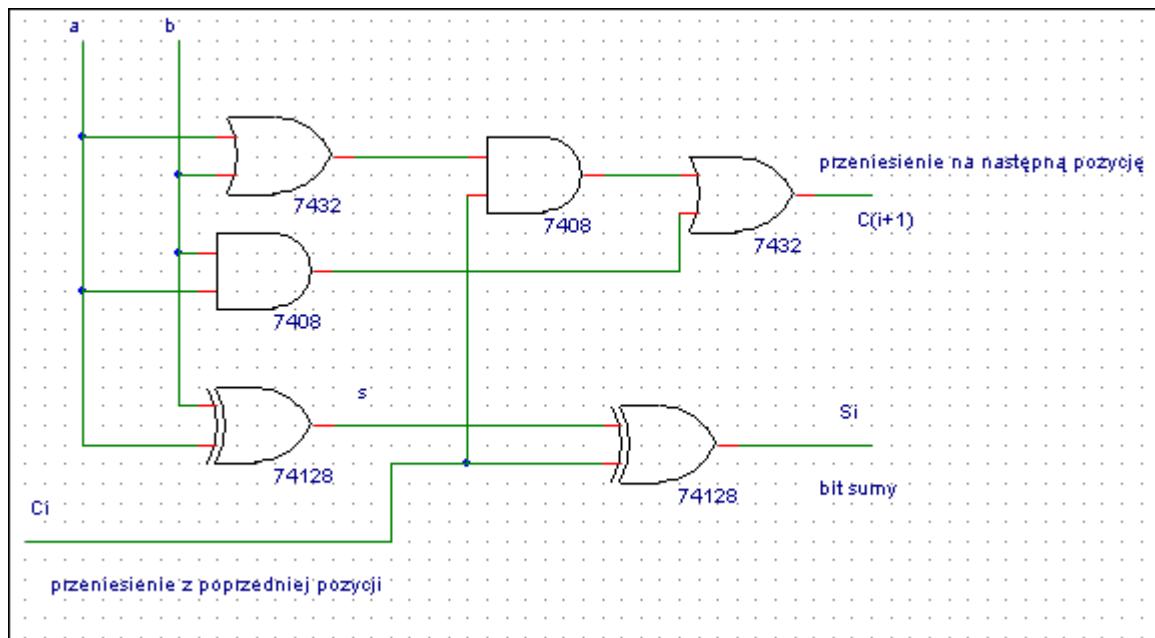


Rys 1. Układ realizujący półsumator $s = a \oplus b$ (bit sumy), $c = a \cdot b$ (bit przeniesienia)

Sumator jednobitowy pełny to układ kombinacyjny dodający 2 bity z uwzględnieniem przeniesień. Sumator jednobitowy pełny ma 3 wejścia (a_i , b_i - bity sumowane oraz c_i przeniesienie z poprzedniej pozycji) oraz 2 wyjścia: wyjście sumy s_i i wyjście przeniesienia na następną pozycję c_{i+1} . Łatwo zauważyć, że układ sumatora opisują 2 funkcje boolowskie

$$s_i = a_i \oplus b_i \oplus c_i \quad \text{oraz} \quad c_{i+1} = a_i \cdot b_i + (a_i + b_i) \cdot c_i,$$

gdzie s_i jest sumą, c_{i+1} przeniesieniem na pozycję $i+1$ a c_i przeniesieniem na pozycję i . Układ realizujący sumator jednobitowy pełny pokazany jest na rys 2. Jeśli czas opóźnienia bramki oznaczmy przez Δt to czas po którym uzyskujemy w tym układzie poprawny bit sumy to $2\Delta t$. Poprawny bit przeniesienia uzyskujemy po czasie $3\Delta t$.

