Opracował:
G. Wasilewski
26.XI.2007

Katedra Automatyki i Biomechaniki P.Ł.

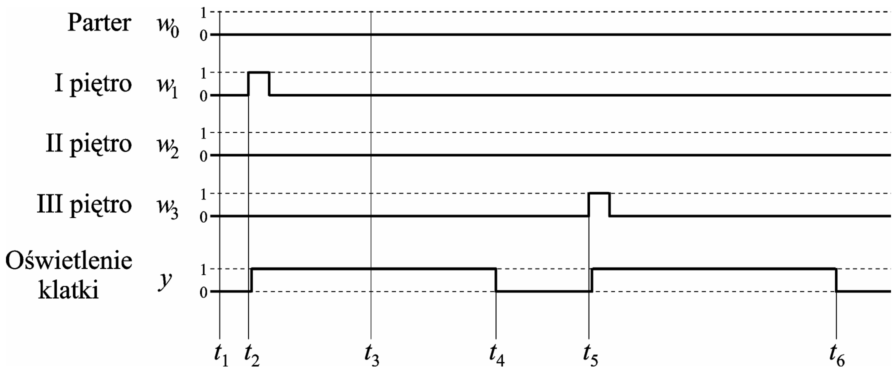
LABORATORIUM PODSTAW AUTOMATYKI

Ćwiczenie G: ***Układy logiczne sekwencyjne***

Cel ćwiczenia: Zapoznanie się z zasadą działania przerzutników typu RS, JK i D. Zaprojektowanie układu sekwencyjnego metodą tabel kolejności łączeń, zmontowanie i sprawdzenie poprawności działania.

Wprowadzenie

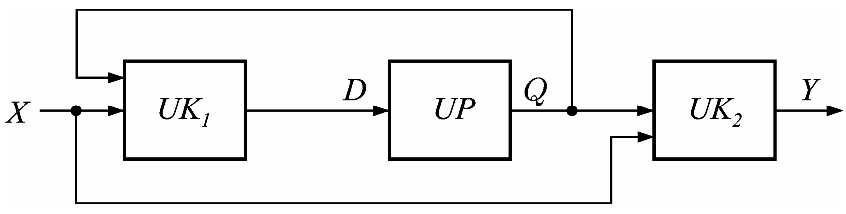
W przypadku układów kombinacyjnych (układów przełączających jednotaktowych) mieliśmy do czynienia z sytuacją, gdy dla określenia stanu dowolnej wielkości wyjściowej wystarczyła jedynie znajomość stanu sygnałów wejściowych układu w danej chwili. W układach sekwencyjnych (układach przełączających wielotaktowych) stan wielkości wyjściowej zależy zarówno od aktualnego stanu wejść, jak i od tego, co działo się wcześniej, a więc od określonej *sekwencji* (kolejności) zmian sygnałów wejściowych układu. Oznacza to, że identyczna kombinacja sygnałów wejściowych może raz oznaczać załączenie danego urządzenia, a innym razem jego wyłączenie, zależnie od tego, w którym *takcie* układ się znajduje. Według definicji podanej w pracy [1] można napisać, że dany układ logiczny jest układem sekwencyjnym, jeżeli ma przynajmniej jeden taki stan wejść, któremu odpowiadają różne stany wyjść. Jako prosty przykład można podać układ automatu schodowego, zapalającego na pewien czas światło na klatce schodowej po krótkim wciśnięciu dowolnego z włączników, umieszczonych na każdym piętrze. Na rys. 1 pokazano możliwe przebiegi sygnałów.



Rys. 1. Przebiegi sygnałów automatu schodowego

Po wciśnięciu włącznika w_1 w chwili t_2 lub włącznika w_3 w chwili t_5 oświetlenie zostaje załączone na zadany czas $t_2 \div t_4$ lub $t_5 \div t_6$. Z rysunku widać, że w chwilach t_1 i t_3 poziomy logiczne czterech sygnałów wejściowych $w_0 \div w_3$ są identyczne (0000, czyli stan wyłączony), jednak sygnał wyjściowy y raz przyjmuje wartość 0, raz 1. Wynika to stąd, że układ „pamięta” fakt załączenia któregoś z włączników. Istnienie *elementów pamięci* to cecha charakterystyczna każdego układu sekwencyjnego.

Ogólny schemat blokowy układu logicznego sekwencyjnego przedstawia rys. 2.



