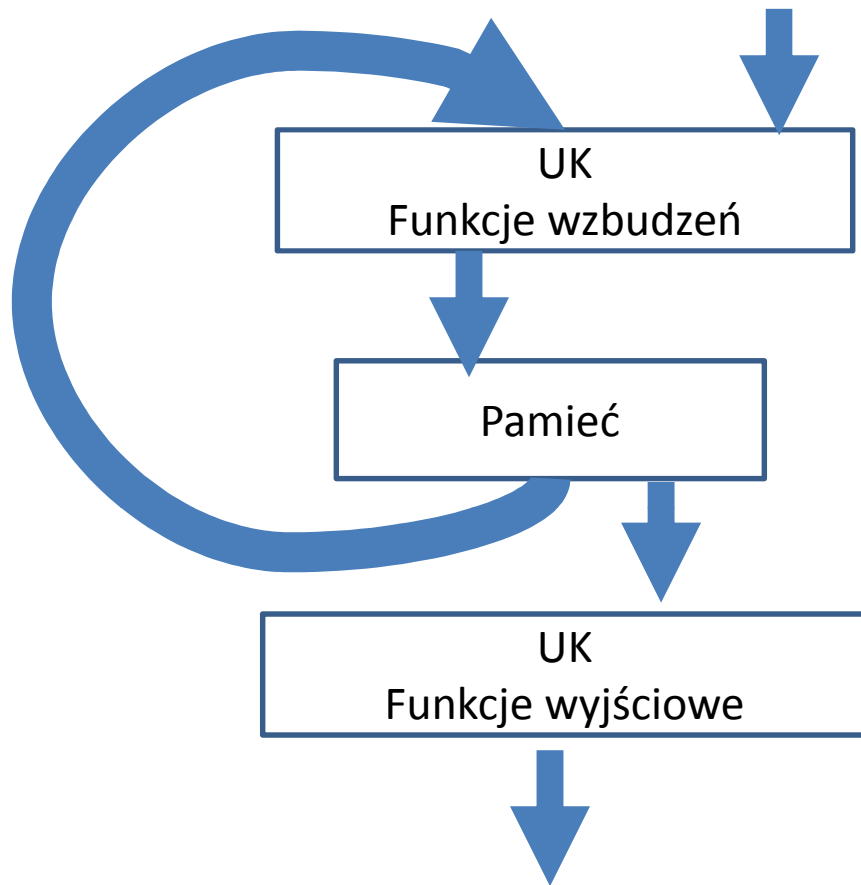


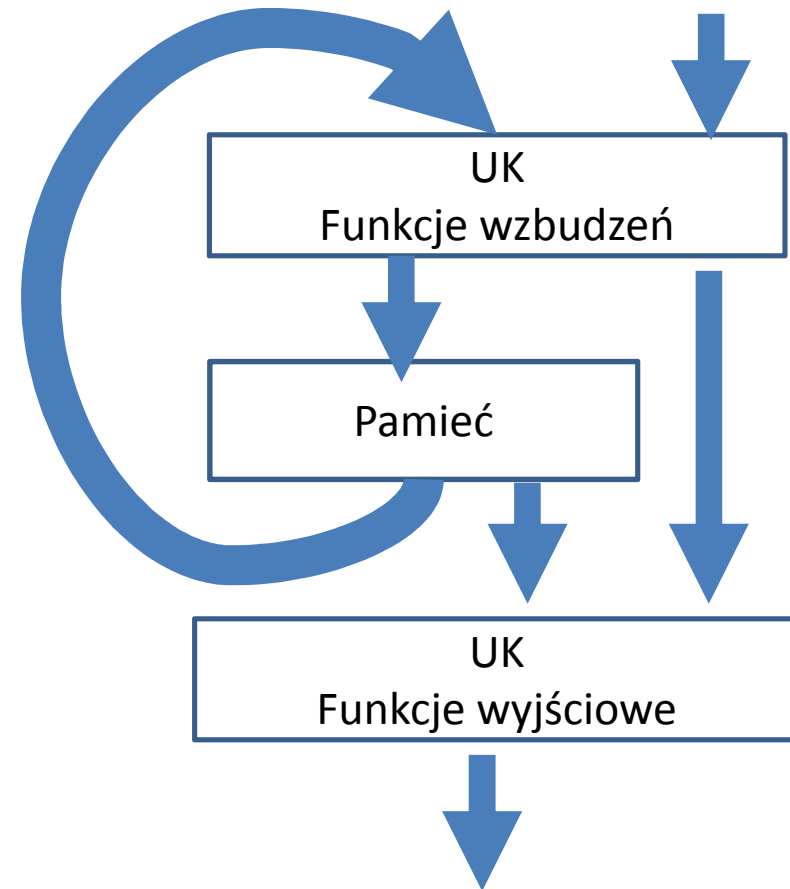
Podstawowe moduły układów cyfrowych
układy sekwencyjne cz.2
Projektowanie automatów
synchronicznych

