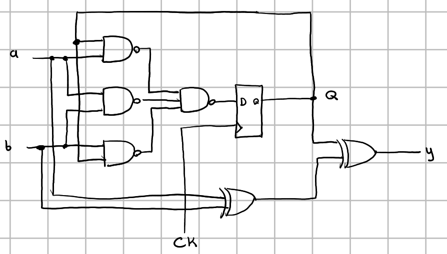
Automaty asynchroniczne

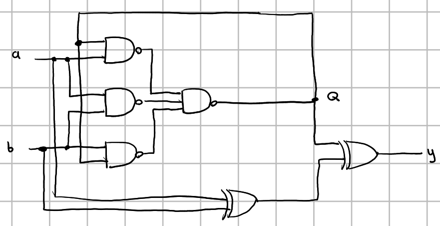
W automacie musi być sprzężenie zwrotne,
nie musi być przerzutnika

Zaletą automatu synchronicznego jest natychmiastowa reakcja bez czekania na zegar ale powstają przy tym nowe problemy.

Synchroniczny

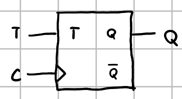


Asynchroniczny



Przykład projektu automatu asynchronicznego

asynchroniczny przerzutnik T
aktywowany narastającym zboczem sygnału zegarowego

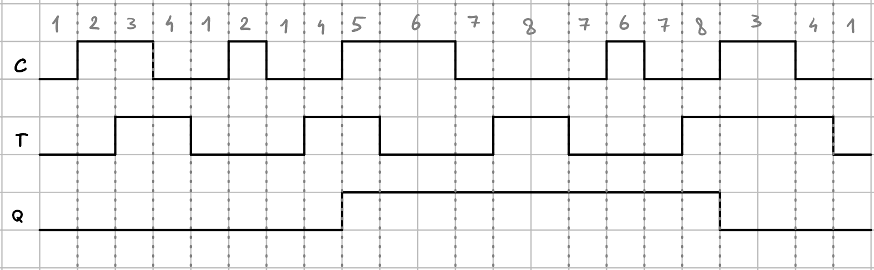


Dwa sygnały między nie mogą się zmienić w tym samym momencie

- układy synchroniczne nie są czułe na szybkie zmiany sygnałów
- układy asynchroniczne reagują na szybkie zmiany sygnałów

Stan - przedział czasu, kiedy automat jest w stabilnej sytuacji
(między pionowymi kreskami na wykresie czasowym)

Wykres czasowy
pozwala na precyzyjne opisanie działania układu



Przejście od wykresu czasowego do tabeli pierwotnej
(zawsze automat typu Moore'a)

S \ CT	00	01	11	10	Q
1	1	4	-	2	0
2	1	-	3	2	0
3	-	4	3	2	0
4	1	4	5	-	0
5	-	8	5	6	1
6	7	-	5	6	1
7	7	8	-	6	1
8	7	8	3	-	1

stan stabilny - kółko

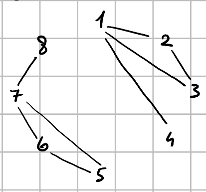
zmiana 2 bitów względem stanu stabilnego (niemożliwe) - kreska

trzeba analizować wykres czasowy dopóki nie wypełni się tabeli

Minimalizacja liczby stanów
(uproszczona)

S \ CT	00	01	11	10	Q
1	1	4	–	2	0
2	1	–	3	2	0
3	–	4	3	2	0
4	1	4	5	–	0
5	–	8	5	6	1
6	7	–	5	6	1
7	7	8	–	6	1
8	7	8	3	–	1

graf zgodności



przykładowy wybór nowych stanów

A	{1,4}	A	{4}
B	{2,3}	B	{1,2,3}
C	{5,6}	C	{5,6,7}
D	{7,8}	D	{8}

Minimalna tabela

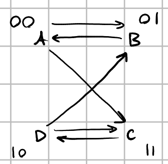
S \ CT	00	01	11	10	Q
A	A	A	C	B	0
B	A	A	B	B	0
C	D	D	C	C	1
D	D	D	B	C	1

