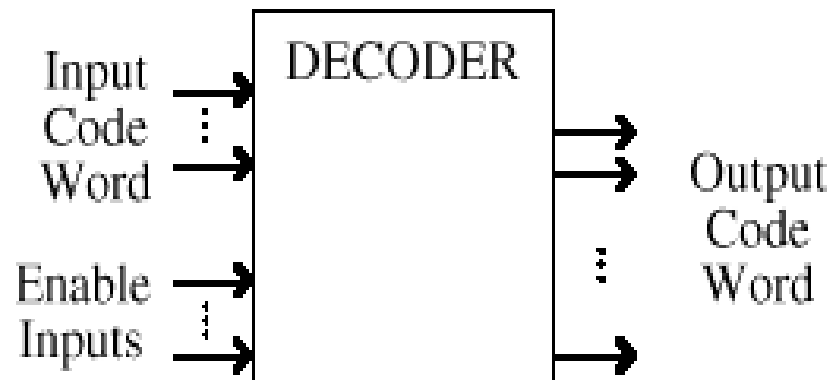


Dekoder.

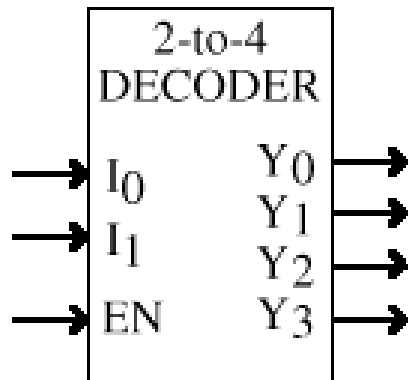
Dekoder jest elementem logicznym służącym do zmiany jednego kodu na inny. Dekodery oprócz sygnałów wejściowych i wyjściowych mogą posiadać jeszcze sygnały sterujące **Enable**.

Jeżeli którekolwiek z wejść sterujących jest nieaktywne, to wszystkie wyjścia przyjmują wartość niezależnie od stanu wejść.

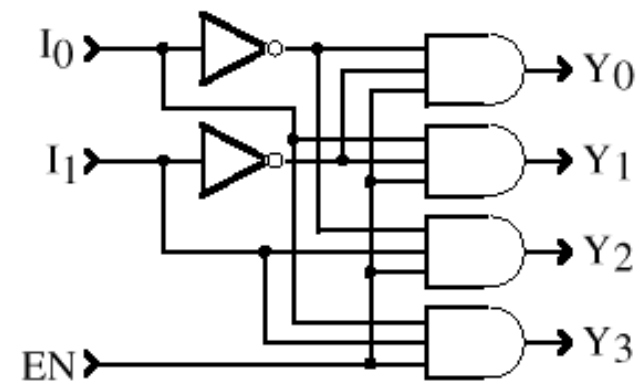


Demultiplekser

Multiplexer jest to dekodery który dla każdej binarnej wartości wejściowej uaktywnia jedną linię wyjściową. Dla m wejść posiada 2^m wyjść.



EN	I ₁	I ₀	Y ₃	Y ₂	Y ₁	Y ₀
0	x	x	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0



Ćwiczenie

Pokazać jak przy pomocy demultipleksera m wejściowego zrealizować funkcję logiczną m zmiennych

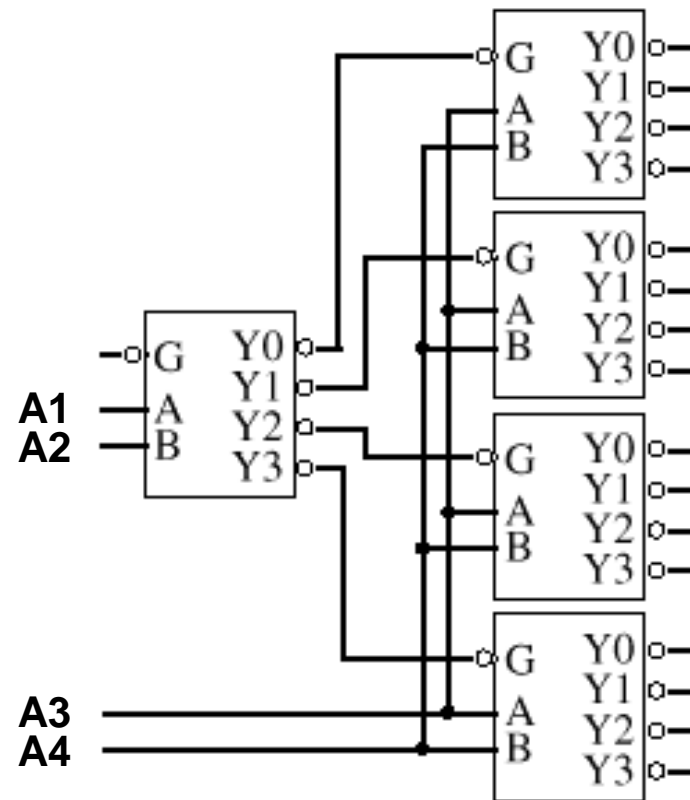
Kaskadowanie multiplekserów.