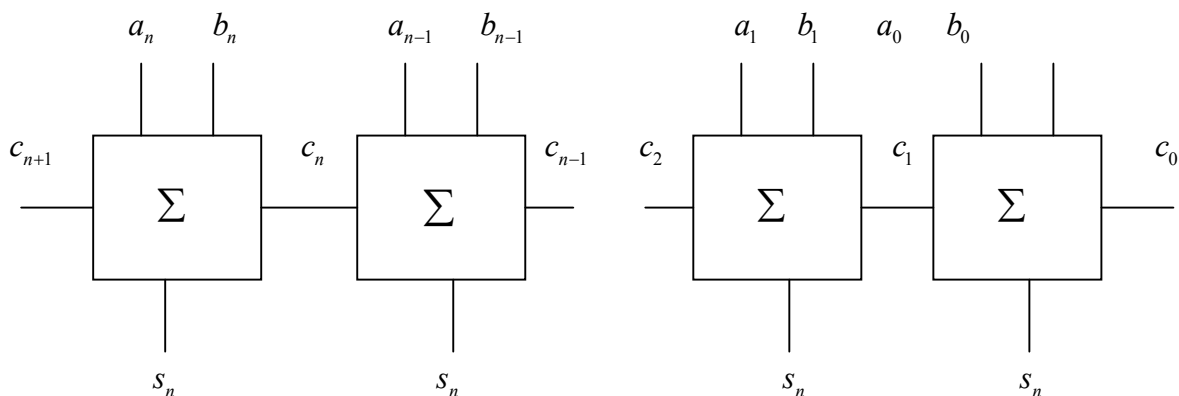


Rys. 2. Układ realizujący jednobitowy sumator pełny

Sumator równoległy jest układem kombinacyjnym o strukturze pokazanej na rys. 3. Dodajemy 2 liczby a i b w zapisie NKB (o stałej długości $n+1$ słowa kodowego). Liczby dodawane a i b reprezentowane są słowami binarnymi

$$a = a_n a_{n-1} \dots a_1 a_0 \quad \text{ i } \quad b = b_n b_{n-1} \dots b_1 b_0$$

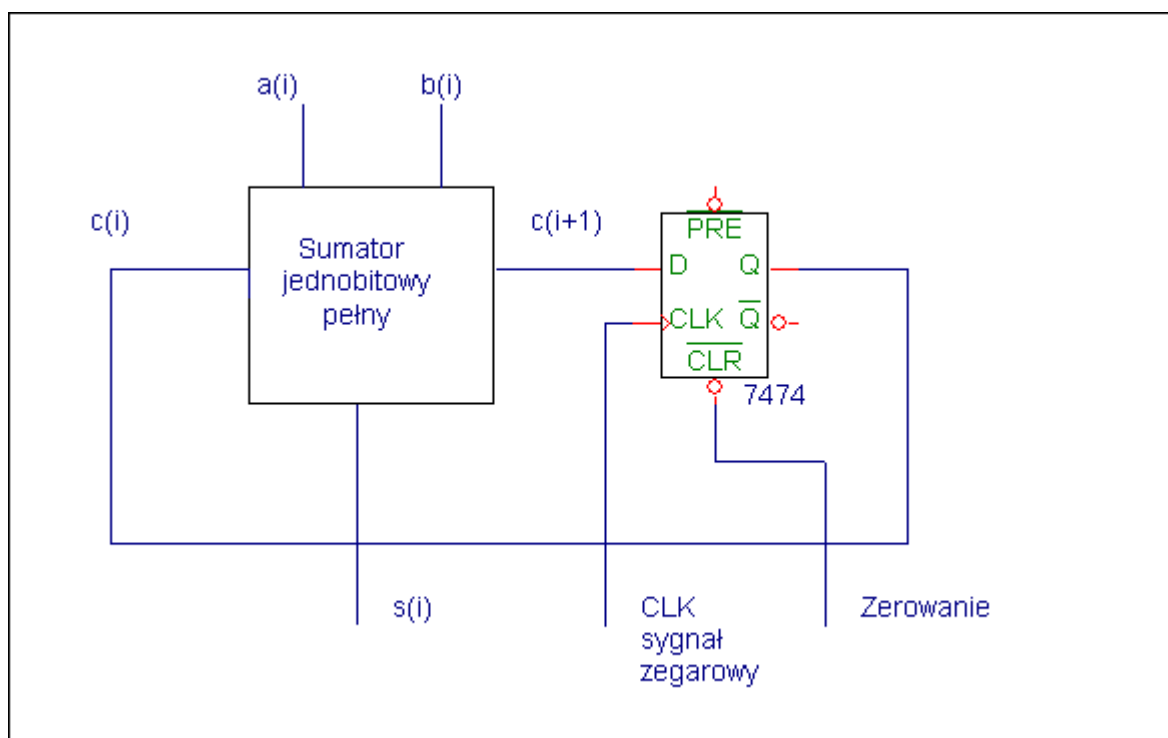
Suma $s = a + b$ reprezentowana jest słowem $s_n s_{n-1} \dots s_1 s_0$. Współpracujące ze sobą sumatory pełne realizują na kolejnych pozycjach słowa dodawanie $a_i + b_i$ (z uwzględnieniem przeniesień) a więc $s_i = a_i \oplus b_i \oplus c_i$ oraz $c_{i+1} = a_i \cdot b_i + (a_i + b_i) \cdot c_i$. Warto zauważyć, że słowo wyniku sumowania $s_n s_{n-1} \dots s_1 s_0$ (odczytywane w $n+1$ bitowym w kodzie NKB) jest równe liczbie $s = a + b$ wtedy i tylko wtedy gdy $c_{n+1} = 0$. Jeśli $c_{n+1} = 1$ to mówimy, że wystąpił nadmiar przy dodawaniu w kodzie NKB. W typowych mikroprocesorach pojawienie się $c_{n+1} = 1$ jest zapamiętywane, powoduje ustawienie przerzutnika znacznika przeniesienia CF (ang. carry flag) na 1. Oczywiście słowo $n+2$ bitowe $c_{n+1} s_n s_{n-1} \dots s_1 s_0$ zawsze reprezentuje poprawnie w kodzie NKB ($n+2$ bitowym) liczbę s .