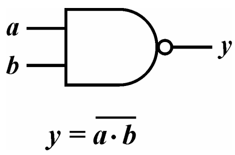
Rys. 2. Ogólny schemat blokowy układu sekwencyjnego (wg [4]).
Oznaczenia:
X - sygnały wejściowe;
Y - sygnały wyjściowe;
D, Q - sygnały wewnętrzne;
UK_1 - układ kombinacyjny wejściowy;
UK_2 - układ kombinacyjny wyjściowy;
UP - układ pamięci.

Podanie sygnałów wewnętrznych *Q* na wejście układu pełni rolę sprzężenia zwrotnego. A więc stan wyjścia *Y* układu w danym takcie zależy nie tylko od kombinacji sygnałów wejściowych *X*, ale również od sygnałów wewnętrznych Q, które z kolei zależą od sygnałów powstałych w takcie poprzedzającym.

Jeżeli wszystkie elementy układu pamięci zmieniają swój stan w ściśle określonej chwili, wyznaczonej dodatkowym sygnałem synchronizującym (zegarowym), to taki układ nazywamy *układem sekwencyjnym synchronicznym*. W przypadku braku takiego sygnału synchronizującego mamy do czynienia z *układem sekwencyjnym asynchronicznym*.

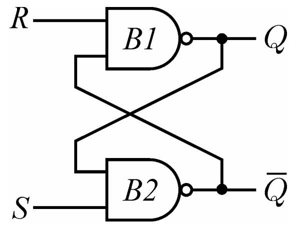
Przerzutniki

Przerzutnik *RS*. W układach logicznych rolę pamięci spełniają elementy zwane *przerzutnikami*. Najprostszy układ logiczny z pamięcią można zrealizować wykorzystując dwa elementy (bramki) NOR lub NAND. W tab. 1 przypomniana została tabela stanów bramki NAND, a na rys. 3 przedstawiono układ logiczny, zbudowany z dwóch bramek NAND, zwany *przerzutnikiem RS*. Obydwie bramki są jak widać wzajemnie sprzężone ze sobą.

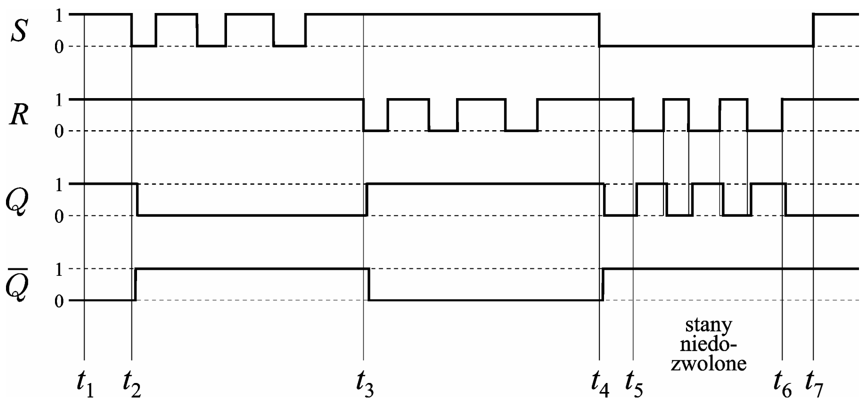


Tab. 1. Tabela stanów bramki NAND

Stany wejściowe		Stan wyjścia
a	*b*	*y*
0	0	1
0	1	1
1	0	1
1	1	0



Rys. 3. Przerzutnik *RS* z dwóch bramek NAND



Rys. 4. Przebiegi sygnałów przerzutnika *RS*

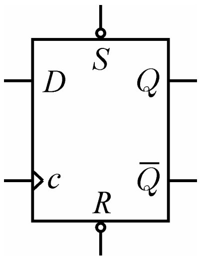
Tab. 2. Tabela stanów przerzutnika *RS*

Stany wejściowe		Stan wyjść	
S	*R*	*Q*	$\overline{Q}$
1	1	bez zmian	
1	0	1	0
0	1	0	1
0	0	1	1
stan niedozwolony			

