

POLITECHNIKA POZNAŃSKA

INSTYTUT AUTOMATYKI, ROBOTYKI I INŻYNIERII INFORMATYCZNEJ

ZAKŁAD STEROWANIA I ELEKTRONIKI PRZEMYSŁOWEJ



UKŁADY SEKWENCYJNE SYNCHRONICZNE

TECHNIKA CYFROWA I MIKROPROCESOROWA

MATERIAŁY DO ZAJĘĆ LABORATORYJNYCH

DR INŻ. DOMINIK ŁUCZAK

DOMINIK.LUCZAK@PUT.POZNAN.PL

I. CEL

WIEDZY

Celem zajęć jest zapoznanie z:

- układem sekwencyjnym synchronicznym,
- podstawowymi typami przerzutników,
- implementacją układu sekwencyjnego za pomocą układów sprzętowych (małej, średniej i dużej skali integracji) oraz języka wysokiego poziomu.

UMIEJĘTNOŚCI

Celem zajęć jest nabycie umiejętności programistycznych pozwalających na programową realizację układu sekwencyjnego:

- implementacją układu sekwencyjnego synchronicznego za pomocą układów sprzętowych (małej, średniej i dużej skali integracji) oraz języka wysokiego poziomu,
- wykorzystanie przerzutników różnych typów do realizacji automatu.

KOMPETENCJI SPOŁECZNYCH

Celem zajęć jest kształtowanie właściwych postaw:

- tworzenie kodu programu w sposób czytelny,
- stosowanie dyrektyw preprocesora oraz makroinstrukcji,
- prawidłowa obsługa programowa przerwań mikrokontrolera,
- stosowanie stosownych komentarzy,
- weryfikacja poprawności działania systemu.

II. POLECENIA KOŃCOWE

Dokończ zadania nie zrealizowane w trakcie zajęć. Przygotuj raport z przeprowadzonych zajęć. Jeden raport może być przygotowany przez maksymalnie 2 osoby. W projekcie zamieść:

1. Opis rozwiązania problemu oraz jego implementację.
2. Rysunki lub fotografie ilustrujące poprawne działanie programu.
3. Opis działania własnych programów. Udokumentuj przeprowadzenie testów funkcjonalnych z wykorzystaniem symulatora.
4. Ważne instrukcje programu należy zaopatrzyć stosownym komentarzem. Wszystkie przygotowane funkcje oraz zmienne globalne należy zaopatrzyć komentarzem zgodnym z składnią generatora dokumentacji (*Doxygen*).

Jako załącznik dołącz spakowane kody programów. Dla każdego zadania przygotuj osobny projekt. Raport należy przygotować i przekazać do kolejnego spotkania. Raporty przekazywane później nie będą oceniane.

Na ocenę raportu będą miały wpływ następujące elementy (łącznie 5 punktów):

1. spełnienie wymogów redakcyjnych (1p),
2. wykonanie, udokumentowanie oraz opis wykonanych zadań (2p),
3. zastosowanie prawidłowego warsztatu programistycznego (2p).

III. SCENARIUSZ DO ZAJĘĆ

a) ŚRODKI DYDAKTYCZNE

Sprzętowe: ▪ komputer.

Programowe: ▪ symulatora układów logicznych (Circuit Maker).

b) PRZEBIEG ZAJĘĆ

1. Realizacja automatu z wykorzystaniem układów dużej skali integracji.
 - a. Jako automat należy przyjąć automat $A = (S, \Sigma, \delta)$ wygenerowany za pomocą algorytmu 1.
 - b. Zaprojektować i zbudować układ sekwencyjny synchroniczny dużej skali integracji realizujący automat A za pomocą: rejestru równoległego (układ scalony 175) oraz pamięci RAM (układ scalony 89 – użyj pamięci PROM32 w CircuitMaker zamiast 89).
 - i. Stany automatu należy zakodować w naturalnym kodzie binarnym.
 - ii. Pamięć zapisz klikając PPM na element i wybierając Edit PROM/RAM.
 - iii. Układ realizuje część UK1 – Pamięć bez wyjść.
 - c. Sprawdź działanie automatu.
2. Realizacja automatu z wykorzystaniem układów średniej skali integracji.
 - a. Jako automat należy przyjąć automat $A = (S, \Sigma, \delta)$ wygenerowany za pomocą algorytmu 1.
 - b. Zaprojektować i zbudować minimalny układ sekwencyjny synchroniczny z wykorzystaniem elementów malej lub średniej skali integracji realizujący automat $A = (S, \Sigma, \delta)$ za pomocą:
 - i. przerzutników D (175),
 - ii. wyznaczyć postać minimalną dla UK1,
 - iii. realizacja UK1 dowolnym sposobem z wykluczeniem układów dużej skali integracji (RAM),
 - iv. brak jest realizacji UK2 – wyjść.
 - c. Sprawdź działanie automatu.