Rys. 3. Sumator równoległy zbudowany z jednobitowych sumatorów pełnych

Ogólnie rzecz biorąc w systemach cyfrowych słowa binarne możemy przetwarzać równolegle (tak jest szybciej) lub szeregowo czyli bit po bicie (tak jest oszczędniej tzn. mniejsza ilość

bramek jest potrzebna do realizacji układu). *Sumator szeregowy* jest typowym układem z szeregowym przetwarzaniem informacji. Schemat układu pokazany jest na rys. 4. Układ dodaje szeregowo dwa słowa binarne $a = a_n a_{n-1} \dots a_1 a_0$ i $b = b_n b_{n-1} \dots b_1 b_0$ bit po bicie od strony najmniej znaczących bitów tzn. w pierwszym takcie zegara podajemy na wejścia sumatora bity a_0 i b_0 w drugim a_1 i b_1 itd. Dodawanie trwa więc $n + 1$ taktów zegara.



Rys. 4. Sumator szeregowy

Układ sumatora równoległego o strukturze pokazanej na Rys. 3. sumuje 2 liczby w czasie $(n+1)\Delta t$, gdzie Δt jest czasem opóźnienia bramki. Stosując *układ przewidywania przeniesień* (ang. look ahead) możemy znacznie przyspieszyć działanie układu sumatora równoległego. Żeby opisać działanie układu przewidywania przeniesień wprowadzamy dla każdej i tej pozycji 3 pomocnicze funkcje (określające stan pozycji) zdefiniowane następująco

$$P_i = a_i \oplus b_i \quad - P_i = 1 \text{ wtedy i tylko wtedy gdy pozycja } i \text{ jest w stanie propagacji}$$

$$G_i = a_i b_i \quad - G_i = 1 \text{ wtedy i tylko wtedy gdy pozycja } i \text{ jest w stanie generacji}$$

$$S_i = \bar{a}_i \cdot \bar{b}_i \quad - S_i = 1 \text{ wtedy i tylko wtedy gdy pozycja } i \text{ jest w stanie generacji 0}$$