Rozpatrzmy zachowanie się bramek *B1* i *B2* połączonych w ten sposób. Załóżmy, że w chwili początkowej t_1 wejścia i wyjścia znajdują się w stanie pokazanym na rys. 4. Jak wiadomo bramka NAND ma na wyjściu stan niski, czyli 0, tylko w tym jednym przypadku, gdy na obu jej wejściach jest stan wysoki, czyli 1. W takiej właśnie sytuacji znajduje się bramka *B2*, gdyż obydwa jej wejścia, którymi są sygnały *S* oraz Q, znajdują się w stanie 1. Zatem wyjście bramki *B2,* czyli $\overline{Q}$, przyjmuje stan 0. Ponieważ $\overline{Q}$ jest jednocześnie wejściem bramki *B1*, więc podtrzymuje na jej wyjściu Q stan 1, co jest zgodne z początkowym założeniem. Cały układ znajduje się zatem w stanie stabilnym. Stan ten utrzymuje się do chwili t_2, gdy na wejściu *S* pojawi się stan niski. W tym momencie na wyjściu $\overline{Q}$ bramki *B2* zostaje wymuszony stan 1, a to z kolei powoduje, że na obu wejściach bramki *B1* jest stan 1, więc jej wyjście Q przechodzi do stanu 0. Stan 0 na wyjściu Q podtrzymuje wyjście bramki *B2*, czyli $\overline{Q}$ w stanie wysokim, tak że mimo powrotu wejścia *S* do stanu wysokiego układ utrzymuje się w nowym, stabilnym stanie. Ten nowy stan jest niczym innym, jak zapamiętaniem chwilowego, niskiego stanu na wejściu *S*. Należy tu zauważyć, że dalsze zmiany stanu wejścia *S* bramki *B2*, zachodzące między t_2 a t_3, nie mogą już spowodować zmian wyjścia $\overline{Q}$, gdyż na drugim wejściu tej bramki jest ciągle stan 0. Powrót do pierwotnego stanu może nastąpić dopiero wówczas, gdy na wejściu *R* pojawi się stan niski. Na rys. 4 następuje to w chwili t_3, a przejście (lub „przerzucenie”) przerzutnika do nowego stanu odbywa się analogicznie, jak to opisano dla zmian na wejściu *S*. Aby stan przerzutnika był stabilny, a wyjścia Q i $\overline{Q}$ były zawsze komplementarne (jeżeli jedno 0, to drugie 1), układ współpracujący z tym przerzutnikiem nie powinien podawać na wejścia *R* i *S* jednocześnie stanu niskiego. Jest to warunek poprawnego działania tego przerzutnika, gdyż jak widać z rysunku 4, jeżeli np. między t_4 a t_7 wejście *S* utrzymuje się w stanie niskim, i w tym samym czasie wystąpią niskie stany na wejściu *R* (między t_5 a t_6), to stan taki nie jest stabilny, gdyż każda zmiana *R* wywołuje zmianę wyjścia Q, a jednocześnie w chwilach, gdy R=S=0 obydwa wyjścia Q i $\overline{Q}$ są w stanie wysokim, co narusza warunek ich komplementarności. Stan R=S=0 określa się jako niedozwolony lub zabroniony. Reasumując, działanie przerzutnika *RS* można przedstawić w postaci tabeli stanów (analogicznie jak dla bramki), pokazanej w tab. 2.

Oznaczenie przerzutnika i jego wejść powstało od angielskich słów S=Set (ustawić) i R=Reset (kasować). Jak już wcześniej wspomniano, podobny przerzutnik można zbudować z bramek NOR. Zasada działania jest analogiczna z tą tylko różnicą, że stanem niedozwolonym będzie R=S=1, a stanem aktywnym wejść *R* lub *S* będzie stan wysoki.

Przerzutnik *D*. Mimo małych możliwości funkcjonalnych proste przerzutniki *RS* są dość często wykorzystywane w praktyce. Znacznie częściej jednak stosowane są przerzutniki bardziej rozbudowane i uniwersalne, produkowane w postaci gotowych układów scalonych. Jednym z nich jest przerzutnik typu *D*. Symbol graficzny i tabela stanów przedstawiona jest na rys. 5 i w tab. 3 (szczegółowy schemat logiczny i elektroniczny przerzutników w technologii TTL można znaleźć w pracy [2]).



Rys. 5. Symbol graficzny przerzutnika typu *D*.
Oznaczenia:
D - wejście informacyjne;
c - wejście zegarowe;
S, *R* - wejścia ustawiające;
Q i $\overline{Q}$ - komplementarne wyjścia.

