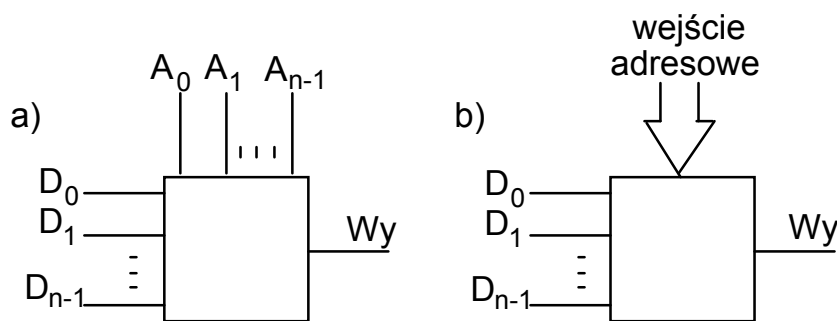
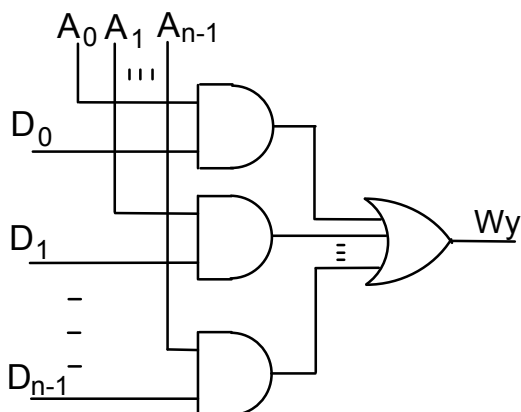


## Multiplexer, dekodery, demultiplexer, koder.

Wykorzystując  $n$ -wejściową bramkę logiczną OR oraz  $n$  dwuwejściowych bramek AND można zbudować układ (rysunki: 1 oraz 2), w którym poprzez podanie odpowiedniej kombinacji stanów logicznych na wejścia  $A_0 - A_{n-1}$  spowodujemy, że na wyjście układu  $Wy$  będzie przenoszony sygnał cyfrowy z jednego, wybranego przez nas wejścia, spośród wejść  $D_0 - D_{n-1}$ . Układ przenoszący sygnał na wyjście z jednego wybranego wejścia spośród wielu wejść nazywa się multiplexerem. Pod pojęciem „sygnał cyfrowy” będziemy rozumieli ciąg stanów logicznych, L i H, zmieniających się w czasie. Zwykle przyjmuje się, że stan log. L reprezentuje cyfrę 0, zaś stan H - cyfrę 1. Wejścia  $A_0 - A_{n-1}$  stanowią tzw. wejście adresowe. Wejścia  $D_0 - D_{n-1}$  nazywają się wejściami informacyjnymi lub wejściami danych. Rysunek 1b przedstawia ten sam układ, z tym, że inaczej zostało zaznaczone wejście adresowe. Rysunek 2 przedstawia schemat szczegółowy układu. Jak widzimy, układ składa się z szeregu dwuwejściowych bramek AND oraz jednej wielowejściowej bramki OR.



Rys. 1. Schematy ogólne multiplexera.



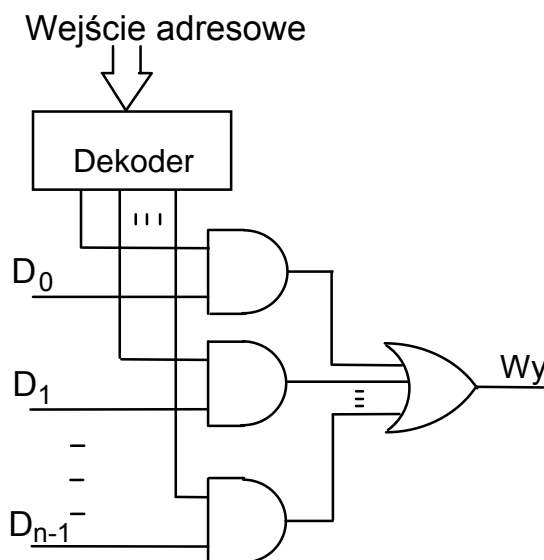
Rys. 2. Schemat szczegółowy multiplexera.

Warunkiem na to, aby sygnał cyfrowy był przenoszony na wyjście bramki OR z danego wejścia bramki jest istnienie stanu logicznego L na pozostałych wejściach bramki. Szereg bramek AND w układzie przedstawionym na rys. 2 umożliwia spełnienie tego warunku. Otóż, podając na je-

den przewód w wejściu adresowym stan logiczny H a na pozostałe przewody - stan L, spowodujemy, że na wyjście układu będzie przenoszony sygnał z jednego tylko wejścia informacyjnego: z tego, które prowadzi do bramki AND, mającej na przewodzie "adresowym" stan logiczny H. Na przykład gdy w układzie przedstawionym na rys. 2 na przewodzie  $A_1$  istnieje stan H, zaś na pozostałych przewodach w wejściu adresowym istnieją stany L, to na wyjście układu przechodzi sygnał z wejścia  $D_1$ .

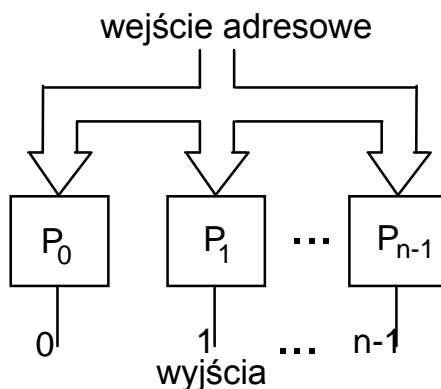
W celu zmniejszenia liczby przewodów w wejściu adresowym stosuje się kodowanie adresu. Zamiast podawać  $n$  stanów logicznych na  $n$  przewodów, podaje się kombinację stanów logicznych, reprezentującą liczbę binarną, określającą wejście informacyjne, z którego ma być przenoszony sygnał na wyjście układu. Za pomocą  $k$ -przewodowej linii można przesyłać dwójkowo  $2^k$  adresów.