IV. PRZYGOTOWANIE DO ZAJĘĆ

a) ZAPOZNANIE Z PRZEPISAMI BHP

Wszystkie informacje dotyczące instrukcji BHP laboratorium są zamieszczone w sali laboratoryjnej oraz u prowadzącego zajęcia. Wszystkie nieścisłości należy wyjaśnić z prowadzącym laboratorium. Wymagane jest zaznajomienie i zastosowanie do regulaminu.

Na zajęcia należy przyjść przygotowanym zgodnie z tematem zajęć. Obowiązuje również materiał ze wszystkich odbytych zajęć.

b) ALGORYTM 1. GENEROWANIE AUTOMATU O 6-CIU STANACH I JEDNYM WEJŚCIU BINARNYM

Wejście: nazwisko, imię

Wyjście: automat $A = (S, \Sigma, \delta)$, gdzie stany $S = \{0, 1, \dots, 5\}$, sygnał wejściowy $\Sigma = \{\sigma_0 = 0, \sigma_1 = 1\}$ oraz δ jest funkcją przejść.

Metoda:

1. Tworzenie tabeli pomocniczej (6 wierszy / 12 kolumn)
 - a. W wierszu 1 wpisz 12 kolejnych liter ciągu: <nazwisko><imię><nazwisko><imię>... rozpoczynając od litery pierwszej.
 - b. W wierszu 2 wpisz ciąg 12 liczb: 0, 1, ..., 11.
 - c. Posortuj 12 dwuwierszowych kolumn (litera, liczba) zawartych w obszarze tabeli obejmującym wiersze 1–2 w kolejności alfabetycznej liter, w przypadku jednakowych liter – w kolejności rosnących liczb, i wpisz posortowane kolumny w obszarze tabeli obejmującym wiersze 3–4.
 - d. W każdej kolumnie podziel przez liczbę 6 zawartość wiersza 4 i wpisz:
 - i. resztę z dzielenia w wierszu 5,
 - ii. część całkowitą ilorazu w wierszu 6.
2. Tworzenie tabeli przejść automatu (6 stanów, 2 litery wejściowe)
 - a. Dla każdej kolumny tabeli pomocniczej przyjmij: $\delta(s, \sigma) = s'$ gdzie:
 - i. s – oznacza zawartość wiersza 5 (stan automatu),
 - ii. σ – oznacza zawartość wiersza 6 (literę wejściową),
 - iii. s' – oznacza zawartość wiersza 5 w kolumnie następnej (stan następny automatu).
 - b. Należy przyjąć, że kolumną następną względem ostatniej jest kolumna pierwsza.
3. Wartości funkcji przejść wyznaczone w kroku 2.a zapisz w tabeli przejść automatu w sposób następujący: jeżeli $\delta(s, \sigma) = s'$, to wpisz s' w komórce na przecięciu wiersza s i kolumny σ .

Przykład

Wyznaczenie automatu dla imienia i nazwiska Adam Mickiewicz.

Na początku litery ciągu: <nazwisko><imię><nazwisko><imię>... tworzą wiersz 1 (12 kolejnych liter rozpoczynając od litery pierwszej) w Tab. 1. Następnie należy posortować kolumny uzyskując wiersze 3 oraz 4. Wiersz 5 zawiera ciąg stanów automatu, a wiersz 6 – ciąg liter alfabetu wejściowego uzyskanych na podstawie 1.d. Wiersze 5–6 tworzą razem ciąg par (stan, litera wejściowa) definiujący funkcję przejść automatu. Ciąg ten będzie wykorzystywany jako tzw. ścieżka testująca do weryfikacji działania automatu w minimalnej liczbie kroków.