Tab. 3. Tabela stanów przerzutnika typu *D*

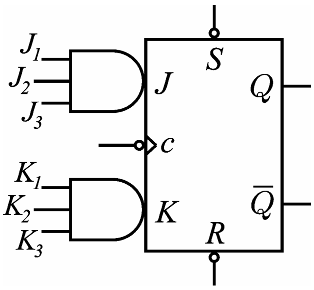
L.p.	Stany wejściowe				Stan wyjść	
	S	*R*	*D*	*c*	Q	$\overline{Q}$
1	1	1	X	0	bez zmian	
2	1	1	1	↑	1	0
3	1	1	0	↑	0	1
4	1	0	X	X	1	0
5	0	1	X	X	0	1
6	0	0	X	X	1	1
	stan niedozwolony					

X - oznacza stan dowolny

Zasada działania tego przerzutnika polega na tym, że jeżeli wejścia *R* i *S* są w stanie wysokim, to w momencie dodatniego zbocza wejścia zegarowego *c* informacja, czyli stan wejścia *D*, zostaje przeniesiony (lub „przepisany”) na wyjście Q. Przerzutnik zapamiętuje więc w sposób trwały stan wejścia aż do momentu następnego dodatniego zbocza zegarowego. Pomiędzy kolejnymi zboczami stan wyjść jest stabilny i żadne zmiany na wejściu *D* nie oddziałują na wyjście Q. Wyjście $\overline{Q}$ przyjmuje przy tym stan przeciwny (komplementarny) względem Q.

W przypadku, gdy jedno z wejść ustawiających *R* lub *S* znajdzie się w stanie niskim, to wyjścia Q i $\overline{Q}$, bez względu na stan wejść *D* i *c*, przyjmą stan wg 4 i 5 wiersza tabeli. Można zauważyć, że wejścia *R* i *S* oraz wyjścia *Q* i $\overline{Q}$ zachowują się w tym przypadku identycznie jak w prostym przerzutniku *RS*, zatem przerzutnik *D* może być wykorzystany również jako zwykły przerzutnik *RS* (wejścia *D* i *c* są wtedy po prostu nie używane). Podobnie też, jak to było przy przerzutniku *RS*, stan *R*=*S*=0 jest stanem niedozwolonym, gdyż obydwa wyjścia znajdują się wtedy w niestabilnym, wysokim stanie, a niewłaściwe przejście *R* i *S* do stanu wysokiego może spowodować, że wyjścia ustawią się w przypadkowym, nieprzewidywalnym stanie. Na symbolu graficznym przerzutnika przy wejściu zegarowym *c* rysuje się zwykle trójkąt (rys. 5), który oznacza, że aktywnym stanem tego wejścia jest dodatnie zbocze sygnału, czyli moment przejścia z 0 do 1. Natomiast kółeczka przy wejściach *R* i *S* oznaczają, że aktywnym stanem jest stan 0.

Przerzutnik *JK*. Bardziej rozbudowanym i uniwersalnym, z większą liczbą wejść jest przerzutnik typu *JK*. Symbol graficzny i tabelę stanów przedstawiono na rys. 6 i w tab. 4.



Rys. 6. Symbol graficzny przerzutnika typu *JK*.
Oznaczenia:
J1, *J2*, *J3* - wejścia informacyjne (*J=J1·J2·J3*);
K1, *K2*, *K3* - wejścia informacyjne (*K=K1·K2·K3)*;
c - wejście zegarowe;
S, *R* - wejścia ustawiające;
Q i $\overline{Q}$ - komplementarne wyjścia.

Tab. 4. Tabela stanów przerzutnika typu *JK*

L.p.	Stany wejściowe					Stan wyjść	
	S	*R*	*J*	*K*	*c*	*Q*	$\overline{Q}$
1	1	1	0	0	⎍	bez zmian	
2	1	1	1	0	⎍	1	0
3	1	1	0	1	⎍	0	1
4	1	1	1	1	⎍	zmiana stanu wyjść	
5	1	0	X	X	X	1	0
6	0	1	X	X	X	0	1
7	0	0	X	X	X	1	1
stan niedozwolony							

X - oznacza stan dowolny

W przerzutniku jest 6 wejść informacyjnych. Sygnał wewnętrzny *J* przerzutnika jest wyjściem bramki AND, co oznacza, że *J*=1 tylko wtedy, gdy na wszystkich wejściach informacyjnych J_1, J_2 i J_3 jest stan 1. Analogicznie *K*=1 tylko wtedy, gdy K_1=K_2=K_3=1. Zasadnicze działanie przerzutnika odbywa się przy wysokim stanie wejść ustawiających *S* i *R*. Z pierwszych czterech wierszy tab. 4, widać, że:

- dla *J*=*K*=0 impuls na wejściu zegarowym *c* nie powoduje żadnych zmian wyjść przerzutnika;
- jeżeli tylko *J* lub tylko *K* jest równe 1, to następuje „przepisanie” stanu *J* na wyjście *Q*, a stanu *K* na wyjście $\overline{Q}$;
- dla *J*=*K*=1 każde ujemne zbocze impulsu zegarowego powoduje zmianę stanu wyjść na odwrotną. Jest to przypadek tzw. dwójki liczącej.