Oczywiście, układ z tak kodowanym wejściem adresowym musi zawierać w odpowiednim miejscu dekodery przekładający liczby binarne na wymagane kombinacje stanów logicznych, takie, że na jednym (ściśle określonym) wyjściu występuje stan H, zaś na pozostałych wyjściach - stan L. (Może być i na odwrót: na jednym wyjściu stan L, zaś na pozostałych wyjściach - stan H; zależy to od konkretnego rozwiązania całości układu.) Rysunek 3 przedstawia schemat ogólny układu z dekoderym.



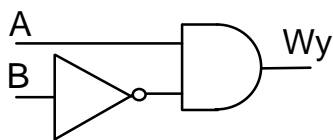
Rys. 3. Schemat ogólny multipleksersa z zaznaczonym dekoderym.

Dekoder można przedstawić jako  $n$  podukładów  $P_0 - P_{n-1}$  (rys. 4). Liczba binarna jest podawana jednocześnie na wejścia wszystkich podukładów. Każdy podukład wykrywa właściwą sobie liczbę binarną  $i$  - jeśli ją wykryje - podaje stan logiczny H na swoje wyjście.

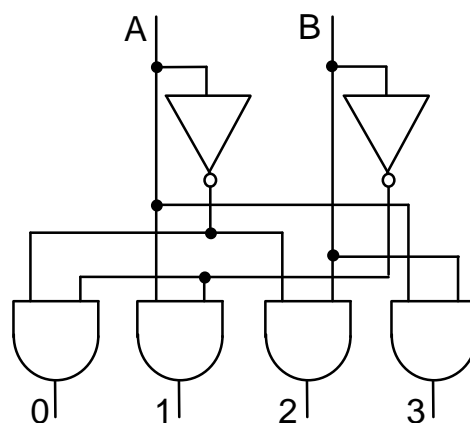


Rys.4. Schemat ogólny dekodera, jako n podukładów.

Podukład wykrywający daną liczbę można łatwo zbudować z wielowejsciowej bramki logicznej i pewnej liczby elementów zaprzeczenia logicznego. Jeśli dana binarna  $m$ -pozycyjna liczba jest reprezentowana przez  $k$  stanów logicznych H i  $l$  stanów logicznych L ( $k+l=m$ ), to podukład wykrywający tę liczbę będzie się składał z  $m$ -wejściowej bramki AND oraz  $l$  elementów zaprzeczenia logicznego. Elementy zaprzeczenia logicznego powinny zmienić stany logiczne L tak, by - wtedy, gdy do podukładu dochodzi właściwa jemu liczba - wszystkie stany dochodzące do bramki AND były stanami H. Wtedy na wyjściu bramki, które jest zarazem wyjściem podukładu, będzie stan H. Na rysunku 5 został przedstawiony schemat podukładu wykrywającego liczbę dwójkową 10. Jeśli na wejście A jest podawany stan logiczny H (tzn. cyfra 1), zaś na wejście B stan logiczny L (cyfra 0), to na wyjściu istnieje stan logiczny H. W pozostałych przypadkach na wyjściu będzie stan logiczny L.



Rys. 5. Podukład wykrywający liczbę 01.

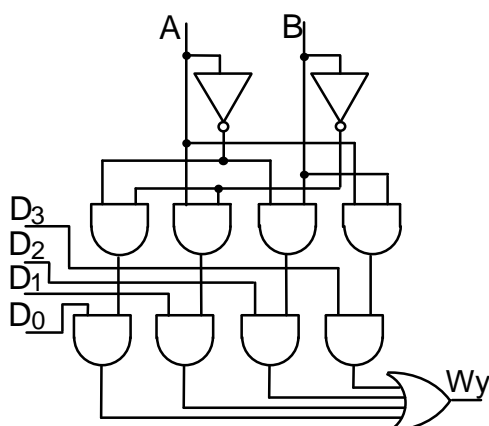


Rys. 6. Dekoder kodu binarnego na kod 1 z 4.

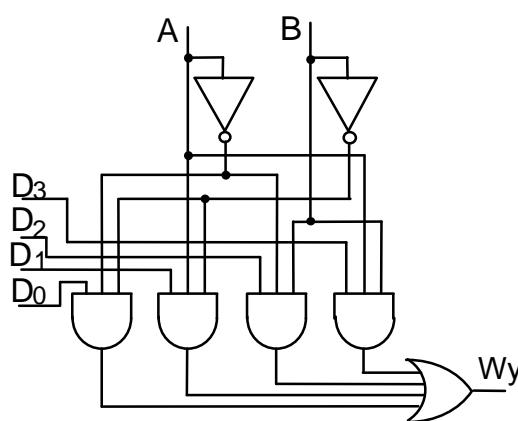
Okazuje się, że w celu zbudowania dekodera korzystnie jest uzyskać w oddzielnym zespole elementów NOT zaprzeczenia wszystkich stanów logicznych występujących w wejściu adresowym i na wejścia bramek AND podać odpowiednie kombinacje niezaprzeczonych i zaprzeczonych

stanów logicznych z wejścia adresowego. Na rysunku 6 został przedstawiony schemat dekodera posiadającego dwuprzewodowe wejście. Dekoder ten nosi nazwę „dekodera kodu binarnego na kod 1 z 4”. W ogólności dekodery takie noszą nazwę „dekodera kodu binarnego na kod 1 z n”.