Podstawa do dalszego testowania

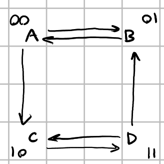
Kodowanie stanów

Sposób kodowania ma znaczenie

Przy złym kodowaniu automat nie zadziała poprawnie



wyścig krytyczny



dobre kodowanie

kwadrat narysowany tak, żeby wzdłuż przekątnej zmieniał się tylko 1 bit

Naiwna funkcja wzbudzeń i realizacja

Q_1Q_0 \ CT	00	01	11	10	Q
00	00	00	10	01	0
01	00	00	01	01	0
11	11	11	01	10	1
10	11	11	10	10	1

zakodowana tabela

Q_1Q_0 \ CT	00	01	11	10
00	0	0	1	0
01	0	0	0	0
11	1	1	0	1
10	1	1	1	1

Q_1'

Q_1Q_0 \ CT	00	01	11	10
00	0	0	0	1
01	0	0	1	1
11	1	1	1	0
10	1	1	0	0

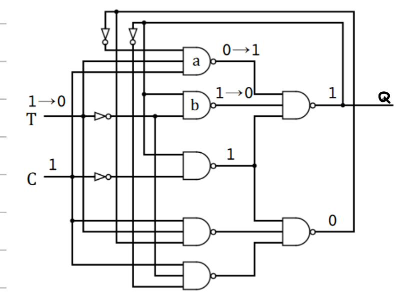
Q_0'

$$Q_1' = Q_1\overline{C} + Q_1\overline{T} + \overline{Q}_0 CT$$

$$Q_0' = Q_1\overline{C} + Q_0 CT + \overline{Q}_1 C\overline{T}$$

$$Q = Q_1$$

niepoprawna realizacja



Jeśli bramka a przełączy się szybciej niż b to na wyjściu Q pojawi się szpilka – hazard statyczny

Automat może przełączyć się w inny stan

Likwidacja hazardu

Trzeba zlokalizować w mapach Karnaugha gdzie występuje hazard i dodać nowe implikanty tam, gdzie sąsiadują jedynki różnych implikantów

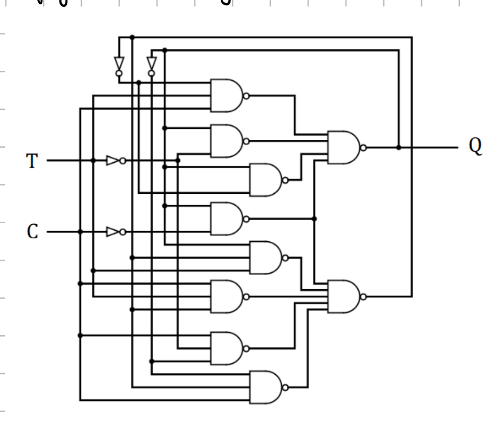
$Q_1Q_0 \backslash CT$	00	01	11	10
00	0	0	1	0
01	0	0	0	0
11	1	1	0	1
10	1	1	1	1

Q_1'

$Q_1Q_0 \backslash CT$	00	01	11	10
00	0	0	0	1
01	0	0	1	1
11	1	1	1	0
10	1	1	0	0

Q_0'

poprawna realizacja



$$Q_1' = Q_1\overline{C} + Q_1\overline{T} + \overline{Q}_0 CT + Q_1\overline{Q}_0$$

$$Q_0' = Q_1\overline{C} + Q_0 CT + \overline{Q}_1 C\overline{T} + Q_1Q_0\overline{T} + \overline{Q}_1Q_0C$$

$$Q = Q_1$$

Założenia projektowe Huffmana

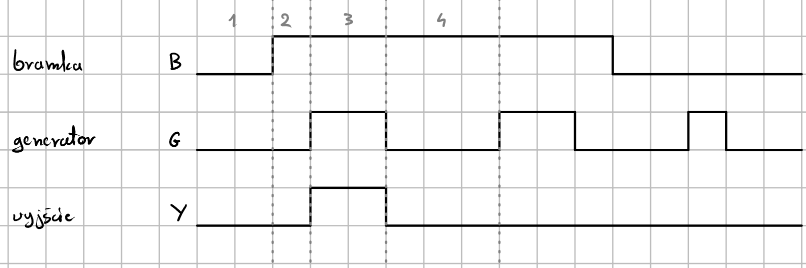
- Zmiana stanu wejść na skutek zmiany tylko jednego sygnału wejściowego
- Następna zmiana na wejściu po czasie τ potrzebnym do ustabilizowania się stanu wewnętrznego automatu
- Sygnały sprzężeń zwrotnych mogą zmieniać się w dowolnej kolejności
- Opóźnienia układu są związane z liniami sprzężenia zwrotnego