Tab. 1 Tabela pomocnicza

	1	2	3	4	5	6	7	8	9	10	11	12	
1	M	I	C	K	I	E	W	I	C	Z	A	D	ciąg liczb
2	0	1	2	3	4	5	6	7	8	9	10	11	ciąg liczb
3	A	C	C	D	E	I	I	I	K	M	W	Z	posortowane kolumny
4	10	2	8	11	5	1	4	7	3	0	6	9	
5	4	2	2	5	5	1	4	1	3	0	0	3	reszta z dzielenia
6	1	0	1	1	0	0	0	1	0	0	1	1	część całkowita ilorazu

Dla każdej kolumny tabeli pomocniczej została określona jedna wartość $\delta(s, \sigma) = s'$ funkcji przejść automatu:

kol. 1 $\rightarrow \delta(4, 1) = 2$, kol. 2 $\rightarrow \delta(2, 0) = 2$, kol. 3 $\rightarrow \delta(2, 1) = 5$, kol. 4 $\rightarrow \delta(5, 1) = 5$,
kol. 5 $\rightarrow \delta(5, 0) = 1$, kol. 6 $\rightarrow \delta(1, 0) = 4$, kol. 7 $\rightarrow \delta(4, 0) = 1$, kol. 8 $\rightarrow \delta(1, 1) = 3$,
kol. 9 $\rightarrow \delta(3, 0) = 0$, kol. 10 $\rightarrow \delta(0, 0) = 0$, kol. 11 $\rightarrow \delta(0, 1) = 3$, kol. 12 $\rightarrow \delta(3, 1) = 4$.

Dla zilustrowania sposobu wyznaczania funkcji przejść, została wybrana jej jedna wartość: $\delta(4, 0) = 1$. Wszystkie fragmenty przykładu związane z tą wartością zostały zaznaczone cieniowaniem (kolor pomarańczowy).

W końcowym efekcie uzyskano automat przedstawiony w Tab. 2.

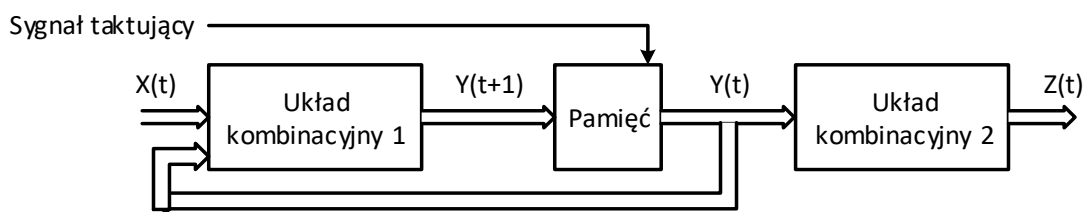
Tab. 2 Tabela przejść automatu

S	$\sigma = 0$	$\sigma = 1$
0	0	3
1	4	3
2	2	5
3	0	4
4	1	2
5	1	5

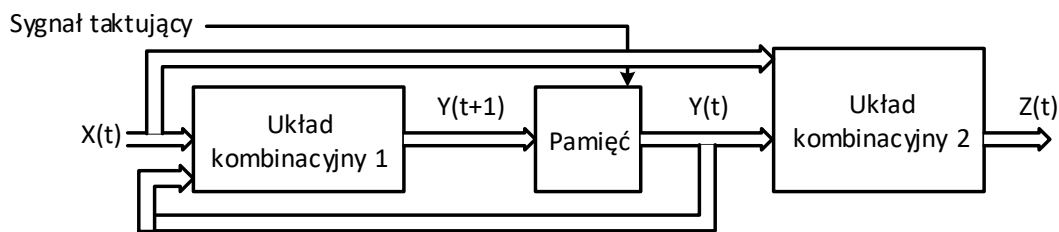
c) WPROWADZENIE

Podobnie, jak miało to miejsce w przypadku układów kombinacyjnych, cyfrowe układy sekwencyjne również możemy wykonywać w technologiach małej, średniej oraz dużej skali integracji. Wykorzystanie dwóch pierwszych struktur następuje przy użyciu przerzutników w roli pamięci stanów i zastosowaniu bramek logicznych do realizacji ich funkcji logicznych zmieniających stan (UK1) oraz realizacji wyjść (UK2). Układ sekwencyjny to układ logiczny składający się z $X=\{x_1, x_2, \dots, x_i\}$ wejść, $Y=\{y_1, y_2, \dots, y_k\}$ sygnałów wewnętrznych oraz $Z=\{z_1, z_2, \dots, z_j\}$ wyjść określonych wzorem (1). Układ sekwencyjny, którego stan wyjść nie zależy bezpośrednio od stanu wejść nazywamy automatem Moor'a (Rys. 1), natomiast jeśli stan wyjść zależy bezpośrednio od stanu wejść układ taki nazywamy automatem Mealy'ego (Rys. 2).