Możliwe jest kaskadowanie demultiplekserów.

Przykład

Przy pomocy 5 demultiplekserów 2 na 4 budujemy demultiplekser 4 na 16.

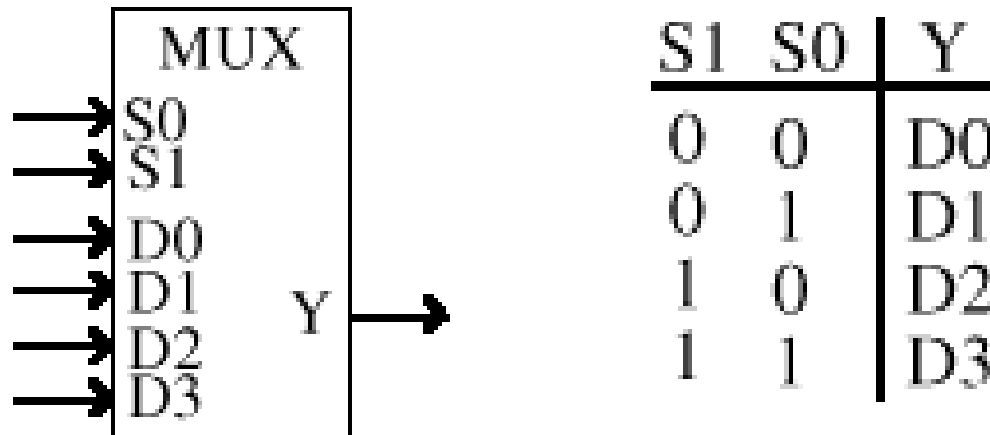
- linie sterujące pierwszego demultipleksera tworzą starsze bity numeru wybranego wyjścia
- linie sterujące pozostałych demultiplekserów są połączone ze sobą tworząc bity młodsze.



Multiplekser.

Multiplekserem nazywamy układ kombinacyjny o m wejściach adresowych 2^m wejściach informacyjnych i jednym wyjściu.

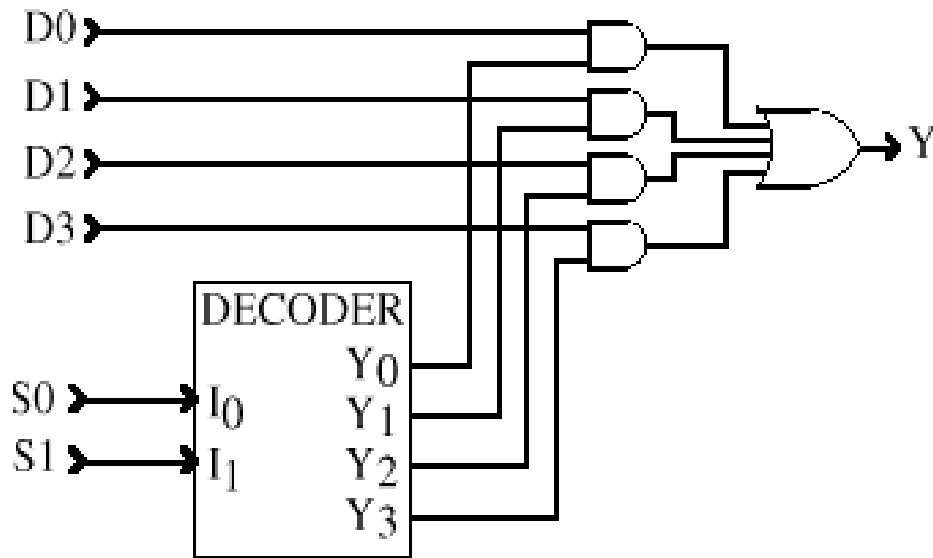
Sygnał pojawiający się na wyjściu jest równy sygnałowi na wejściu o numerze wybranym przez wejścia adresowe.