Algorytm projektowania układów asynchronicznych

1. Sporządzić wykres czasowy lub graf automatu na podstawie opisu słownego
2. Oznaczyć stany na wykresie czasowym lub grafie
3. Utworzyć pierwotną tabelę przejść i wyjść na podstawie wykresu lub grafu
4. Zminimalizować tabelę przejść i wyjść (liczbę stanów)
5. Zakodować stany tak, żeby wyeliminować wyścigi krytyczne
6. Wybrać typ realizacji i wyznaczyć funkcję sprzężenia zwrotnego lub funkcje wzbudzeń przerzutników
7. Wyeliminować hazard
8. Wyznaczyć testy uruchomieniowe

Stany na wykresie czasowym

układ przepuszcza pierwszy impuls generatora po otwarciu bramki



Stany 2 i 4 mają identyczny układ wejść i wyjść
ale niosą inną informację – nie są jednakowe

Może być więcej niż 2^n stanów, bo te same układy
mogą różnić się znaczeniem

Tabela pierwotna

- Tabela automatu Moore'a
- 1 stan stabilny w każdym wierszu
  jeśli jest więcej to była niejawna minimalizacja
- W wierszu są zdefiniowane tylko te przejścia,
  które różnią się 1 bitem ze stanem stabilnym
- W kolumnie, gdzie są zdefiniowane jakieś przejścia
  musi być conajmniej 1 stan stabilny
- W kolumnie mogą być tylko przejścia do stanów,
  które są stabilne w tej kolumnie

s \ ab	00	01	11	10	Y
1	1	4	–	3	0–
2	–	9	2	3	01
3	1	–	2	3	01
4	1	4	7	–	00
5	1	5	8	–	0–
6	6	5	–	3	10
7	–	4	7	3	01
8	–	5	8	3	–1
9	6	9	2	–	11

W ten sposób można łatwo rozpoznać niepoprawną tabelę

Minimalizacja liczby stanów

1. Połączenie stanów pseudorównoważnych – warunkowe
2. Połączenie stanów pseudozgodnych – bezwarunkowe

Tabela trójkątna nie musi dać dobrego wyniku

Stany pseudorównoważne
stany zgodne, które mają stan stabilny w tej samej kolumnie

s \ ab	00	01	11	10	Y
1	1	4	–	3	0–
2	–	10	2	3	01
3	1	–	2	3	01
4	1	4	7	–	00
5	1	5	9	–	0–
6	6	5	–	8	10
7	–	4	7	8	01
8	1	–	2	8	–1
9	–	5	9	3	–1
10	6	10	2	–	11

Pseudorównoważne

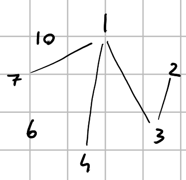
- ✓ 4,5 jeśli 7,9
- ✗ 2,7 jeśli 4,10 i 3,8
- ✗ 2,9 jeśli 5,10
- ✓ 7,9 jeśli 4,5 i 3,8
- ✓ 3,8

różne wyjścia – nie są zgodne

	s \ ab	00	01	11	10	Y
	1	1	4	–	3	0–
	2	–	10	2	3	01
(3,8)	3	1	–	2	3	01
(4,5)	4	1	4	7	–	00
	6	6	4	–	3	10
(7,9)	7	–	4	7	3	01
	10	6	10	2	–	11

Stany pseudozgodne
(po połączeniu stanów pseudorównoważnych)

S \ ab	00	01	11	10	Y
1	1	4	-	3	0-
2	-	10	2	3	01
3	1	-	2	3	01
4	1	4	7	-	00
6	6	4	-	3	10
7	-	4	7	3	01
10	6	10	2	-	11



graf skracania

	S \ ab	00	01	11	10	Y
(1,4)	A	A	A	D	B	00
(2,3)	B	A	E	B	B	01
(6)	C	C	A	-	B	10
(7)	D	-	A	D	B	01
(10)	E	C	E	B	-	11

zminimalizowana tabela przejść i wyjść automatu

Kodowanie stanów

Kodowanie musi być takie, żeby każda zmiana stanu powodowała zmianę tylko jednego sprzężenia zwrotnego