$$Z(t) = f(X(t), X(t-1), X(t-2), \dots) \quad (1)$$



Rys. 1 Automat Moor'a

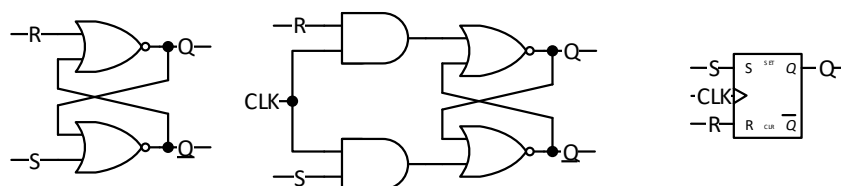


Rys. 2 Automat Mealy'ego

d) RODZAJE UKŁADÓW REALIZUJĄCYCH BLOK PAMIĘCI

PRZERZUTNIK TYPU RS

Przerzutnik typu RS (Rys. 3) w podstawowej wersji jest układem asynchronicznym (bez sygnału CLK). Posiada dwa wejścia: S (ang. Set) - wejście ustawiające oraz R (ang. Reset) - wejście zerujące oraz jedno wyjście Q. Układ można rozbudować o dodatkowe bramki czyniąc z niego układ synchroniczny. Działanie przerzutnika prezentuje tabela przejść (Tab. 3).



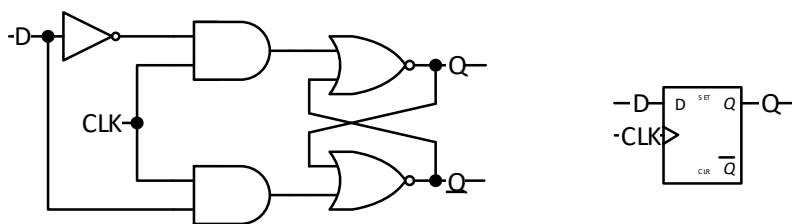
Rys. 3 Przerzutnik typu RS

Tab. 3 Tabela przejść przerzutnika typu RS

S	R	$Q^t \rightarrow Q^{t+1}$
0	0	$Q^t \rightarrow Q^{t+1}$
1	0	$Q^t \rightarrow 1$
0	1	$Q^t \rightarrow 0$
1	1	$Q^t \rightarrow -$

PRZERZUTNIK TYPU D

Przerzutnik typu D (Rys. 4) (ang. data lub delay flip-flop) przepisuje stan wejścia D na wyjście Q . Przepisanie informacji następuje tylko przy odpowiednim stanie wejścia zegarowego CLK . Zgodnie z tabelą przejść (Tab. 4) wyjście przerzutnika po taktowaniu linia CLK zależy wyłącznie od stanu wejścia D .



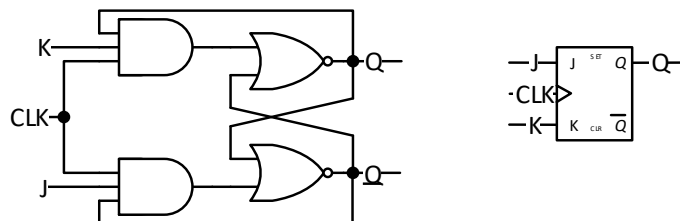
Rys. 4 Przerzutnik typu D

Tab. 4 Tabela przejść przerzutnika typu D

D	$Q^t \rightarrow Q^{t+1}$
0	$0 \rightarrow 0$
0	$1 \rightarrow 0$
1	$0 \rightarrow 1$
1	$1 \rightarrow 1$

PRZERZUTNIK TYPU JK

Przerzutnik typu JK (Rys. 5) to jeden z podstawowych układów synchronicznych. Działanie przerzutnika prezentuje tabela przejść (Tab. 5 i Tab. 6). Stan wyjścia Q^{t+1} zależy od: wejścia J , wejścia K oraz stanu wyjścia z poprzedniego taktowania Q^t .