Ćwiczenie

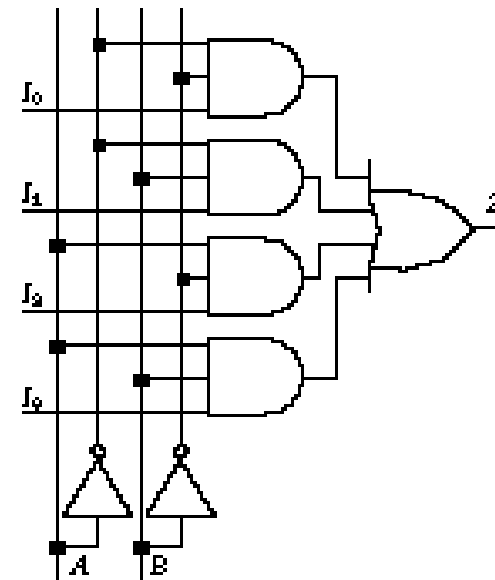
Przy pomocy multipleksera m wejściowego zrealizować funkcję logiczną $m+1$ zmiennych

Multiplexer. Przykłady realizacji.



Realizacja z dekodерem

Realizacja na bramkach



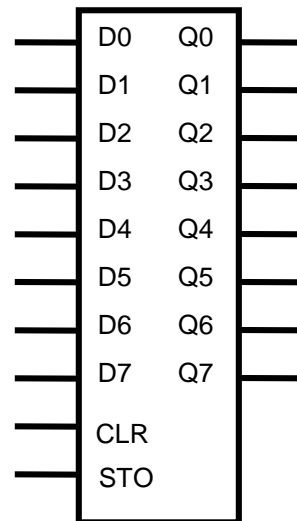
Rejestry

Rejestr jest elementem pamiętającym używanym do pamiętania i manipulacji więcej niż jednym bitem informacji.

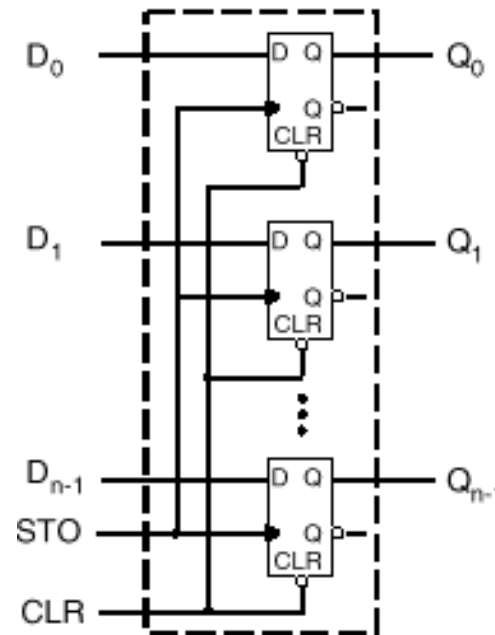
Rejestry można podzielić ze względu na sposób wpisu informacji na rejestry szeregowe i równoległe

Rejestry są realizowane jako układy przerzutników flip-flop wyposażone we wspólne sygnały sterujące wpisywaniem i czytaniem danych z rejestru (wspólne w tym znaczeniu, że wpływają na pracę poszczególnych przerzutników w taki sam sposób).