Na rysunku 7 został pokazany schemat układu posiadającego cztery wejścia informacyjne, dwuprzewodowe wejście adresowe dla kodu binarnego oraz dekodery. Można zmniejszyć ilość bramek w układzie, stosując bramki o większej ilości wejść. Na rysunku 8 został przedstawiony schemat układu równoważny (logicznie) układowi z rys. 7. Układ przedstawiony na rys. 8 jest szybszy w działaniu, ponieważ posiada o jeden szereg bramek mniej.

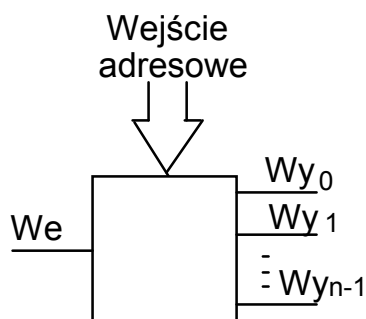


Rys. 7. Schemat czterowejściowego multiplexera z dekoderni.



Rys. 8. Multiplexer z pojedynczym szeregiem bramek.

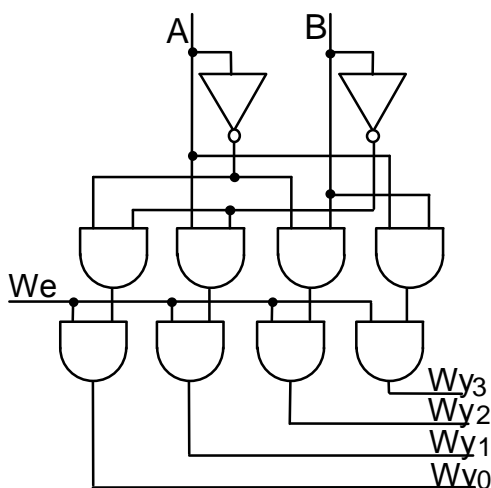
Możliwe jest zbudowanie układu działającego odwrotnie, niż układ już opisany. Układ przenoszący sygnał z wejścia na jedno wybrane wyjście spośród wielu wyjść nazywa się demultiplexerem. W układzie takim (rys. 9) sygnał cyfrowy jest przesyłany z wejścia na jedno wybrane wyjście spośród wielu wyjść. Na pozostałych wyjściach istnieje ustalony stan logiczny, np. stan L.



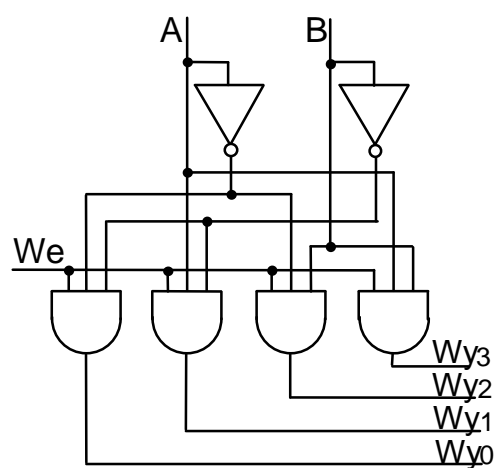
Rys. 9. Schemat ogólny demultiplexera.

Aby można było numer wybranego wyjścia podawać w kodzie binarnym, należy także i tutaj zastosować dekodery kodu binarnego na kod 1 z n. Rysunek 10 przedstawia schemat takiego układu, posiadającego

dwuprzewodowe wejście dla kodu binarnego, jedno wejście informacyjne oraz cztery wyjścia. Dwa elementy zaprzeczenia logicznego wraz z górnym szeregiem dwuwejściowych bramek AND tworzą znany nam już dekodery kodu binarnego na kod 1 z 4. W szeregu dwuwejściowych bramek AND sygnał wejściowy jest poddawany „bramkowaniu” sygnałami pochodzącymi z dekodera. Rysunek 11 przedstawia układ równoważny układowi przedstawionemu na rys. 10, zawierający tylko jeden szereg bramek AND.

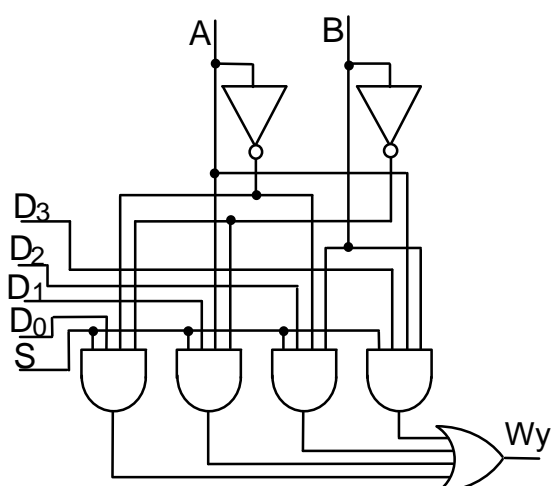


Rys. 10. Schemat szczegółowy cztero-wyjściowego demultiplexera.

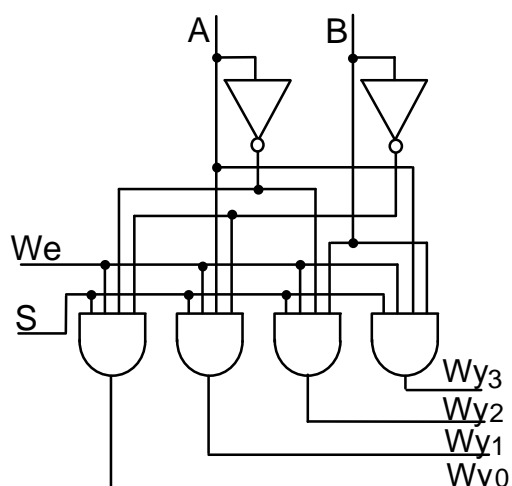


Rys. 11. Układ równoważny układowi z rys. 10.