Rafał Walkowiak

Wersja 0.2 2014/2015

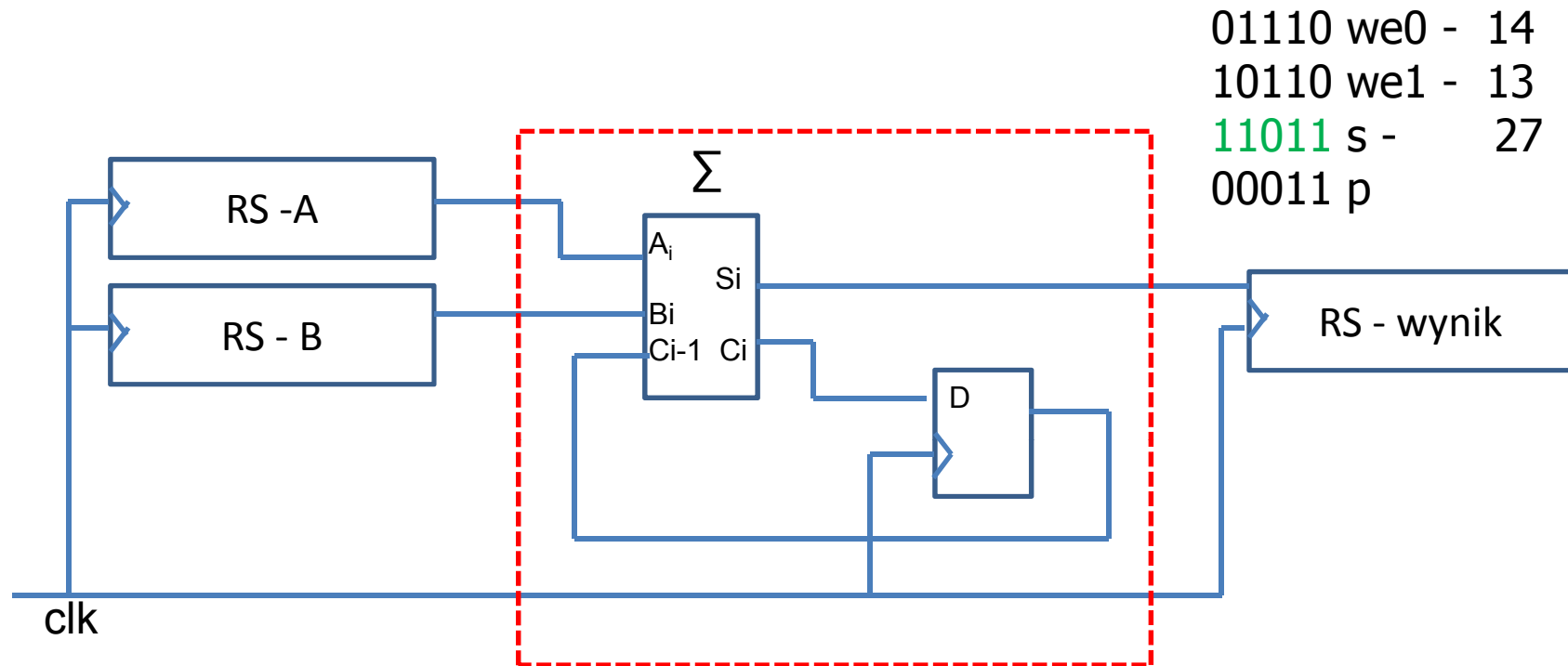


- Automat Moore'a



- Automat Mealy'ego

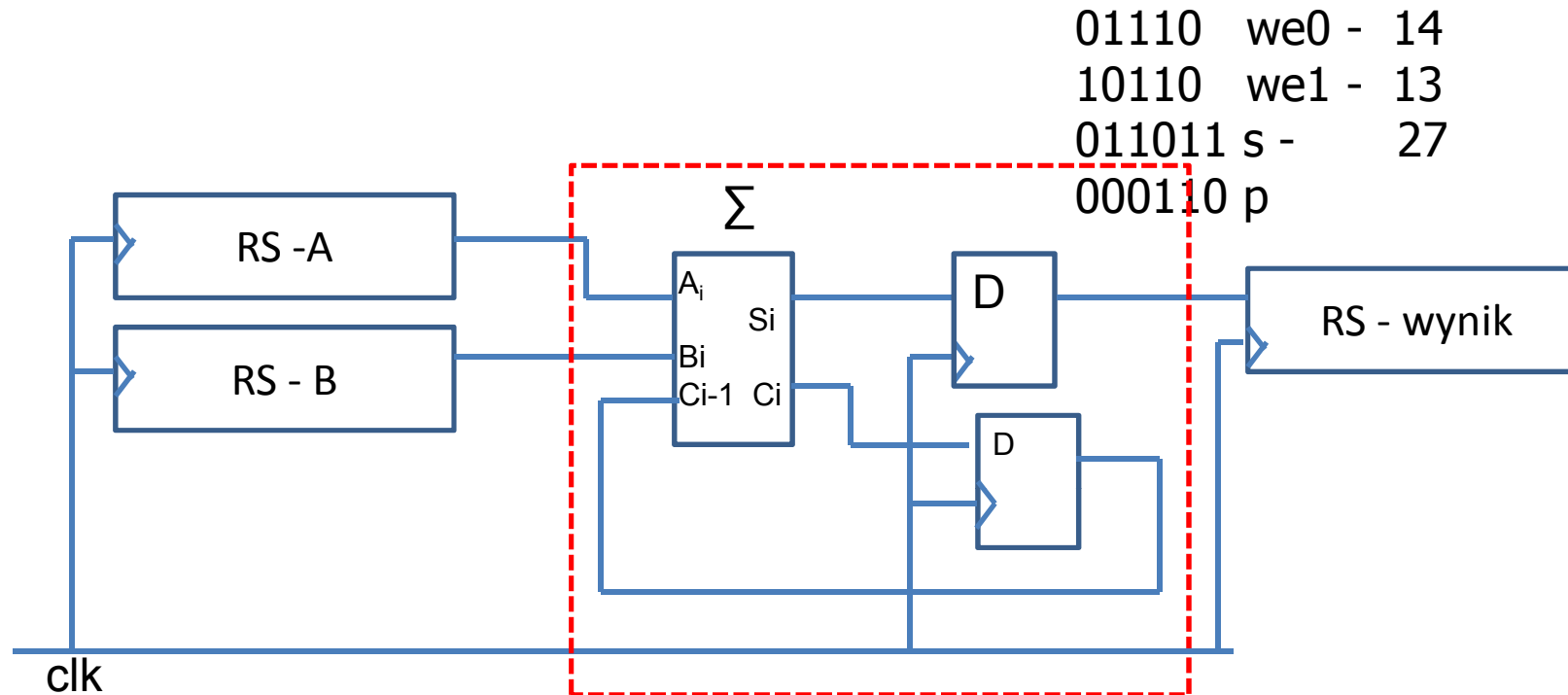
sumator szeregowy Mealy'ego



Sumator jednobitowy pełny:

- generuje wynik sumowania dla liczb dowolnego rozmiaru w czasie zależnym od rozmiaru liczb,