Kiedy zmienia się więcej niż 1 bit "na raz" to występuje wyścig - nie wiadomo który zmieni się pierwszy

00 → $\underline{01}$ → 11 albo 00 → $\underline{10}$ → 11

stan pośredni stan pośredni

Wyścig krytyczny - jeden ze stanów pośrednich jest stabilny
musi zostać usunięty z układu

Wyścig niekrytyczny - żaden ze stanów pośrednich nie jest stabilny
nie trzeba ich usuwać
w kolumnie tabeli jest tylko 1 stan stabilny

Usuwanie wyścigu

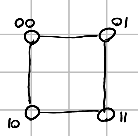
Zapewnienie, że każde przejście w tabeli zmienia tylko 1 bit stanu automatu

- Inne rozmieszczenie stanów (zmiana kodowania)
- Przejścia cykliczne (przez istniejący stan)
- Dodanie nowych stanów

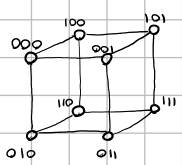
Hipersześcian

Wizualizacja pomaga sprawdzić czy kodowanie jest poprawne.

Poprawne kodowanie - wszystkie przejścia są wzdłuż krawędzi
(wierzchołki jednej krawędzi różnią się jednym bitem)

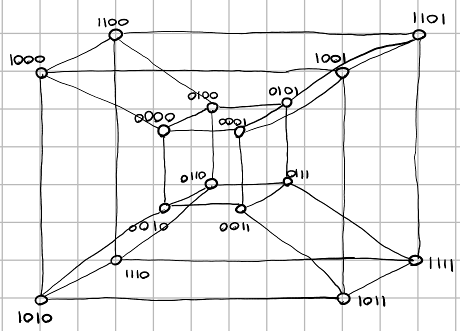


2 bity

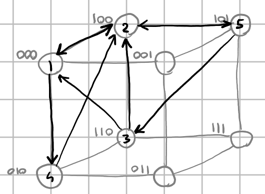


3 bity

4 bity



Przykładowy wybór kodowania



s \ ab	00	01	11	10	Y
1	1	1	4	2	00
2	1	5	2	2	01
3	3	1	–	2	10
4	–	1	4	2	01
5	3	5	2	–	11

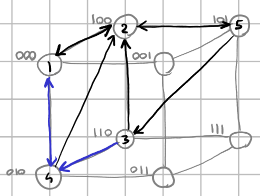
Wyścigi

$3 \to 1$ krytyczny

$4 \to 2$ niekrytyczny (1 stan stabilny w 4 kolumnie)

$5 \to 3$ krytyczny

Przejście cykliczne

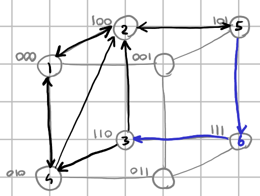


s \ ab	00	01	11	10	Y
1	1	1	4	2	00
2	1	5	2	2	01
3	3	4	–	2	10
4	–	1	4	2	01
5	3	5	2	–	11

Przejście $3 \to 4 \to 1$ zamiast $3 \to 1$

na wyjściu powstanie szpilka

Dodanie nowych stanów

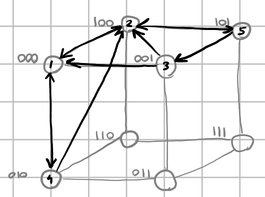


s \ ab	00	01	11	10	Y
1	1	1	4	2	00
2	1	5	2	2	01
3	3	4	–	2	10
4	–	1	4	2	01
5	6	5	2	–	11
6	3	–	–	–	11

Przejście $5 \to 6 \to 3$ zamiast $5 \to 3$

wyjście jest tak dobrane, żeby nie było szpilki

Zmiana ułożenia stanów



s \ ab	00	01	11	10	Y
1	1	1	4	2	00
2	1	5	2	2	01
3	3	1	–	2	10
4	–	1	4	2	01
5	3	5	2	–	11