Układy opisane są produkowane fabrycznie. Na rysunku 12 został przedstawiony schemat multiplexera, na rysunku 13 - schemat demultiplexera. Oprócz wejść informacyjnych i adresowych przedstawione układy posiadają tzw. wejścia strobowe. Logiczne działanie sygnału strobowego sprowadza się do „bramkowania” w elemencie AND tym sygnałem: sygnału wyjściowego w multiplexersze i sygnału wejściowego w demultiplexersze. Elementy AND posiadają w tych układach o jedno wejście więcej, niż w układach przedstawionych na rysunkach 8 i 11. Na te dodatkowe wejścia podajemy sygnał strobowy.



Rys. 12. Schemat multipleksera.



Rys. 13. Schemat demultipleksera.

S - wejście strobowe.

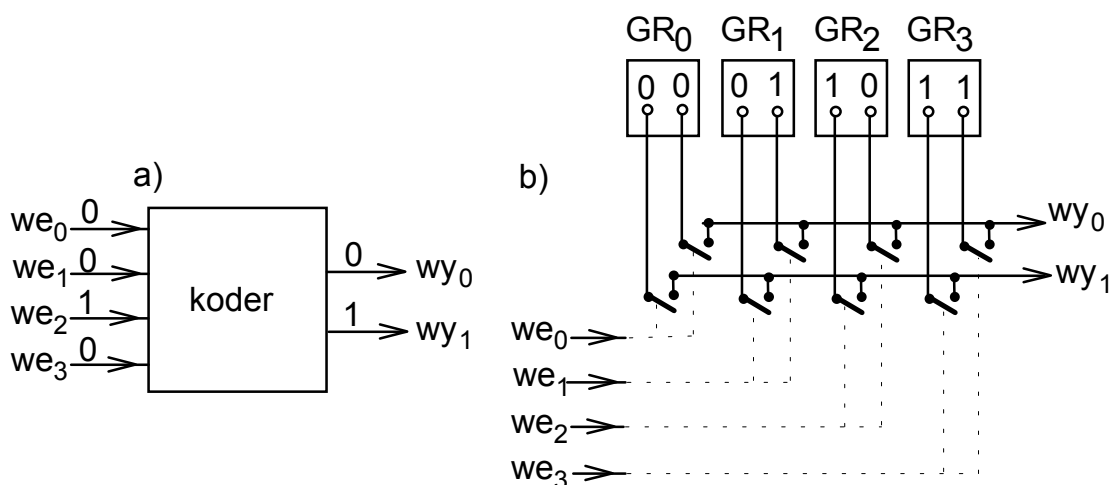
Omówione układy znajdują rozmaite zastosowania w technice cyfrowej. Multipleksery bywają wykorzystywane np. w układach do przesyłania informacji pochodzących z wielu źródeł za pomocą linii jedнопроводowej. Demultiplekser, stający się w łatwy sposób dekoderym kodu binarnego na kod 1 z n, znajduje zastosowanie w układach zmieniających kod binarny na kody inne (niż 1 z n).

16-nóżkowy układ scalony typu UCY 74151 jest przykładem produkowanego fabrycznie multipleksera, posiadającego 3-przewodowe wejście adresowe, 8-przewodowe wejście danych, wejście strobowe oraz 2 przewody wyjściowe (jeden z nich stanowi "zaprzeczenie" drugiego).

Przykładem fabrycznie produkowanego demultipleksera może być 24-nóżkowy układ scalony UCY74154, posiadający 4-przewodowe wejście adresowe, 2 przewody, z których jeden możemy uważać za wejście danych a drugi za wejście strobowe oraz 16 przewodów wyjściowych. Jeżeli w tym układzie na wejście strobowe podamy stan logiczny niski, to - podając liczbę dwójkową na wejście adresowe - wskazujemy numer przewodu wyjściowego, na które jest przenoszony sygnał z wejścia danych; na pozostałych przewodach wyjściowych będzie stan logiczny wysoki.

Układ scalony typu UCY 7442 jest produkowanym fabrycznie dekoderym przekładającym binarną postać liczb z zakresu od 0 do 9 na postać "1 z 10".