Rys. 5 Przerzutnik typu JK

Tab. 5 Pełna tabela przejść przerzutnika typu JK

J	K	$Q^t \rightarrow Q^{t+1}$
0	0	$0 \rightarrow 0$
0	1	$0 \rightarrow 0$

1 0	$0 \rightarrow 1$
1 1	$0 \rightarrow 1$
0 0	$1 \rightarrow 1$
0 1	$1 \rightarrow 0$
1 0	$1 \rightarrow 1$
1 1	$1 \rightarrow 0$

Tab. 6 Skrócona tabela przejść przerzutnika typu JK

J K	$Q^t \rightarrow Q^{t+1}$
0 -	$0 \rightarrow 0$
1 -	$0 \rightarrow 1$
- 0	$1 \rightarrow 1$
- 1	$1 \rightarrow 0$

e) REALIZACJA AUTOMATU Z WYKORZYSTANIEM UKŁADÓW DUŻEJ SKALI INTEGRACJI

Realizowany automat Moor'a przedstawiono w Tab. 7. Automat posiada trzy stany zatem minimalne kodowanie w systemie dwójkowym wymaga dwóch pozycji. Przyjęte kodowanie stanów przedstawiono w Tab. 8. Uwzględniając przyjęte kodowanie przepisano automat do postaci Tab. 9 z adresowaniem naturalnym. Wartości słów Q_j komórek pamięci stałej dla poszczególnych adresów A_i przedstawiono w Tab. 10. Adres zbudowany został z: dwóch kodów stanu oraz sygnału wejściowego σ . Wartość komórki pamięci o wskazanym adresie składa się z: wartości wyjścia λ oraz kodów stanu (s_1^{t+1} , s_2^{t+1}) po kolejnym taktowaniu układu. Realizację automatu przedstawia Rys. 6 oraz Rys. 7.

Tab. 7 Realizowany automat

S	$\sigma=0$	$\sigma=1$	λ
1	1	2	0
2	2	3	1
3	3	2	1

Tab. 8 Kodowanie stanów automatu

S	$s_1 s_2$
0	0 0
1	0 1
2	1 0
3	1 1

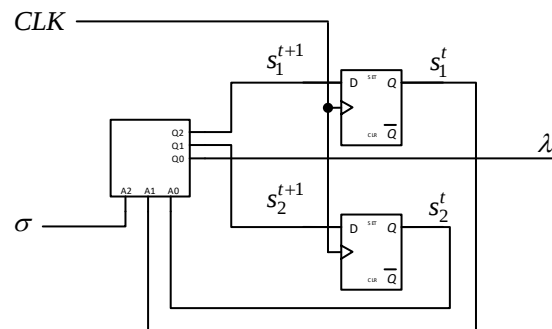
Tab. 9 Realizowany automat z kodowaniem stanów

$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ
0 0	- -	- -	-
0 1	0 1	1 0	0
1 0	1 0	1 1	1
1 1	1 1	1 0	1

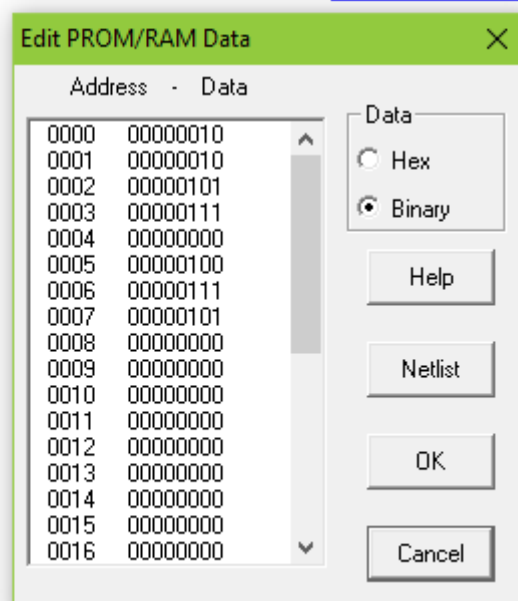
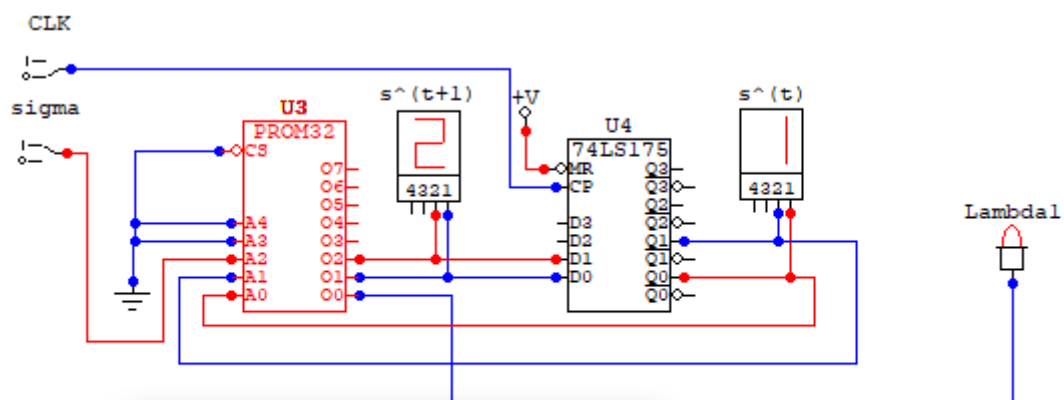
Tab. 10 Zapis w pamięci stałej