Przykład realizacji rejestru równoległego

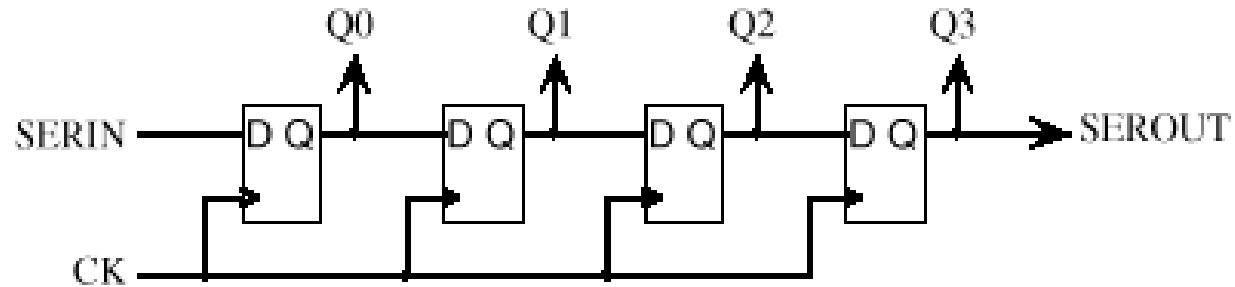


Sygnal STO jest w istocie sygnałem zegarowym

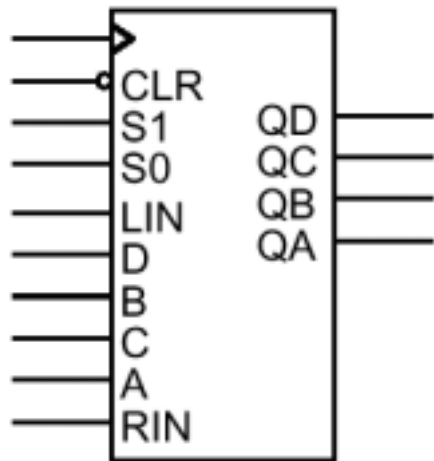


Różne rodzaje rejestrów

Rejestr szeregowy



Rejestr szeregowo-równoległy



S1	S0	Action	QA*	QB*	QC*	QD*
0	0	hold	QA	QB	QC	QD
0	1	shift right	RIN	QA	QB	QC
1	0	shift left	QB	QC	QD	LIN
1	1	load	A	B	C	D

Liczniki

Liczniki jest rejestrem zdolnym do automatycznego zwiększania lub zmniejszania zapisanej w nim wartości.

$Q \leftarrow Q \text{ plus } n$

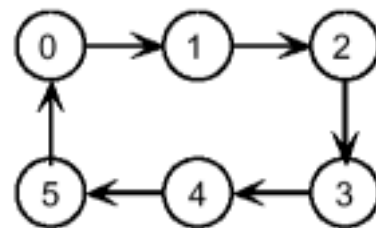
$Q \leftarrow Q \text{ minus } n$

Funkcję zwiększania i zmniejszania definiuje sposób w jaki licznik koduje całkowite wartości liczbowe.

Przykład: liczniki binarne, liczniki dziesiętne (BCD), w kodzie Grey'a.

Przykład. Licznik modulo 6

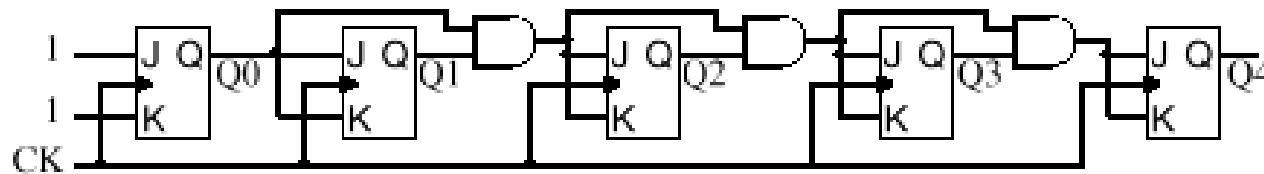
000	001
001	010
010	011
011	100
100	101
101	000
110	x x x
111	x x x