- liczby podawane są począwszy od najmłodszego bitu

sumator szeregowy automat Moore'a

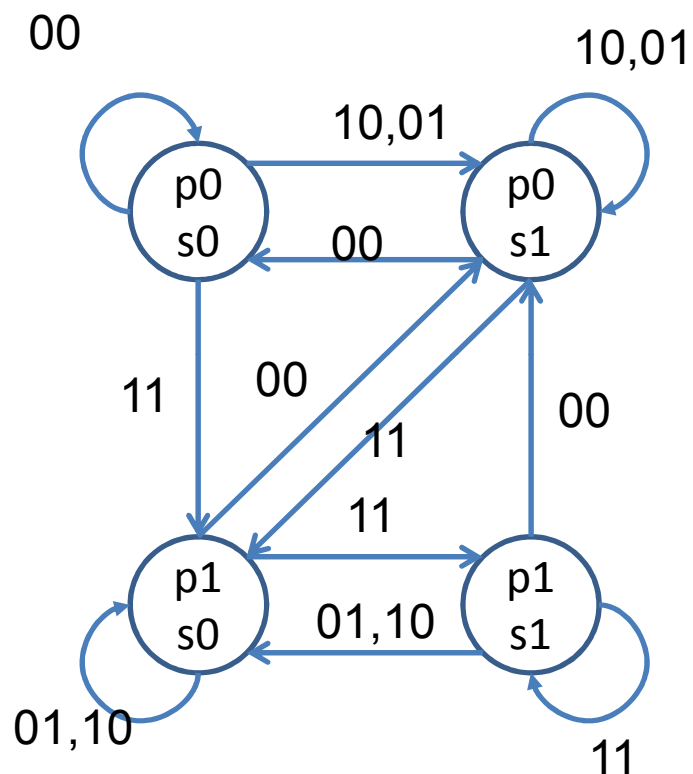


Sumator jednobitowy pełny:

- generuje wynik sumowania dla liczb dowolnego rozmiaru w czasie zależnym od rozmiaru liczb,
- liczby podawane są począwszy od najmłodszego bitu