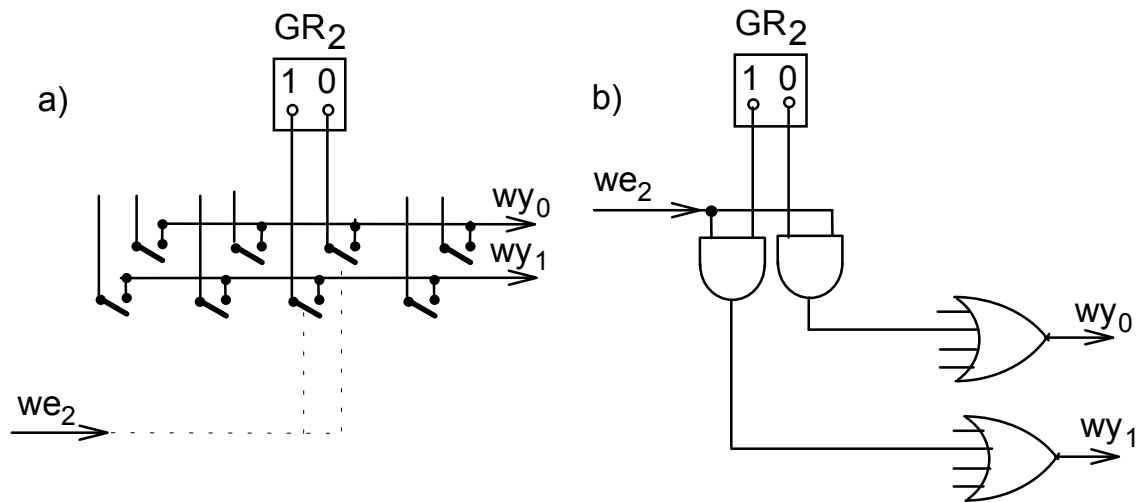
Urządzenie działające odwrotnie w stosunku do dekodera nazywa się koderem. Układ taki ma n wejść i określoną liczbę wyjść - taką, aby - w określonym kodzie - dało się zaprezentować każdą z liczb od 0 do n-1. Wejścia są "ponumerowane" od 0 do n-1. Wyróżnienie jednego określonego wejścia poprzez podanie na nie stanu wysokiego (na pozostałych wejściach ma być wtedy stan niski), oznaczające wprowadzenie na wejście kodera liczby równej numerowi wejścia, powoduje pojawienie się na przewodach wyjściowych kombinacji stanów logicznych reprezentującej w jakimś kodzie (często jest to kod binarny) wprowadzonej liczby.



Rys. 14. Koder 4-wejściowy dający na wyjściu zapis binarny: schemat ogólny (rys. a) oraz schemat działania (rys. b).

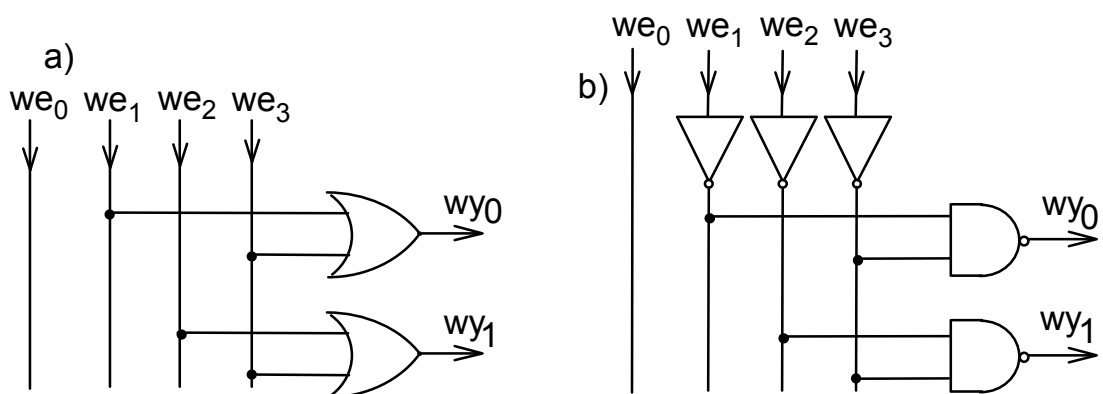
Możliwa jest także logika odwrotna: na jedno, wyróżnione wejście kodera podajemy stan niski a na pozostałe wejścia stan wysoki.

Rysunek 14a przedstawia schemat ogólny kodera 4-wejściowego, zamieniającego liczbę wyrażoną w postaci "1 z 4" na liczbę wyrażoną w postaci binarnej. Układ ma 2 wyjścia. Przy podanym - dla przykładu - na wejście "we<sub>2</sub>" stanie logicznym 1 (czyli przy liczbie 2 na wejściu) mamy na wyjściu binarną postać liczby "dwa": 10. Rysunek 14 b zawiera schemat tłumaczący sposób działania kodera. Schemat taki może być pomocny przy projektowaniu budowy kodera. Układ na rys. 14b posiada 4 wejścia oraz 4 grupy ( $GR_0$ ,  $GR_1$ ,  $GR_2$  i  $GR_3$ ) przewodów, na które zostały podane "na stałe" odpowiednie kombinacje stanów logicznych. Kolejnym wejściom, poczynając od wejścia "we<sub>0</sub>", zostały przyporządkowane grupy przewodów z kombinacjami stanów logicznych, reprezentujących binarne zapisy liczb: 0, 1, 2 i 3. Podanie na któreś wejście jedynki logicznej powoduje połączenie przewodów wyjściowych układu z grupą przewodów przyporządkowanych temu wejściu. Widoczne na schemacie pary kluczy, włączane podaniem na jedno z czterech wejść stanu wysokiego, można zastąpić odpowiednimi zestawami bramek. Na rysunku 15 pokazano fragment układu z rys. 14b oraz jego realizację za pomocą bramek.



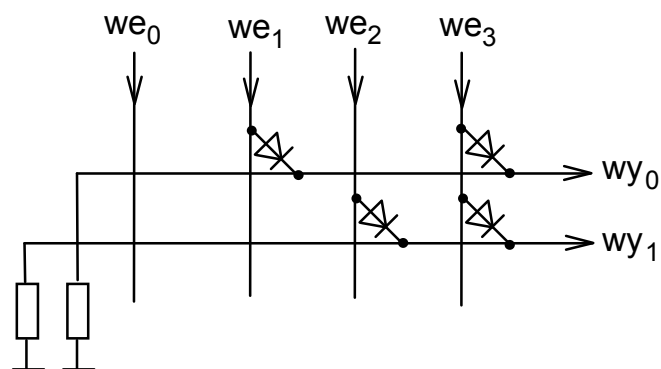
Rys. 15. Realizacja funkcji kluczy za pomocą bramek.