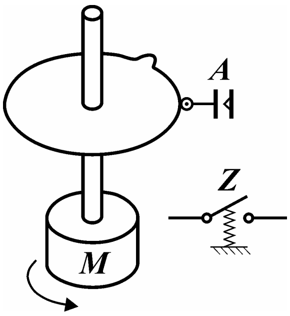
Należy tu dodać, że dla przerzutnika *JK* zmiany stanu wyjść następują w chwili opadania zbocza zegarowego, odwrotnie niż to było w przypadku przerzutnika *D*. Fakt ten jest uwidoczniony na rys. 6 w postaci dodatkowego kółeczka na wejściu zegarowym, które w połączeniu z trójkątem oznacza, że aktywne jest ujemne zbocze sygnału, czyli przejście z 1 do 0. Ponadto warunkiem poprawnego działania jest to, aby podczas wysokiego stanu impulsu zegarowego nie następowały żadne zmiany wejść informacyjnych (stanowi to pewną wadę tego przerzutnika). Wejścia *R* i *S* pełnią identyczną funkcję, jak to opisano dla przerzutnika *D*; podobnie też stan *R*=*S*=0 jest niedozwolony.

W układach logicznych sekwencyjnych, po włączeniu napięcia zasilania wyjścia przerzutników mogą ustawić się w przypadkowy sposób. Można to zaobserwować włączając zasilanie tablicy dydaktycznej, wykorzystywanej podczas ćwiczenia. Dlatego też w praktycznych wykonaniach układów generowany jest zwykle specjalny sygnał, który przez krótki czas po włączeniu zasilania ustawia układ pamięci w żądanym położeniu wyjściowym.

Przykład rozwiązania układu asynchronicznego metodą tabel kolejności łączeń

Jedną z metod opisu i projektowania układów asynchronicznych jest metoda *tabel kolejności łączeń*. Polega ona na zapisie warunków działania układu w rozbiciu na poszczególne takty, przy czym w sąsiednich taktach zmianie ulega tylko jeden z sygnałów układu. Metoda ta zostanie przedstawiona na prostym przykładzie zaczerpniętym z pracy [3].

Przykład. Tarczę z występem obraca silnik *M*, jak pokazano na rys. 7. Należy zaprojektować taki układ logiczny sterujący załączaniem silnika, aby po krótkotrwałym wciśnięciu przycisku *Z* tarcza wykonała tylko jeden obrót.



Rys. 7. Schemat układu silnika z tarczą

Tab. 5. Tabela kolejności łączeń

	Numer taktu →	0	1	2	3	4	5	6
$2^0=1$	przycisk *Z*	0	1	1	0	0	0	0
$2^1=2$	zestyk *A*	0	0	0	0	1	0	0
$2^2=4$	silnik *M*	0−	0+	1+	1+	1+	1−	0−
	Stopień łączenia	0	1	5	4	6	4	0

Z treści zadania i rysunku wynika, że po wciśnięciu przycisku silnik powinien zostać włączony, a następnie zatrzymać się, gdy występ na tarczy minie zestyki *A*. Zaprojektowany układ nie może być tylko układem kombinacyjnym, gdyż przy rozwartych zestykach *A* i *Z* silnik może się obracać lub nie, w zależności od tego, czy nastąpiło poprzednio załączenie tych zestyków; przypadki te układ musi więc zapamiętać. Wprowadzimy oznaczenie, że stan logiczny 1 oznacza załączenie zestyków *A* i *Z* oraz pracę silnika, natomiast stan 0 oznacza rozłączenie zestyków i unieruchomienie silnika. W rozważanym układzie sygnały *A* i *Z* są *sygnałami wejściowymi*, natomiast *M* jest *sygnałem wyjściowym*, sterującym bezpośrednio silnikiem. W tabeli 5 pokazano tabelę kolejności łączeń, zawierającą stany wejść i wyjść wszystkich elementów układu. Takty 0÷6 przedstawiają jeden *cykl pracy* układu. W dolnym wierszu tabeli wpisano tzw. *stopień łączenia*, który jest po prostu dziesiętnym odpowiednikiem liczby dwójkowej, utworzonej przez stany sygnałów *M*, *A*, *Z* (kolejność nie ma tu istotnego znaczenia, równie dobrze można by przyjąć odwrotną). Np. w takcie 4 stany sygnałów *M*, *A*, *Z* możemy podać jako liczbę dwójkową 110, co odpowiada dziesiętnej liczbie 6, natomiast w taktach 3 i 5 stany sygnałów *M*, *A*, *Z* są 100, co odpowiada liczbie dziesiętnej 4. Stopień łączenia wpisuje się w tych tabelach głównie po to, aby łatwiej zauważyć, czy występują identyczne stany sygnałów. Będzie to potrzebne do określenia, czy utworzona tabela jest rozwiązalna.