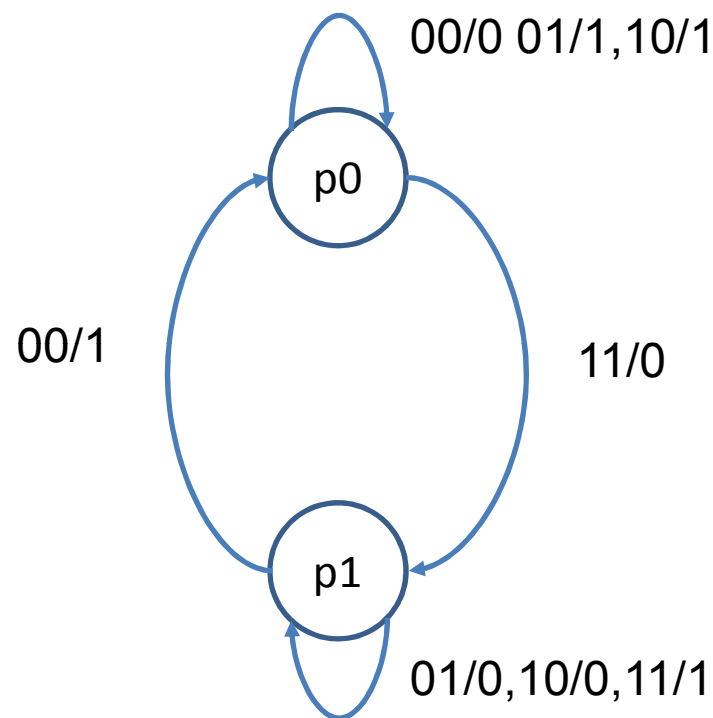
Graf automatu – sumator sekwencyjny

Model Moore'a



Stany oznaczone przez przeniesienie i sumę

Model Mealy'ego

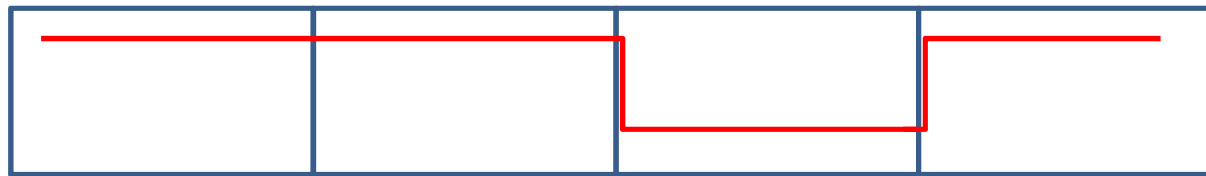


Stany oznaczone przez przeniesienie, przejścia oznaczone wejściami/sumą, w obu stanach wyjście sumy może mieć różne wartości w zależności od stanu wejść

Procedura projektowania automatu

- Opis działania
- Sporządzenie grafu przejść i tablicy przejść
- Redukcja stanów
- Kodowanie stanów (liczba i typ przerzutników)
- Tablica wzbudzeń (przejść) i wyjść
- Wyznaczenie funkcji wzbudzeń i funkcji wyjść
- Realizacja układu
- Testowanie poprawności działania