Nr	A2 A1 A0	Q2 Q1 Q0
	$\sigma \quad s_1^t \quad s_2^t$	$s_1^{t+1} \quad s_2^{t+1} \quad \lambda$
0	0 0 0	- - -

1	0	0	1	0	1	0
2	0	1	0	1	0	1
3	0	1	1	1	1	1
4	1	0	0	-	-	-
5	1	0	1	1	0	0
6	1	1	0	1	1	1
7	1	1	1	1	0	1



Rys. 6 Realizacja automatu z wykorzystaniem układu dużej skali integracji



Rys. 7 Realizacja automatu z wykorzystaniem układu dużej skali integracji - schemat

f) REALIZACJA AUTOMATU Z WYKORZYSTANIEM UKŁADÓW ŚREDNIEJ SKALI INTEGRACJI

PRZERZUTNIK TYPU D

Realizowany automat Moor'a przedstawiono w Tab. 7. Automat posiada trzy stany zatem minimalne kodowanie w systemie dwójkowym wymaga dwóch pozycji. Przyjęte kodowanie stanów przedstawiono w Tab. 8. Uwzględniając przyjęte kodowanie przepisano automat do postaci Tab. 9 z adresowaniem naturalnym, następnie przekształcono zapis do tablicy Karnaugh Tab. 11 by możliwa była minimalizacja układu. W celu ułatwienia procesu minimalizacji utworzono dwie tablice pomocnicze Tab. 12 oraz Tab. 13, dla których zaznaczono „sklejane jedynki”. W obu przypadkach występują tylko dwa „bloki sklejanych jedynek”. W wyniku minimalizacji uzyskano wyrażenia (2)-(4). Końcowa realizacja sprzętowa analizowanego automatu została przedstawiona na Rys. 8.

$$s_1^{t+1} = D_1 = s_1 + \sigma = \overline{s_1 + \sigma} = \overline{s_1} \cdot \overline{\sigma} \quad (2)$$

$$s_2^{t+1} = D_2 = s_2 \cdot \overline{\sigma} + \overline{s_2} \cdot \sigma = \overline{s_2 \cdot \overline{\sigma} + \overline{s_2} \cdot \sigma} = \overline{s_2 \cdot \overline{\sigma}} \cdot \overline{\overline{s_2} \cdot \sigma} \quad (3)$$

$$\lambda = s_1 \quad (4)$$

Tab. 11 Realizowany automat z kodowaniem stanów - tablica Karnaugh

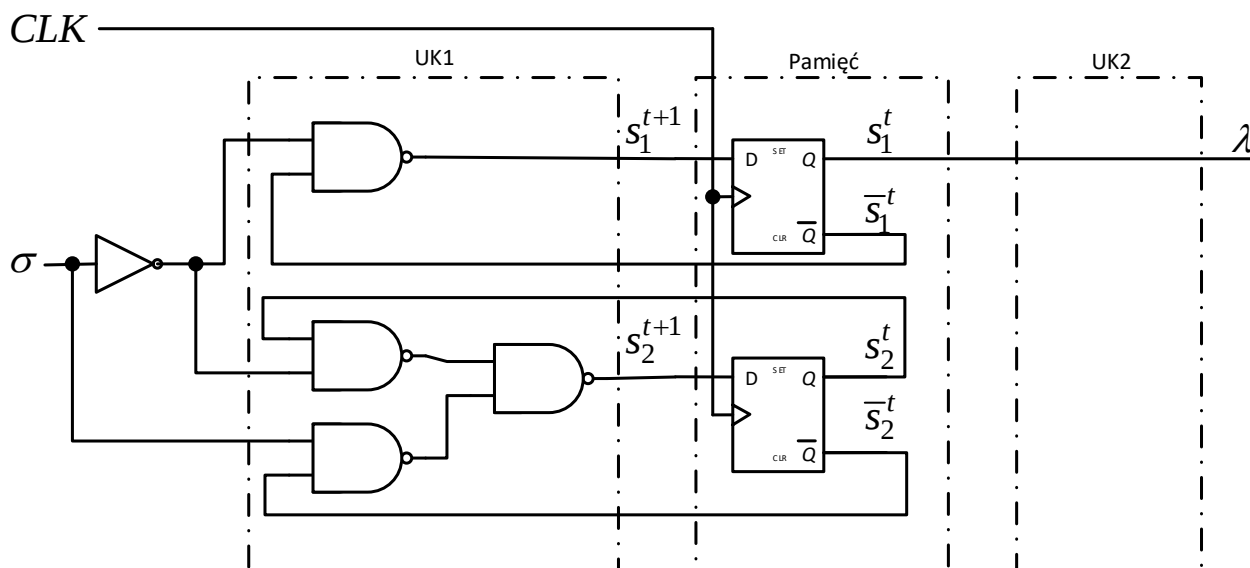
$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ
0 0	-	-	-
0 1	0	1	0
1 1	1	1	1
1 0	1	1	1
	$D_1 D_2$	$D_1 D_2$	