W takcie początkowym 0 wszystkie elementy są wyłączone. Możemy zapisać, że [*M A Z*] = [0 0 0]. Kolejne takty to:

Takt 1 - [*M A Z*] = [0 0 1] - przycisk *Z* zostaje włączony,;

Takt 2 - [*M A Z*] = [1 0 1] - następuje uruchomienie silnika M;

Takt 3 - [*M A Z*] = [1 0 0] - przycisk *Z* zostaje zwolniony;

Takt 4 - [*M A Z*] = [1 1 0] - występ na tarczy załącza zestyki *A*;

Takt 5 - [*M A Z*] = [1 0 0] - występ mija zestyki *A* i rozłącza je;

Takt 6 - [*M A Z*] = [0 0 0] - silnik M zostaje zatrzymany, tarcza przyjmuje położenie początkowe.

Aby stwierdzić, czy otrzymana tabela kolejności łączeń jest rozwiązalna, należy w każdym takcie zaznaczyć tzw. *funkcję wzbudzenia* dla istniejących w układzie elementów wyjściowych i pośredniczących (w tab. 5 dotyczy to tylko elementu *M*). Zasada określania tej funkcji jest następująca:

funkcja wzbudzenia jest dodatnia ("**+**"), jeżeli w następnym takcie element jest załączony (stan 1);

funkcja wzbudzenia jest ujemna ("**−**"), jeżeli w następnym takcie element jest wyłączony (stan 0).

Zgodnie z tą zasadą w takcie nr 0 jest "**−**", w takcie nr 1 jest "**+**" (gdyż w następnym *M*=1), w taktach 2, 3 4 również "**+**", w takcie nr 5 "**−**" (gdyż w następnym *M*=0) i w takcie 6 "**−**" (takt ten odpowiada taktowi początkowemu, więc musi być identyczny).

Kryterium rozwiązalności tabeli jest następujące:

> Jeżeli w tabeli nie ma *sprzecznych taktów*, tzn. takich, dla których stopień łączenia jest jednakowy, a funkcja wzbudzeń różna, to taka tabela jest rozwiązalna. Można wtedy znaleźć równania logiczne łączące elementy tabeli i zbudować układ realizujący zadane działanie.

Sprawdzamy więc te takty w tab. 5, dla których powtarza się stopień łączenia. Takty 0 i 6 nie są oczywiście sprzeczne, ale takty 3 i 5 przy tym samym stopniu łączenia (równym 4) mają różną funkcję wzbudzenia. Tabela jest więc nierozwiązalna, bo dwa takty są sprzeczne. Sprzeczność tkwi w tym, że te sam stan elementów [*MAZ*] = [100] raz ma powodować działanie silnika w kolejnym takcie, a drugi raz ma powodować wyłączenie tego silnika. W takim przypadku, aby usunąć sprzeczność i uczynić tabelę rozwiązalną należy wprowadzić dodatkowy element pośredniczący, który powinien zadziałać po takcie 4 i w ten sposób rozróżnić jednakowe stany. Wprowadzony element będzie w ten sposób spełniał w układzie rolę brakującej pamięci. Dobranie taktu, w którym dany element pośredniczący ma zadziałać i taktu, w którym ma zostać wyłączony jest w pewnym zakresie dowolne, byleby tylko zapewniało zniknięcie powtarzających się, sprzecznych taktów układu. Najlepiej okres pracy elementów pośredniczących dobierać tak, aby trwał on możliwie najkrócej. Liczba *p* wymaganych elementów pośredniczących i liczba *s* sprzecznych taktów są związane zależnością $2^p \geqslant s$. W naszym przypadku *s*=2, więc jeden element pośredniczący, czyli *p*=1 spełnia ten warunek.