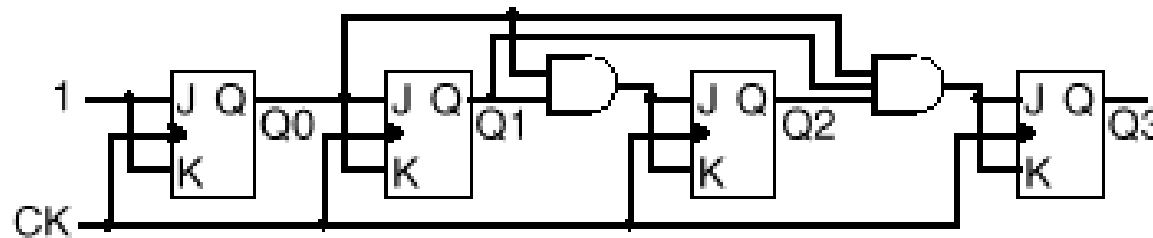
State Diagram

Realizacje liczników binarnych

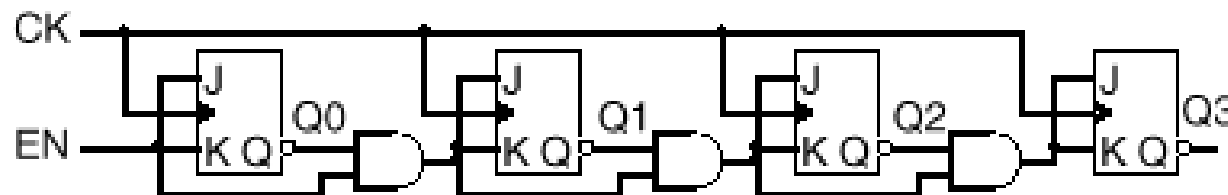
Licznik binarny modulo 16 z przeniesieniem szeregowym



Licznik binarny modulo 16 z przeniesieniem równoległym

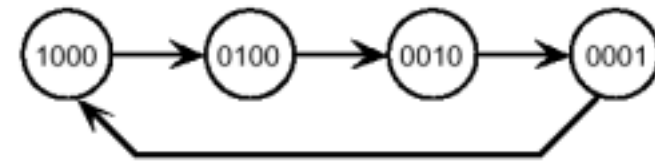
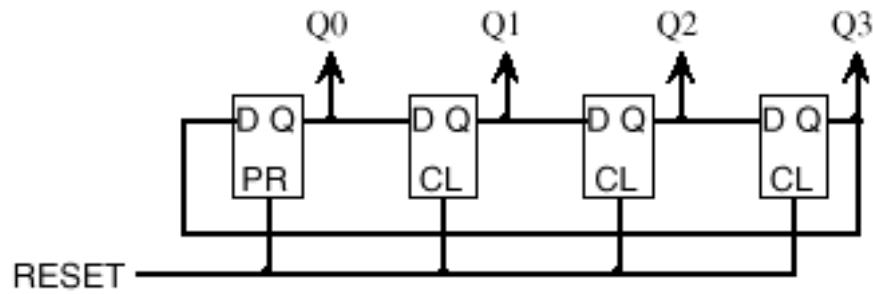


Licznik binarny modulo 16 zliczący w dół.

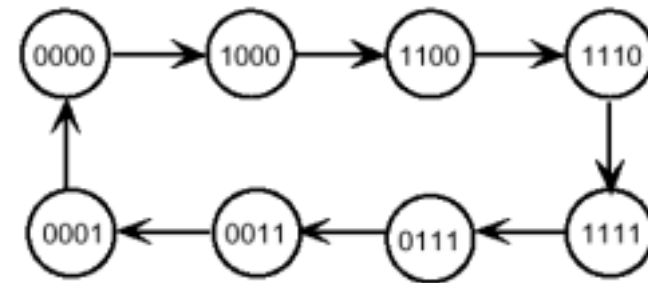
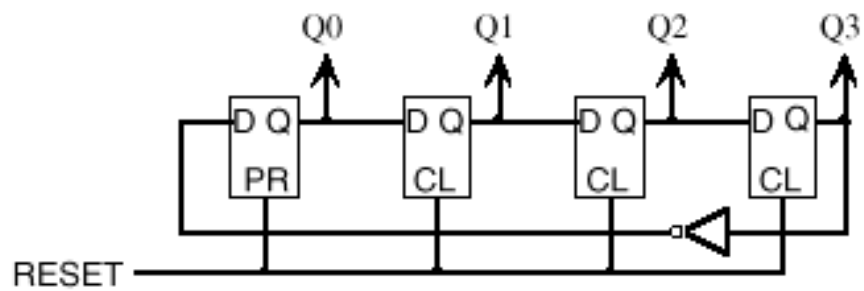


Liczniki specjalne

Licznik pierścieniowy.