Tab. 12 Pomocnicza tablica Karnaugh - D1

$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ		$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ		$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ	
0 0	-	-	-		0 0	-	-	-		0 0	-	-	-	
0 1	0	1	0		0 1	0	1	0		0 1	0	1	0	
1 1	1	1	1		1 1	1	1	1		1 1	1	1	1	
1 0	1	1	1		1 0	1	1	1		1 0	1	1	1	
	D_1	D_1				D_1	D_1				D_1	D_1		
Tablica pomocnicza D1					Pierwszy fragment					Drugi fragment				

Tab. 13 Pomocnicza tablica Karnaugh - D2

$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ		$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ		$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ	
0 0	-	-	-		0 0	-	-	-		0 0	-	-	-	
0 1	1	0	0		0 1	1	0	0		0 1	1	0	0	
1 1	1	0	1		1 1	1	0	1		1 1	1	0	1	
1 0	0	1	1		1 0	0	1	1		1 0	0	1	1	
	D_2	D_2				D_2	D_2				D_2	D_2		
Tablica pomocnicza D2					Pierwszy fragment					Drugi fragment				



Rys. 8 Realizacja automatu z wykorzystaniem przerzutnika typu D

PRZERZUTNIK TYPU JK

W przypadku realizacji automatu z wykorzystaniem przerzutnika JK algorytm syntezy układu jest podobny. Automat (Tab. 14) jest zapisywany do postaci Tab. 16 z uwzględnieniem kodowania stanów (Tab. 15). Następnie tablica jest przepisana do postaci realizowalnej z wykorzystaniem przerzutnika JK (Tab. 17) mając na uwadze tablice przejść przerzutnika JK (Tab. 6). Minimalizację dokonuje się dla każdego z wejść przerzutnika. Dla omawianego przypadku uzyskano postać $J_1 = \sigma$ oraz $K_1 = \sigma$. Realizację sprzętową przedstawiono na Rys. 9.

Tab. 14 Realizowany automat

S	$\sigma=0$	$\sigma=1$	λ
1	1	2	0
2	2	1	1

Tab. 15 Kodowanie stanu automatu

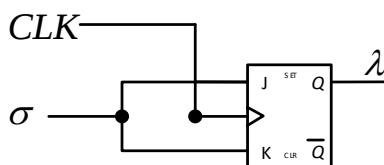
S	s_1
1	0
2	1

Tab. 16 Realizowany automat z kodowaniem stanów

s_1	$\sigma=0$	$\sigma=1$	λ
0	0	1	0
1	1	0	1

Tab. 17 Realizowany automat z kodowaniem stanów dla przerzutnika JK

s_1	$\sigma=0$	$\sigma=1$		s_1	$\sigma=0$	$\sigma=1$		s_1	$\sigma=0$	$\sigma=1$	
0	0-	1-		0	0	1		0	-	-	
1	-0	-1		1	-	-		1	0	1	
	JK_1	JK_1			J_1	J_1			K_1	K_1	
Cała tabela				Fragment tabeli odpowiadający wejściu J				Fragment tabeli odpowiadający wejściu K			