Wykrywanie sekwencji



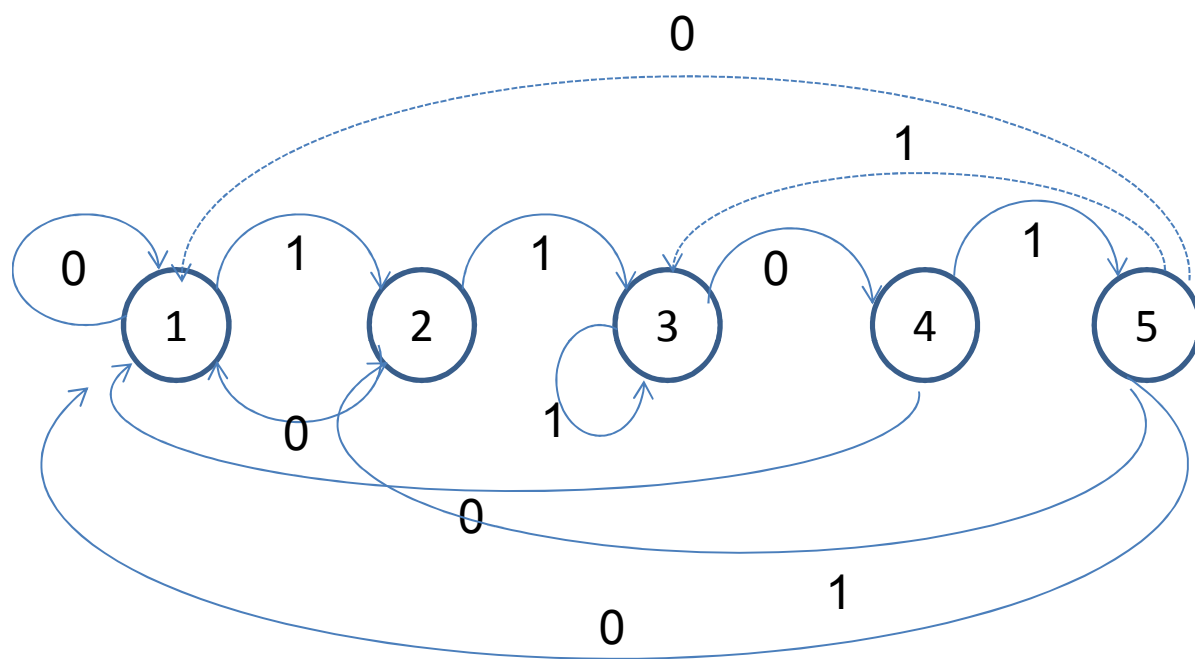
DANE
SEKWENCJA
1101



Dane analizowane
zboczem
narastajacym



Automat Moore'a do wykrywania sekwencji bitów **1101**



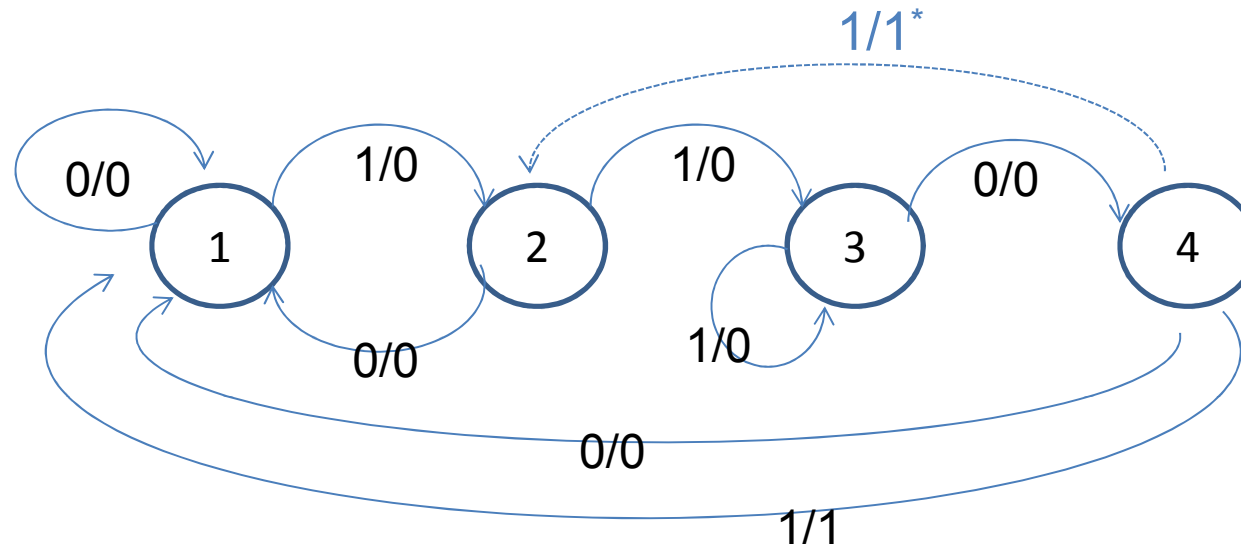
Stan bieżący	We=0	We=1
1	1	2
2	1	3
3	4	3
4	1	5
5	1	2

Graf stanów i tablica przejść

Linie ciągłe – wykrywanie nie nakładających się ciągów

Linie przerywane - wykrywanie nakładających się ciągów

Automat Mealego'a do wykrywania sekwencji bitów 1101



Stan bieżący	We=0	We=1
1	1/0	2/0
2	1/0	3/0
3	4/0	3/0
4	1/0	1/1

Linie ciągłe – wykrywanie nie nakładających się ciągów

Linie przerywane - wykrywanie nakładających się ciągów

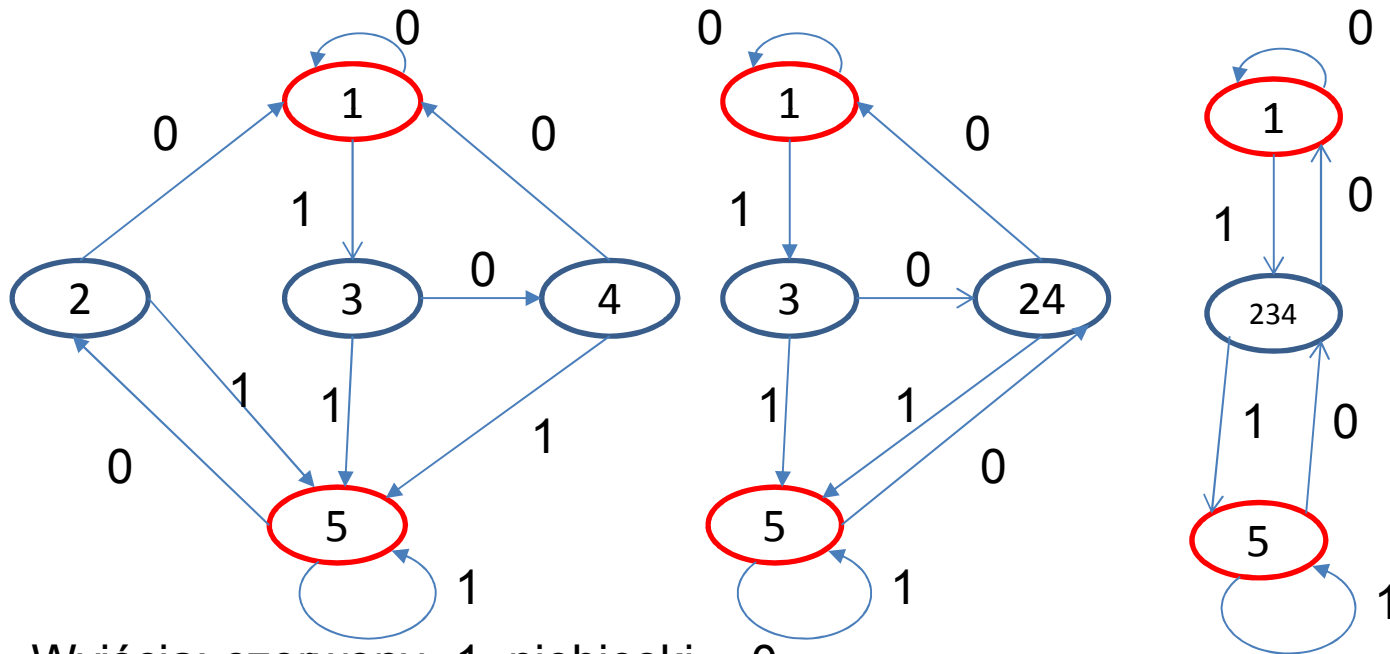
Sygnały wyjściowe są związane ze stanami **bieżącymi** i z wartościami sygnałów wejściowych. **Po wykryciu ciągu w stanie 4 na wyjściu pojawia się 1 i następuje przejście do stanu 1 (bez 1 na wyjściu).**

*Jeżeli dopuszczamy możliwość nakładania się sekwencji przejście następuje do stanu 2.