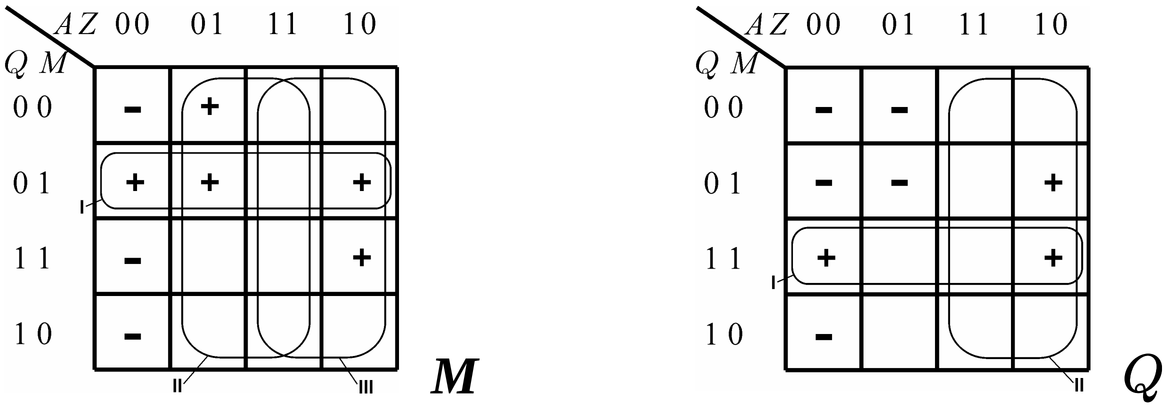
Tab. 6 przedstawia tabelę łączeń otrzymaną po wprowadzeniu elementu dodatkowego *Q*. Zgodnie z podaną wcześniej zasadą zaznaczono w niej funkcję wzbudzeń dla elementów wyjściowych i pośredniczących, jakimi są teraz *M* i *Q*. Jak łatwo zauważyć, szereg kolejnych plusów przesunięty jest zawsze <u>o jeden takt w lewo</u> względem szeregu jedynek. Pamiętając o tym można łatwo wpisać znaki "**+**", a w pozostałych miejscach wpisać "**−**". Teraz nie ma sprzecznych taktów (takty 0 i 7 nie są sprzeczne, bo funkcja wzbudzeń jest dla nich jednakowa), więc tabela jest rozwiązalna. Oznacza to, że elementy *Q*, *M*, *A*, *Z* wystarczą już do znalezienia odpowiednich równań logicznych i zbudowania na tej podstawie układu logicznego, realizującego zadane działanie.

Równania logiczne elementów wyjściowych i pośredniczących *M* i *Q* można znaleźć, sporządzając dla nich tablice Karnaugha i wpisując do nich wszystkie funkcje wzbudzeń "**+**" i "**−**" z tab. 6. Tablice Karnaugha pokazano na rys. 8. Zewnętrze tablic opisane jest stanami elementów *Q*, *M*, *A*, *Z*.

Dokonując minimalizacji, w otrzymanych tablicach traktujemy "**+**"<u> jako logiczne 1</u>, natomiast "**−**"<u> jako logiczne 0</u>. W tablicy występują też puste kratki, co wiąże się z tym, że w tabeli kolejności łączeń nie występują wszystkie możliwe kombinacje czterech sygnałów *Q*, *M*, *A*, *Z*, a tylko te, które są istotne dla żądanego działania układu (teoretycznie wszystkich kombinacji jest tyle, co w tablicy Karnaugha, czyli 2^4=16). Pusta kratka, czyli stan obojętny, jest w tablicy Karnaugha zawsze zjawiskiem jak najbardziej korzystnym, ponieważ można go wykorzystać przy minimalizacji jako zero lub jedynkę, aby uzyskać możliwie największe obszary sklejania (im większe te obszary, tym prostsza funkcja logiczna). Takie właśnie obszary zaznaczono cienką linią w obu tablicach. Jak widać jest to minimalizacja „jedynkowa”, gdyż plusy odpowiadają logicznym jedynkom.