Wyścigi

$3 \to 2$ niekrytyczny

$4 \to 2$ niekrytyczny

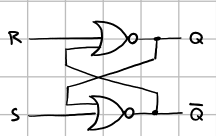
(oba wyścigi są w czwartej kolumnie)

Realizacja

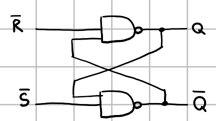
- Sprzężenia zwrotne
  zakodowana tabela jest jednocześnie funkcją wzbudzeń

- Asynchroniczne przerzutniki (Latch)

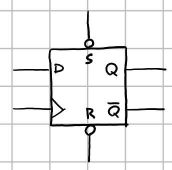
RS



$\bar{R}\bar{S}$



D z asynchronicznym ustawianiem i zerowaniem



Przerzutnik $\bar{R}\bar{S}$

$Q^t \to Q^{t+1}$	$\bar{S}$	$\bar{R}$
0 — 0	1	-
0 — 1	0	1
1 — 0	1	0
1 — 1	-	1

$\bar{S}$

0 dla pogrubionych jedynek
1 dla zer
\- dla niepogrubionych jedynek

$\bar{R}$

0 dla pogrubionych zer
1 dla jedynek
\- dla niepogrubionych zer

Realizacja na przerzutnikach $\bar{R}\bar{S}$

	s \ ab	00	01	11	10	Y
1	000	000	000	010	100	00
3	001	001	000	-	100	10
	011	-	-	-	-	-
4	010	-	000	010	100	01
	110	-	-	-	-	-
	111	-	-	-	-	-
5	101	001	101	100	-	11
2	100	000	101	100	100	01

	s \ ab	00	01	11	10
1	000	111	111	101	011
3	001	11-	111	-	011
	011	-	-	-	-
4	010	-	111	1-1	011
	110	-	-	-	-
	111	-	-	-	-
5	101	11-	-1-	-11	-
2	100	111	-10	-11	-11

$\bar{S}_2\bar{S}_1\bar{S}_0$

	s \ ab	00	01	11	10
1	000	---	---	-1-	1--
3	001	--1	--0	-	1-0
	011	-	-	-	-
4	010	-	-0-	-1-	10-
	110	-	-	-	-
	111	-	-	-	-
5	101	0-1	1-1	1-0	-
2	100	0--	1-1	1--	1--

$\bar{R}_2\bar{R}_1\bar{R}_0$

$\bar{S}_2 = \bar{a} + b$

$\bar{S}_1 = \bar{a} + \bar{b} + Q_2$

$\bar{S}_0 = a + \bar{b} + \overline{Q_2}$

$\bar{R}_2 = a + b$

$\bar{R}_1 = ab$

$\bar{R}_0 = \bar{a}\bar{b} + \bar{a}Q_2$

$y_1 = Q_0$

$y_0 = Q_2 + Q_1$

Nie ma hazardów ale mogłyby być

Realizacja na sprzężeniach zwrotnych

	s \ ab	00	01	11	10	Y
1	000	000	000	010	100	00
3	001	001	000	-	100	10
	011	-	-	-	-	-
4	010	-	000	010	100	01
	110	-	-	-	-	-
	111	-	-	-	-	-
5	101	001	101	100	-	11
2	100	000	101	100	100	01

$Q_2' = a\bar{b} + bQ_2 + aQ_2$

$Q_1' = ab\overline{Q_2}$

$Q_0' = \bar{a}\bar{b}Q_0 + \bar{a}bQ_2 + \bar{a}Q_2Q_0$

$y_1 = Q_0$

$y_0 = Q_2 + Q_1$

Niezależnie od sposobu realizacji,
na końcu trzeba wyeliminować hazard

Testowanie automatu

- Sprawdzenie wszystkich przejść z tabeli
- Sprawdzenie czy wykonuje założone funkcje

Do tego przydaje się zakodowana tabela przejść i wyjść
z zaznaczonymi stanami stabilnymi

Szpilki na wyjściu automatu
Nie są w porządku kiedy wyjście steruje innym automatem asynchronicznym
(wrażliwym na szpilki)

Są ok kiedy wyjście steruje układem kombinacyjnym albo synchronicznym
(niewrażliwym na szpilki)