Istnieje prostsze rozwiązanie budowy kodera. Pokazuje je (dla przypadku 4 wejść) rysunek 16a. Równoważny układ zbudowany z bramek NAND pokazuje rys. 16b. Wzorując się na schematach pokazanych na rys. 16 łatwo jest projektować kodery dla większej ilości wejść.



Rys. 16. Schematy 4-wejściowego kodera binarnego.

Szczególnie proste rozwiązania zawierają diody (i oporniki). Na rys. 17 przedstawiono realizację 4-wejściowego binarnego kodera zbudowanego z diod.



Rys. 17. Diodowa realizacja 4-wejściowego kodera.



Co będzie, gdy w wejściu funkcjonującym na zasadzie "1z n" podamy jedynkę logiczną na dwa wejścia, albo nie podamy jedynki na żadne wejście? - Wystąpi błąd. Możliwe jest zbudowanie układu sygnalizującego istnienie błędu. W takim układzie (z sygnalizacją błędu) istnieje dodatkowe wyjście, sygnalizujące wystąpienie błędu poprzez pojawienie się - na tym wyjściu - określonego stanu logicznego.

Często zamiast „czystego” kodu binarnego stosuje się tzw. kod BCD. Jest to dziesiętny zapis liczby, z tym że poszczególne cyfry (od 0 do 9) są przedstawiane w kodzie binarnym (za pomocą czterech cyfr binarnych). Na przykład liczba „dziesięć”, mająca postać binarną jako „1010” w zapisie BCD ma postać „1 0000” lub „0001 0000”. (W zapisie BCD - aby uzyskać większą przejrzystość, grupuje się w czwórki cyfry reprezentujące poszczególne dekady.) Układ, który przekłada zapis "1 z n" na zapis BCD także jest koderem. Podobnie koderem będzie układ przekładający zapis "1 z n" na zapis w jakimś innym jeszcze kodzie.

Istnieją układy, zamieniające zapis liczby w jednym kodzie na zapis tej samej liczby w innym kodzie. Są to tzw. transkodery lub konwertery. Przykładem może być układ scalony UCY7447, który przekłada postać binarną cyfry na kombinację odpowiednich stanów na siedmiu przewodach zasilających wyświetlacz siedmiosegmentowy.

Transkoder można rozumieć, jako zespół dwu podukładów: dekodera i kodera. W takim zespole każda "używana" (\*) kombinacja stanów wejścia ma "przypisaną" elektrodę (jeden z przewodów wyjścia dekodera), na której występuje jedynka logiczna wtedy i tylko wtedy, gdy dana kombinacja stanów istnieje na wejściu. W ten sposób (w danej) chwili jedynka (\*\*) logiczna występuje tylko na jednym przewodzie wyjściowym dekodera. Przewody wyjściowe dekodera są jednocześnie przewodami wejściowymi kodera. Każdemu danemu przewodowi wejścia kodera odpowiada określona kombinacja stanów logicznych na jego wyjściu, która to kombinacja na wyjściu wystąpi, gdy na pojawi się jedynka logiczna na danym przewodzie wejściowym.

(\*) Nie wszystkie kombinacje stanów są "używane", np. w kodzie BCD nie występują kombinacje odpowiadające liczbom od 10 do 15.

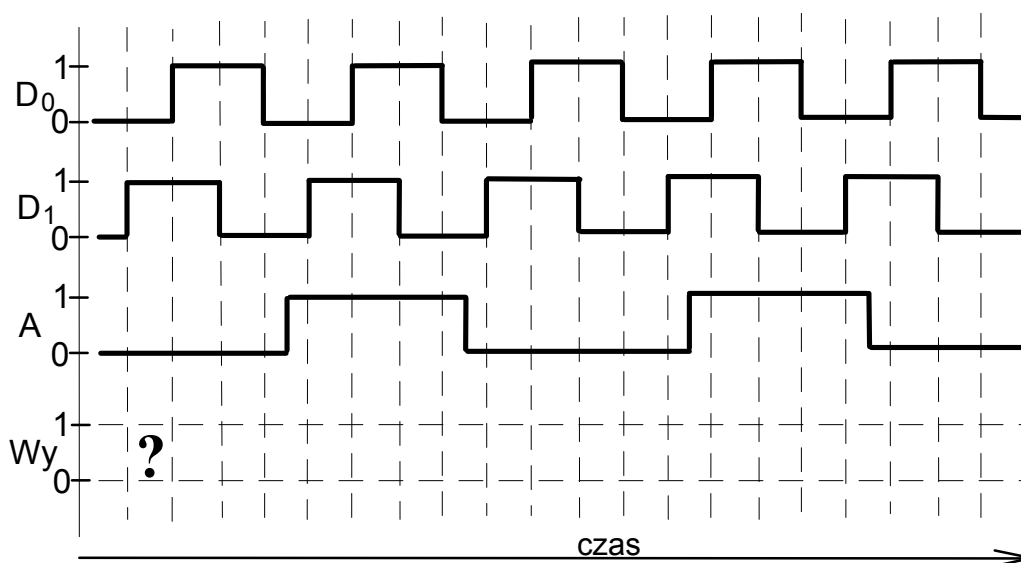
(\*\*) Może być rozwiązanie odwrotne: na "przypisanej" elektrodzie występuje zero logiczne a na pozostałych elektrodach jedynki.

Należy tutaj dodać, że układ realizujący zestaw funkcji logicznych i arytmetycznych, posiadający wejścia dla cyfr czy też liczb oraz wejścia określające rodzaj wykonywanej operacji oraz wyjście dla wyniku (w wyjściu mogą być także przewody określone jako "przeniesienie do następnej pozycji" i "pożyczka") może być zbudowany w podobny sposób, jak założony wyżej sposób budowy transkodera, gdyż każdej kombinacji stanów wejść w takim układzie odpowiada jednoznacznie jakaś jedna kombinacja stanów wyjść.