Zauważmy, że

$$c_1 = G_0 + P_0 c_0$$

$$c_2 = G_1 + P_1 G_0 + P_1 P_0 c_0$$

$$c_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 c_0$$

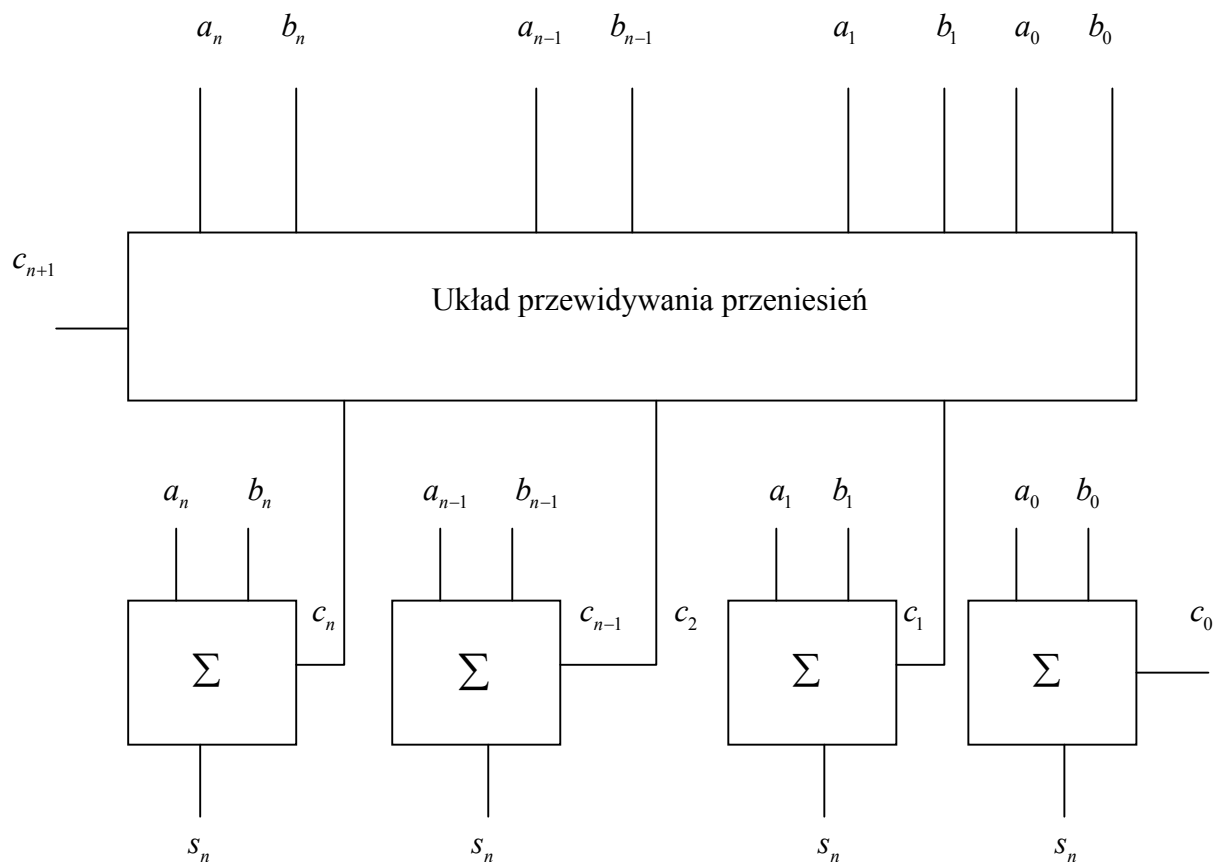
.....

$$c_i = G_{i-1} + P_{i-1} G_{i-2} + P_{i-1} P_{i-2} G_{i-3} + \dots + P_{i-1} P_{i-2} P_{i-3} \dots P_1 G_0 + P_{i-1} P_{i-2} P_{i-3} \dots P_0 c_0$$

.....

$$c_{n+1} = G_n + P_n G_{n-1} + P_n P_{n-1} G_{n-2} + \dots + P_n P_{n-1} P_{n-2} \dots P_1 G_0 + P_n P_{n-1} P_{n-2} \dots P_0 c_0$$

Zatem gdybyśmy dysponowali bramkami o odpowiedniej ilości wejść, można by obliczyć każde przeniesienie w czasie $3\Delta t$. Układ sumatora wykorzystujący układ przewidywania przeniesień jest więc potencjalnie znacznie szybszy od zwykłego układu sumatora równoległego. Układ sumatora z przewidywaniem przeniesień pokazany jest na Rys.5.



Rys. 5. Sumator równoległy z układem przewidywania przeniesień

2. Układy mnożące

Mnożenie wykonujemy jako wielokrotne dodawanie z przesunięciem. Oczywiście można zaprojektować do wykonywania mnożenia układ kombinacyjny złożony z samych sumatorów ale liczba użytych sumatorów jest wtedy równa $m \cdot n$, gdzie m i n są długościami mnożonych słów.