Redukcja stanów automatu

Dwa stany są równoważne (i mogą być zastąpione jednym stanem) jeżeli dla każdej kombinacji wejściowej:

- dają taką samą wartość wyjścia oraz
- przenoszą automat do tego samego stanu lub stanów równoważnych



Wyjścia: czerwony -1, niebieski - 0

Wyznaczanie stanów równoważnych

1. W tablicy trójkątnej dla każdej pary stanów określ warunki ich równoważności – równoważności stanów wyjściowych.
2. Dla pola pary stanów różniących się wartością wyjścia zaznacz X – brak równoważności.
3. Dla każdego pola różnego od X zbadaj w tablicy możliwość równoważności – brak X w polu odpowiadającym parze stanów.
4. Powtarzaj punkt 3 aż będzie on przynosił efekty.
5. Pozycje nieskreślone określają stany równoważne.

UWAGA: Relacja równoważności jest przechodnia w przypadku pełnej określoności tablicy przejść (brak przejść do stanów dowolnych). W przeciwnym przypadku konieczne sprawdzenie warunku pokrycia i zamknięcia dla wyznaczenia klasy stanów równoważnych (rozszerzenie pary stanów) - porównaj projektowanie automatów asynchronicznych.

Redukcja stanów automatu Moora

przykład

Stan akt	00	01	11	10	wyjsc ie
1	5	3	2	1	0
2	5	3	1	4	0
3	3	4	4	5	1
4	5	3	2	2	0
5	6	7	1	1	0
6	3	3	1	7	0
7	7	1	1	5	1

2	12 14					
3	X	x				
4	12	12 24	x			
5	56 37 12	56 37 14	x	56 37 12		
6	35 12 17	35 47	x	35 12 27	36 37 17	
7	x	x	37 14	x	x	x
	1	2	3	4	5	6

Redukcja stanów automatu Moora -przykład

2	12 14					
3	X	x				
4	12	12 24	x			
5	56 37 12	56 37 14	x	56 37 12		
6	35 12 17	35 47	x	35 12 27	36 37 17	
7	x	x	37 14	x	x	x
	1	2	3	4	5	6

Kolejność wykreślenia:

61,62,64(35), 56(36),51,52,54(56)

Stan akt	00	01	11	10	wyjsc ie
1	5	3	2	1	0
2	5	3	1	4	0
3	3	4	4	5	1
4	5	3	2	2	0
5	6	7	1	1	0
6	3	3	1	7	0
7	7	1	1	5	1

Stan akt	00	01	11	10	wyjsc ie
124	5	37	124	124	0
37	37	124	124	5	1
5	6	37	124	124	0
6	37	37	124	37	0

Automat Mealego wykrywania sekwencji (poprzedni slajd) – próba minimalizacji stanów

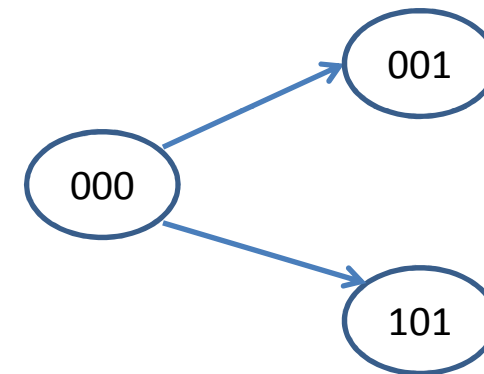
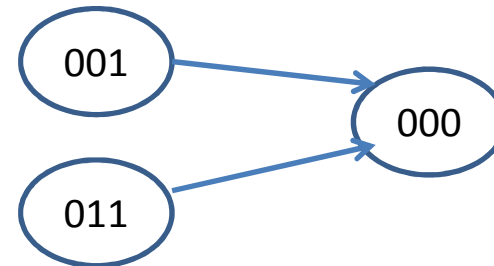
2	23		
3	14 23	14	
4	x	x	x
	1	2	3

Stan bieżący	We=0	We=1
1	1/0	2/0
2	1/0	3/0
3	4/0	3/0
4	1/0	1/1

Stany 14 nie mogą być równoważne gdyż występuje z nich przejście pod wpływem We=1 do stanów 2,1 z różnymi wartościami na wyjściach. W konsekwencji nie równoważna jest są również para stanów 2,3.

Kodowanie stanów -zasady

- Zasada 1 – Należy przyporządkować stanom, które mają ten sam stan następny, słowa kodowe różniące się tylko wartością jednego bitu.
- Zasada 2- Należy przyporządkować stanom następnym, które mają ten sam stan bieżący, słowa kodowe różniące się tylko wartością jednego bitu.
- Stosowanie powyższych zasad ma na celu uzyskiwanie najprostszych funkcji wzbudzeń wymuszających odpowiednie stany następne przerzutników.



W sytuacji gdy każdy stan ma tylko jeden stan następny warto zastosować dla kolejnych stanów słowa kodowe różniące się na jednej pozycji np. wg kodu Greya. Kodowanie 1 z N – każdy stan reprezentowany przez wyróżniony stan jednego przerzutnika.