W praktyce, o ile to tylko możliwe, stosuje się uproszczenia budowy, tak aby układ posiadał jak najmniej elementów. Z tego powodu nie każdy transkoder posiada wewnątrz dekodery, wyraźnie odróżnialny od reszty układu.

Plan ćwiczenia.

1. Zbudować układ posiadający dwa wejścia informacyjne  $D_0$  i  $D_1$  i jednoprzewodowe wejście adresowe  $A$ , pozwalający na przesyłanie sygnału z wybranego wejścia na wyjście. Sprawdzić działanie układu i podać wykres zmian stanów logicznych na wyjściu w zależności od stanów logicznych na wejściach, zmieniających się zgodnie z rys. 18.



Rys. 18. Przebieg stanów logicznych na wejściach dwuwejściowego układu, do punktu 1 ćwiczenia.

Do budowy układu wystarczy jeden układ scalony UCY7400, zawierający cztery dwuwejściowe bramki NAND. W celu wykonania zadania można wziąć za podstawę schemat przedstawiony na rys. 8. Schemat ten należy uprościć tak, by powstał układ posiadający tylko wejścia  $D_0$ ,  $D_1$  i  $A$ . Następnie w otrzymanym schemacie należy zastąpić element zaprzeczenia logicznego dwuwejściową bramką NAND (Jak otrzymać element zaprzeczenia logicznego z elementu NAND?) zaś zespół złożony z dwu dwuwejściowych bramek AND i jednej dwuwejściowej bramki OR zastąpić zespołem złożonym z trzech dwuwejściowych bramek NAND - tutaj wykorzystujemy twierdzenie:  $A+B = \overline{\overline{A} \cdot \overline{B}}$  („zaprzeczenie iloczynu zaprzeczeń jest równoważne sumie logicznej”).

2. Zaprojektować i zbudować układ posiadający 3 wejścia informacyjne, wyjście oraz dwuprzewodowe wejście adresowe dla kodu binarnego. Podstawą wyjściową budowy układu będzie tutaj - podobnie jak w

punkcie 1 ćwiczenia - schemat przedstawiony na rys. 8. Do budowy użyć dwu układów scalonych UCY7410. Każdy z tych układów zawiera trzy trójwejściowe elementy NAND. Dwa takie elementy mają zostać wykorzystane do realizacji funkcji zaprzeczenia logicznego. W miejsce potrzebnego tutaj zespołu złożonego z trzech trójwejściowych bramek AND i jednej trójwejściowej bramki OR (taka potrzeba wynika wprost ze schematu przedstawionego na rys.8) należy zastosować zespół złożony z pozostałych czterech trójwejściowych bramek NAND - postępujemy podobnie, jak w punkcie 1 ćwiczenia. Zbadać i opisać działanie układu dla wszystkich możliwych kombinacji stanów logicznych na przewodach A i B w wejściu adresowym.