Tab. 6. Tabela kolejności łączeń po wprowadzeniu elementu Q

	Numer taktu →	0	1	2	3	4	4'	5	6	7
$2^0=1$	przycisk Z	0	1	1	0	0	0	0	0	0
$2^1=2$	zestyk A	0	0	0	0	1	1	0	0	0
$2^2=4$	silnik M	0 −	0 +	1 +	1 +	1 +	1 +	1 −	0 −	0 −
$2^3=8$	element pośredn. Q	0 −	0 −	0 −	0 −	0 +	1 +	1 +	1 −	0 −
	Stopień łączenia	0	1	5	4	6	14	12	8	0



Rys. 8. Tablice Karnaugha dla funkcji wzbudzeń *M* i *Q*

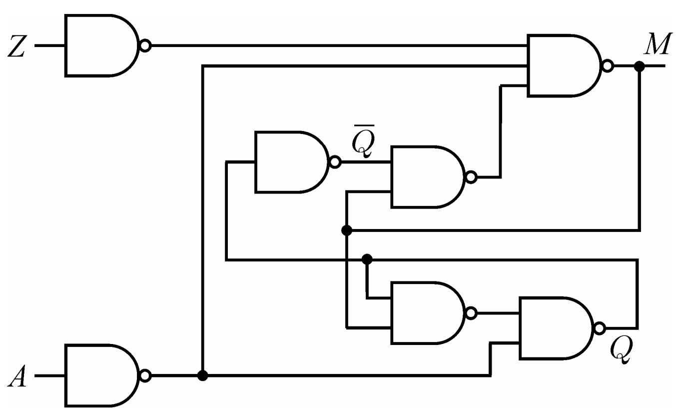
Zminimalizowane postacie równań logicznych, czyli funkcji wzbudzeń dla elementów *M* i *Q*, napisane wg znanej reguły sklejania wyglądają następująco:

$$M = \overline{Q}M + Z + A \qquad \text{oraz} \qquad Q = QM + A \tag{1}$$

Aby zbudować układ logiczny z elementów NAND należy dokonać prostego przekształcenia:

$$M = \overline{\overline{\overline{Q}M + Z + A}} = \overline{\overline{\overline{Q}M} \cdot \overline{Z} \cdot \overline{A}} \quad \text{oraz} \quad Q = \overline{\overline{QM + A}} = \overline{\overline{QM} \cdot \overline{A}} \tag{2}$$

Na podstawie powyższych zależności można już narysować schemat połączeń, pokazany na rys. 9.



Rys. 9. Schemat układu logicznego sterującego silnikiem zrealizowany z bramek NAND

Jak widać, w układzie utworzyły się sprzężenia zwrotne, tworzące elementy pamięci charakterystyczne dla układu sekwencyjnego. Te sprzężenia widać już z postaci zależności (1), gdyż w równaniu logicznym określającym *M* występuje ten sam element *M*; podobnie jest też dla *Q*.

Należy zaakcentować, że podane wiadomości stanowią tylko zasygnalizowanie bogatej problematyki, związanej z teorią i wykorzystaniem cyfrowych układów logicznych w automatyce.

Przebieg ćwiczenia

1. Zmontować i sprawdzić działanie przerzutnika *RS*.
2. Sprawdzić działanie przerzutnika typu *D*, podłączając do niego sygnały wg zadanego schematu.
3. Wykorzystując przerzutnik *JK* jako dwójkę liczącą zmontować układ licznika do 8, łącząc szeregowo 3 przerzutniki. Wyzerować układ podając przez chwilę na wszystkie wejścia *S* sygnał 0. Zarejestrować sygnały wszystkich wyjść *Q*, zadając na wejście zegarowe pierwszego przerzutnika szereg impulsów, aż do powtórzenia się początkowego, zerowego stanu wyjść.
4. Zaprojektować i zmontować układ realizujący założenia podane przez prowadzącego. Sprawdzić poprawność działania zmontowanego układu.
5. a) Zmontować i sprawdzić działanie układu z rys. 9.

   b) Zmodyfikować układ z rys. 9 tak, aby przy początkowym stanie $Q=M=A=Z=0$ przypadkowe wciśnięcie zestyku A (np. przez pracownika obsługującego urządzenie) nie spowodowało załączenia silnika. Dopisać dodatkowe takty cyklu, uzupełnić tablice Karnaugha, napisać nowe, zminimalizowane postacie równań logicznych. Naszkicować schemat poprawionego układu, połączyć i sprawdzić działanie.

Literatura

1. *Układy przełączające w automatyce przemysłowej. Zadania.* Praca zbiorowa pod red. Henryka Małysiaka. WNT, Warszawa 1981.
2. Włodzimierz Sasal: *Układy scalone serii UCA64/UCY74. Parametry i zastosowania*. WKiŁ, Warszawa 1985.
3. Jerzy Siwiński: *Układy przełączające w automatyce.* WNT, Warszawa 1968.
4. *Laboratorium teorii maszyn, drgań mechanicznych i podstaw automatyki*. Praca zbiorowa pod red. Mirosława Roszkowskiego. Wydawnictwo PŁ, Łódź 1988.