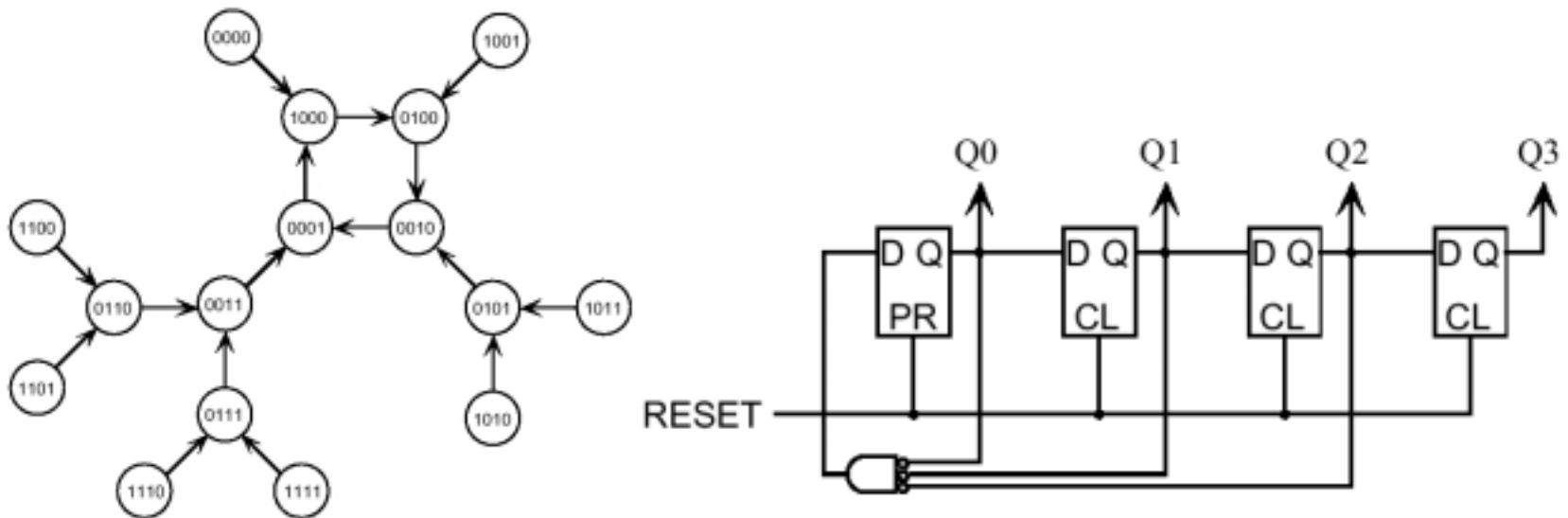
Licznik Johnsona.



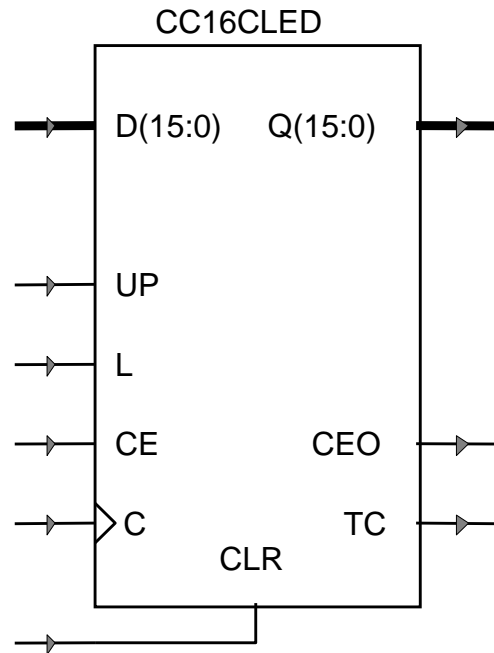
Samokorekcja liczników

Licznik samokorygujący to taki który z każdego możliwego stanu przerzutników, takiego, że stan ten nie jest ważnym stanem licznika powraca do stanu który jest poprawnym stanem licznika

Przykład. Samokorygujący licznik pierścieniowy



Przykład licznika



16-bitowy licznik dwukierunkowy
z ładowaniem równoległym i kasowaniem

$$TC = (Q_z \cdot Q_{(z-1)} \cdot Q_{(z-2)} \cdot \dots \cdot Q_0 \cdot UP) + (Q_z \cdot Q_{(z-1)} \cdot Q_{(z-2)} \cdot \dots \cdot Q_0 \cdot \overline{UP})$$

$$CEO = TC \cdot CE$$

Ćwiczenie

Zaproponować sposób kaskadowania przedstawionego licznika.