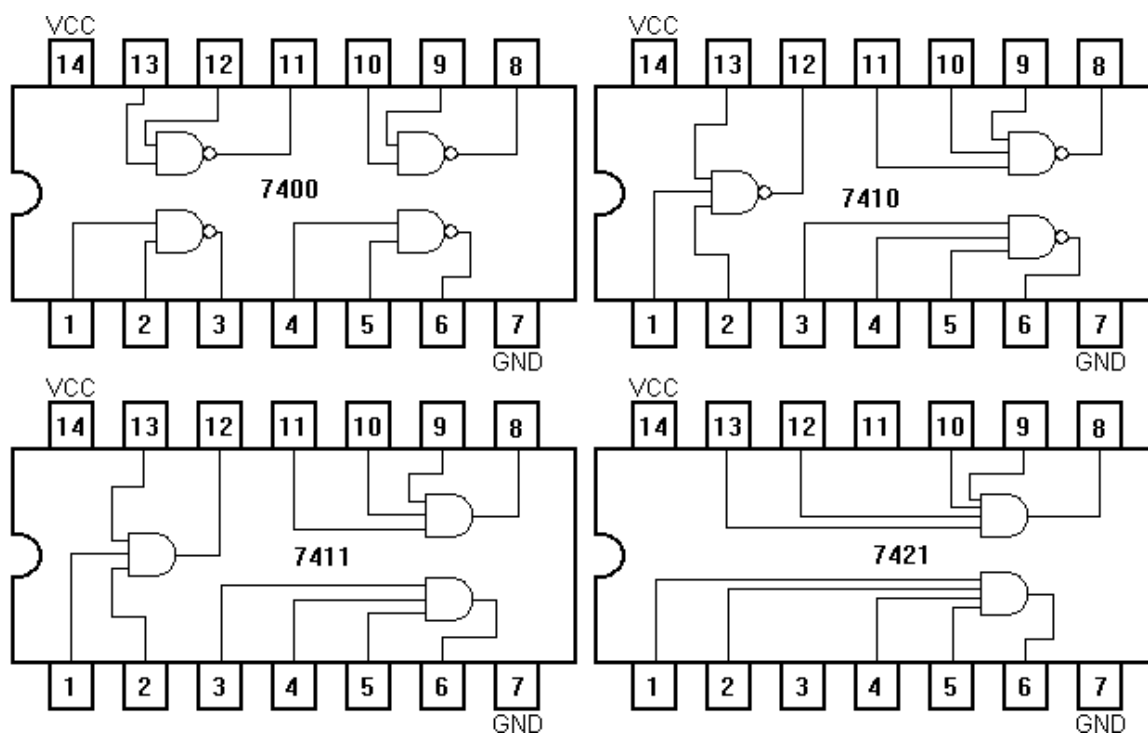
3. Zaprojektować dekodery kodu binarnego na kod 1 z 10. Dekoder powinien posiadać 4 wejścia A, B, C i D oraz 10 wyjść. Podstawą tutaj jest schemat przedstawiony na rys. 6, który należy rozszerzyć o dwa wejścia: C i D (potrzebne będą tutaj dwa dodatkowe elementy zaprzeczenia logicznego). Zamiast czterech dwuwejściowych bramek AND należy narysować dziesięć czterowejściowych bramek AND. Do każdej bramki doprowadzamy (na rysunku) odpowiednią kombinację sygnałów „niezaprzeczonych” i „zaprzeczonych” (trzeba będzie użyć czterech elementów zaprzeczenia logicznego) z wejść A, B, C i D. Zbudować fragment tego dekodera, dekodujący wybrane trzy cyfry, np. 0, 5 i 9. Do budowy użyć czterech dwuwejściowych bramek NAND (mają one spełniać funkcje zaprzeczenia logicznego; potrzebny 1 układ scalony UCY7400) oraz trzy czterowejściowe bramki AND (potrzebne są dwa układy scalone UCY 7421). Sprawdzić działanie zbudowanego fragmentu. Wyniki przedstawić w postaci wykresów przedstawiających zmiany stanów logicznych w czasie na wejściu i na wyjściu dekodera tak, by stan logiczny H (cyfra 1) pojawiał się kolejno na przewodach wyjściowych 0, 1, 2, ..., 9. Zaznaczyć części wykresu odnoszące się do zbudowanego fragmentu.
4. Zbudować układ wg schematu z rys. 11. Do budowy użyć dwu dwuwejściowych bramek NAND (1 układ scalony UCY7400) oraz czterech trójwejściowych bramek AND (dwa układy scalone UCY7411). Bramki NAND mają pełnić rolę elementów zaprzeczenia logicznego. Sprawdzić działanie układu. Wyniki przedstawić w postaci wykresów tak, by dostatecznie jasno tłumaczyły działanie układu. Co należy uczynić, aby układ był dekoderym kodu binarnego na kod 1 z 4?
5. Wzorując się na rys. 16a zaprojektować koder mający 16 wejść (od "we<sub>0</sub>" do "we<sub>15</sub>"). Uprościć narysowany schemat do układu zawierającego 10 wejść przez wykreślenie odpowiedniej liczby linii przewodów na schemacie. Narysować schemat kodera zbudowanego z bramek NAND, przekładającego na postać binarną liczby z zakresu od 0 do 7 a następnie, wykorzystując 1 układ scalony UCY7410 oraz 1 układ UCY7400, zbudować fragment tego kodera, przekładający na postać

binarną 3 dowolnie wybrane liczby z zakresu od 0 do 7. Przeprowadzić zbudowany układ i w opracowaniu zamieścić stosowne wyniki.

Uwaga. W opracowaniu wyników należy zamieścić szczegółowe schematy logiczne zbudowanych układów.

Do wykonania ćwiczenia potrzebny jest zasilacz napięcia stałego 5V, układy scalone do montowania układów: UCY7400 (1 szt.), UCY7410 (2 szt.), UCY7421 (2 szt.) i UCY7411 (2szt.), zespół przełączników do podawania różnych stanów logicznych na wejścia budowanych układów, zespół wyświetlaczy diodowych do badania stanów logicznych na wyjściach układów.

Na rysunku 15 zostały przedstawione schematy wyprowadzeń układów scalonych używanych w ćwiczeniu.



Rys. 19. Schematy wyprowadzeń układów scalonych. VCC oznacza wyprowadzenie do podłączenia dodatniego bieguna zasilacza (+5V), GND oznacza wyprowadzenie do podłączenia ujemnego bieguna zasilacza (0V).

Na wejścia budowanych układów podajemy stany logiczne (potencjały 0V, czyli stany logiczne L oraz potencjały +5V, czyli stany logiczne H) z zespołu przełączników zamontowanych na płytce. Do zespołu przełączników doprowadzamy z zasilacza dwa przewody o potencjałach: 0V i +5V. Gdy przesuwka przełącznika jest przesunięta „do góry”, to na wyjściu przełącznika istnieje stan H, gdy przesuwka

przełącznika jest przesunięta „do dołu”, to na wyjściu przełącznika istnieje stan H.

Do sprawdzania stanów logicznych na wyjściach budowanych układów służy zespół diod świecących zamontowanych na płytce. Elektrode wspólną diod (przewód do którego są przyłączone katody diod świecących) łączymy z "zerem" zasilacza. Na poszczególne pojedyncze elektrody, do których są przyłączone anody poszczególnych diod, podajemy sygnały z wyjść budowanych układów. Świecenie diody oznacza stan logiczny "1", brak świecenia - stan logiczny "0".

#### LITERATURA

1. Jan Pieńkos, Janusz Turczyński: „Układy scalone TTL w systemach cyfrowych”, WKŁ, Warszawa, 1980.
2. Jan Pieńkos, Janusz Turczyński: „Układy scalone TTL serii UCY74 i ich zastosowanie”, WKŁ, Warszawa, 1977.
3. Andrzej Sowiński: „Cyfrowa technika pomiarowa”, WKŁ, Warszawa, 1975.
4. Wiesław Traczyk: „Układy cyfrowe automatyki”, Wyd. Naukowo-Techniczne, Warszawa, 1976.
5. Opis ćwiczenia p.t. „Elementy logiczne”.

Roman Kazański.

Wersja rozszerzona.  
Lublin, 25 maja, 2001r.  
Ostatnia zmiana 17 stycznia 2003r.