Programowany generator impulsów

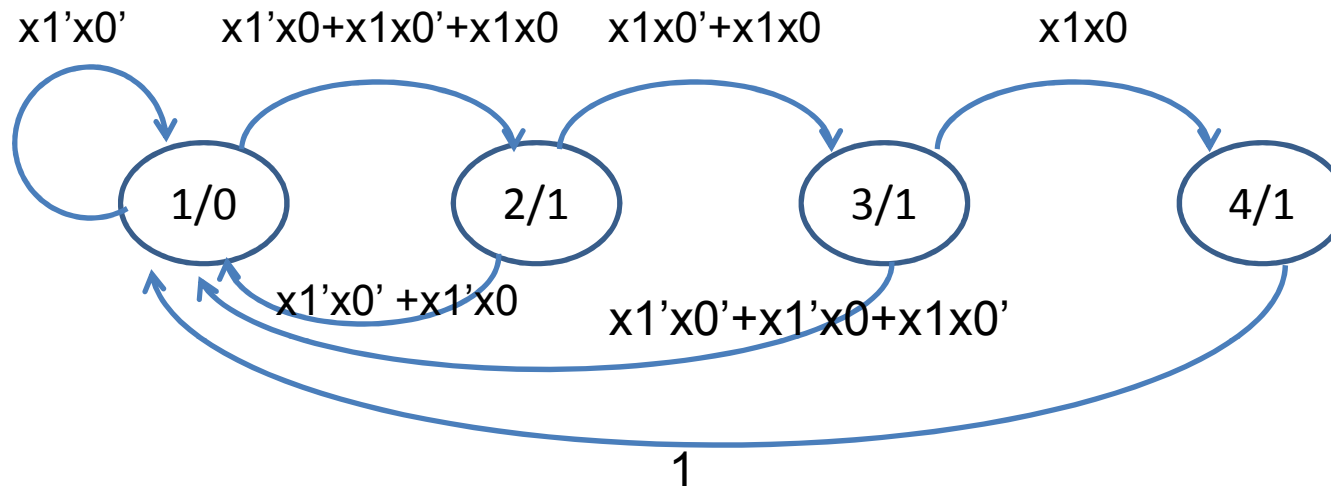
Opis działania układu:

- Układ generuje na wyjściu impuls o długości trwania mierzonej w jednostkach równych wielokrotności zegara wejściowego synchronizującego pracę układu.
- Czas trwania jest kodowany na dwóch wejściach informacyjnych układu.
- Dla wejść równych zero impuls nie jest generowany

Sygnały :

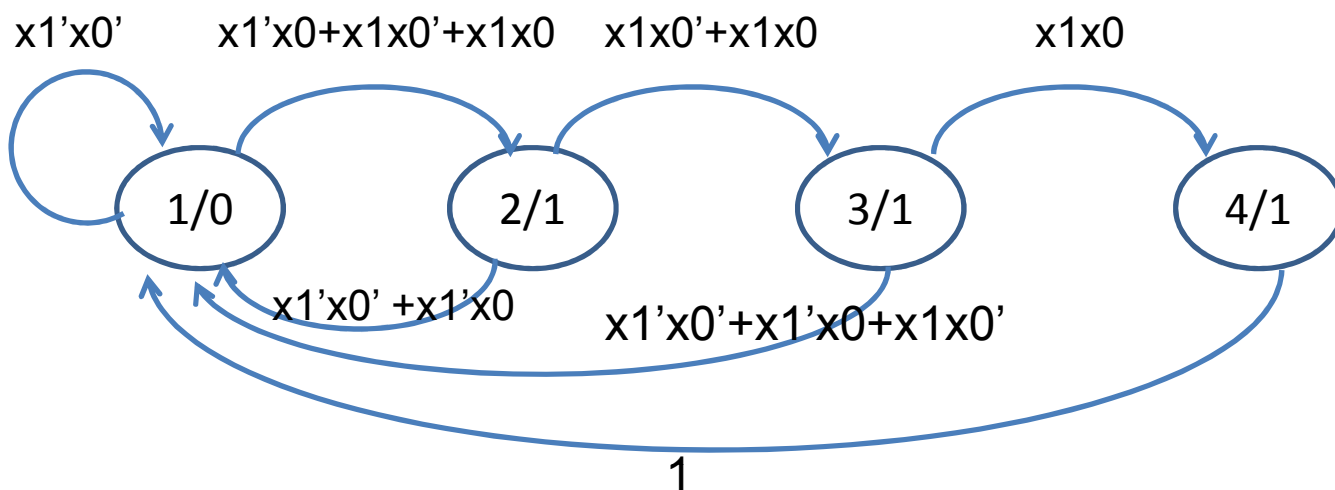
- 2 wejścia kodujące czas trwania impulsu
- Wyjście

Graf stanów z równaniami określającymi przejścia



Generator impulsów z sterowaną wejściami długością trwania impulsu (długość kodowana binarnie). Równania określająca przejścia muszą wyczerpywać wszystkie możliwe kombinacje wartości wejść dla jednoznacznego określenia przejścia

Programowany generator impulsów



Stan bieżący	00	01	11	10	Wejścia ←
00	00	01	01	01	
01	00	00	11	11	
11	00	00	10	00	
10	00	00	00	00	

Graf stanów i tablica stanów generatora impulsów w postaci zakodowanej – kodowanie kolejnych stanów kodem Graya