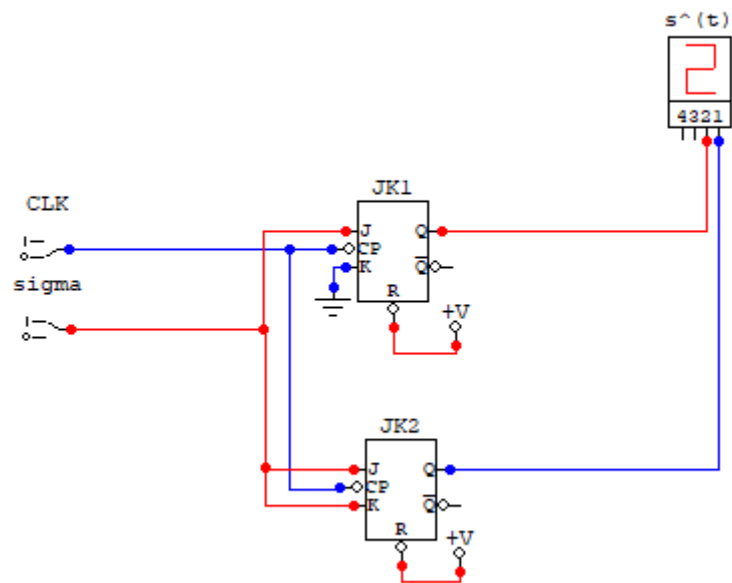


Rys. 9 Realizacja automatu z wykorzystaniem przerzutnika typu JK

W przypadku realizacji automatu Moor'a przedstawionego w Tab. 7 należy przygotować tablicę Karnaugh zakodowanego automatu (Tab. 18) z uwzględnieniem tabeli przejść przerzutnika JK (Tab. 6). Do realizacji automatu wymagane jest zastosowanie dwóch przerzutników JK w celu realizacji kodu stanu. Na żółto zaznaczono kodu adresu brany pod uwagę przy wyznaczaniu wartości wejściowych przerzutnika. Przykładowo z stanu 3 ($s_1 s_2 = 1 1$) pod wpływem $\sigma = 1$ automat przechodzi (po taktowaniu) do stanu 2 ($s_1 s_2 = 1 0$). W omawianym przypadku wartość kodu stanu zmienia się z $s_1^t = 1$ na $s_1^{t+1} = 1$, natomiast dla drugiego kodu z $s_2^t = 1$ na $s_2^{t+1} = 0$, zatem aby nastąpiła taka zmiana wykorzystując przerzutnik typu JK na wejście pierwszego przerzutnika JK_1 należy podać $J_1 = -$ oraz $K_1 = 0$, natomiast na wejście drugiego przerzutnika JK_2 należy podać odpowiednio $J_2 = -$ oraz $K_2 = 1$. Przykład zaznaczono kolorem szarym. W celu ułatwienia procesu minimalizacji zaznaczono „sklejane jedynki” bądź „sklejane zera”. W wyniku minimalizacji uzyskano następujące wyrażenia: $J_1 = \sigma$, $K_1 = 0$, $J_2 = \sigma$ oraz $K_2 = \sigma$. Końcowa realizacja sprzętowa analizowanego automatu została przedstawiona na Rys. 10.

Tab. 18 Realizowany automat z kodowaniem stanów dla przerzutnika JK

$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ	$s_1 s_2$	$\sigma=0$	$\sigma=1$	$s_1 s_2$	$\sigma=0$	$\sigma=1$	$s_1 s_2$	$\sigma=0$	$\sigma=1$
0 0	-	-	-	0 0	--	--	0 0	-	-	0 0	-	-
0 1	0	1	0	0 1	0 -	1 -	0 1	0	1	0 1	-	-
1 1	1	1	1	1 1	- 0	- 0	1 1	-	-	1 1	0	0
1 0	1	1	1	1 0	- 0	- 0	1 0	-	-	1 0	0	0
	Q-JK ₁	Q-JK ₁			$J_1 K_1$	$J_1 K_1$		J_1	J_1		K_1	K_1
$s_1 s_2$	$\sigma=0$	$\sigma=1$	λ	$s_1 s_2$	$\sigma=0$	$\sigma=1$	$s_1 s_2$	$\sigma=0$	$\sigma=1$	$s_1 s_2$	$\sigma=0$	$\sigma=1$
0 0	-	-	-	0 0	--	--	0 0	-	-	0 0	-	-
0 1	1	0	0	0 1	- 0	- 1	0 1	-	-	0 1	0	1
1 1	1	0	1	1 1	- 0	- 1	1 1	-	-	1 1	0	1
1 0	0	1	1	1 0	0 -	1 -	1 0	0	1	1 0	-	-
	Q-JK ₂	Q-JK ₂			$J_2 K_2$	$J_2 K_2$		J_2	J_2		K_2	K_2
Wyjścia przerzutnika				Kodowanie wejść przerzutnika			Wejście J przerzutnika			Wejście K przerzutnika		



Rys. 10 Realizacja automatu z wykorzystaniem przerzutnika JK