Programowany generator impulsów

Stan bieżący	00	01	11	10
00	00	01	01	01
01	00	00	11	11
11	00	00	10	00
10	00	00	00	00

Realizacja
przy użyciu
przerzutników
jk

J0	we	01	11	10
stan	0	1	1	1
01	-	-	-	-
11	-	-	-	-
10	0	0	0	0

K0	we	01	11	10
stan	-	-	-	-
01	1	1	0	0
11	1	1	1	1
10	-	-	-	-

Q	Q+	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

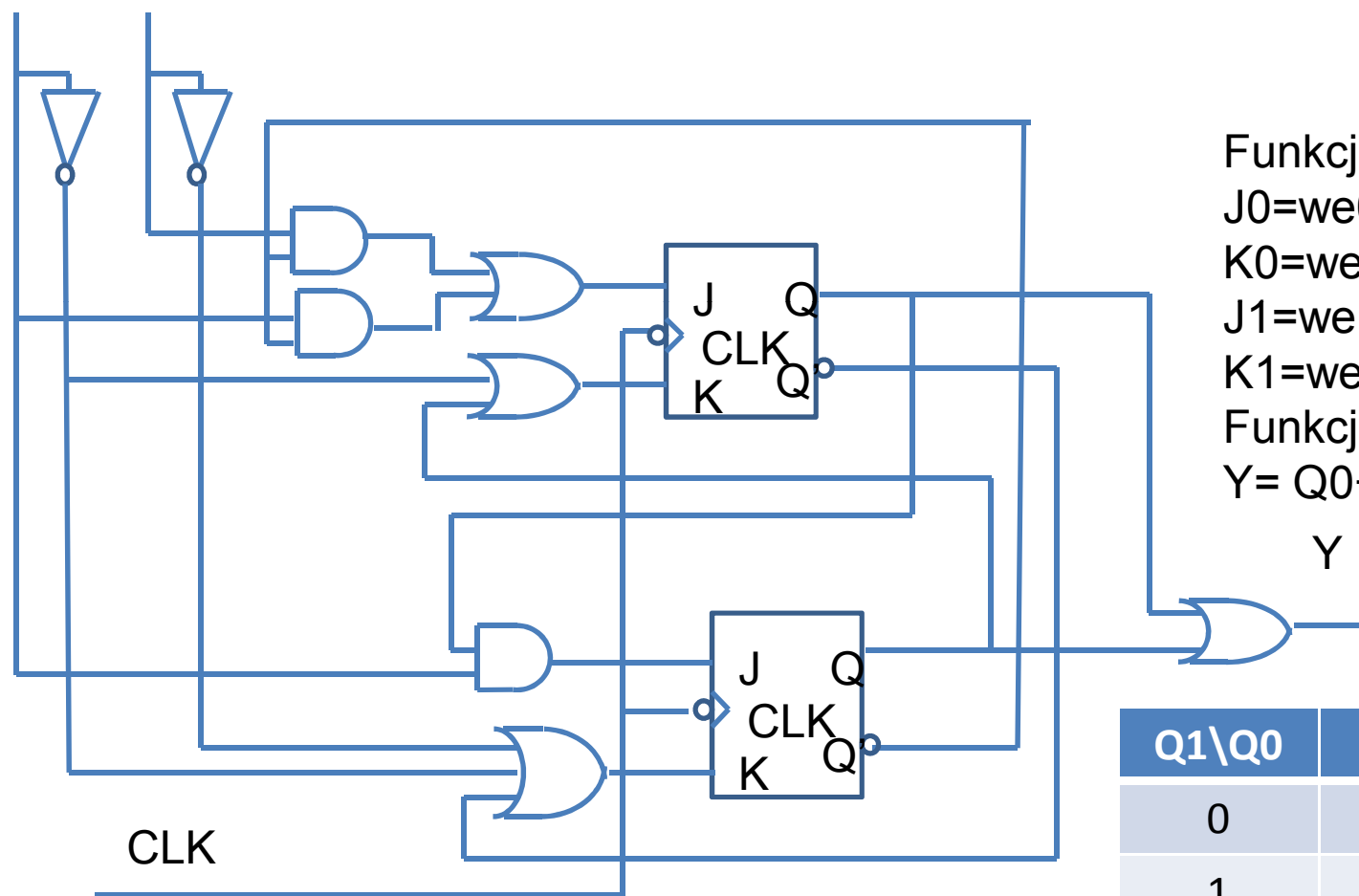
J1	we	01	11	10
stan	0	0	0	0
01	0	0	1	1
11	-	-	-	-
10	-	-	-	-

K1	we	01	11	10
stan	-	-	-	-
01	-	-	-	-
11	1	1	0	1
10	1	1	1	1

$$\begin{aligned}
 J0 &= we0Q1' + we1Q1' \\
 K0 &= we1' + Q1 \\
 J1 &= we1Q0 \\
 K1 &= we1' + we0' + Q0'
 \end{aligned}$$

Programowany generator impulsów

WE1 WE0



Funkcje wzbudzeń

$$J0 = we0 Q1' + we1 Q1'$$

$$K0 = we1' + Q1$$

$$J1 = we1 Q0$$

$$K1 = we1' + we0' + Q0'$$

Funkcja wyjścia

$$Y = Q0 + Q1$$

Y

Q1\Q0	0	1
0	0	1
1	1	1