

Pamięci półprzewodnikowe

na podstawie książki: *Nowoczesne pamięci
półprzewodnikowe*, Betty Prince, WNT

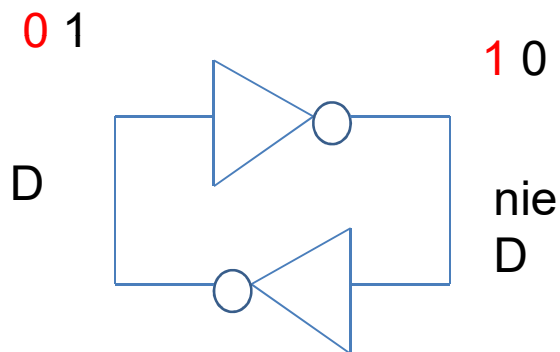
Ptc 2015/2016

11.12.2015

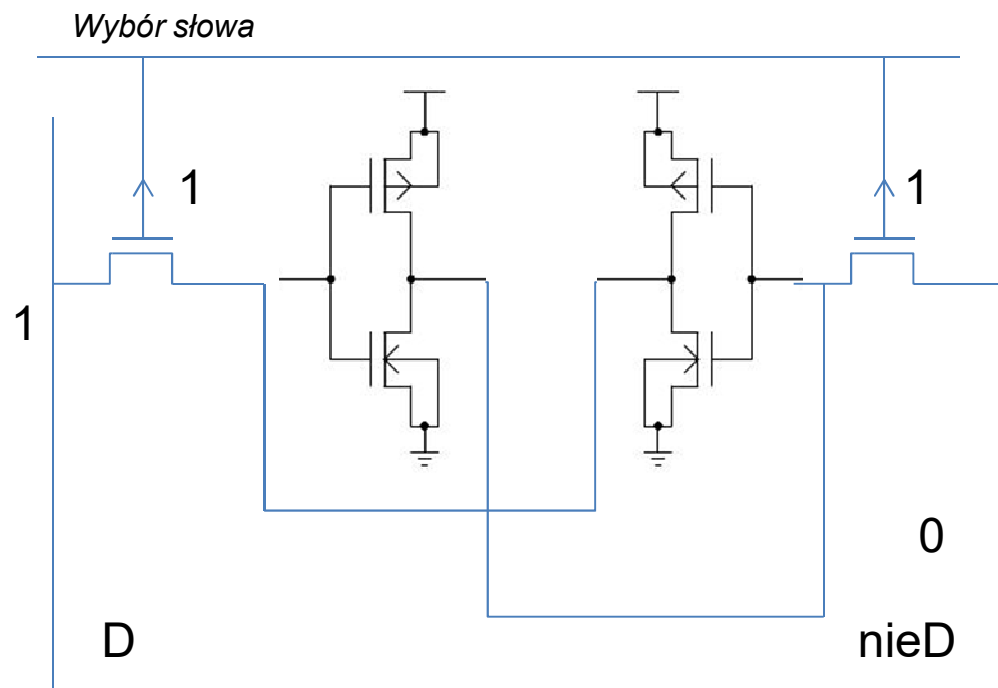
Półprzewodnikowe pamięci statyczne

- **Pamięci statyczne - SRAM**

- przechowywanie informacji w układzie zbudowanym z tranzystorów, dwa inwertery zwarte naprzemiennie wej-wyj – układ bistabilny.
- cykl działania (odczyt, zapis) krótszy niż dla pamięci dynamicznych - DRAM,
- po dostępie gotowość natychmiastowa do realizacji kolejnych cykli dostępu,
- wykorzystywane (ze względu na koszt) głównie jako szybkie pamięci podręczne procesora.

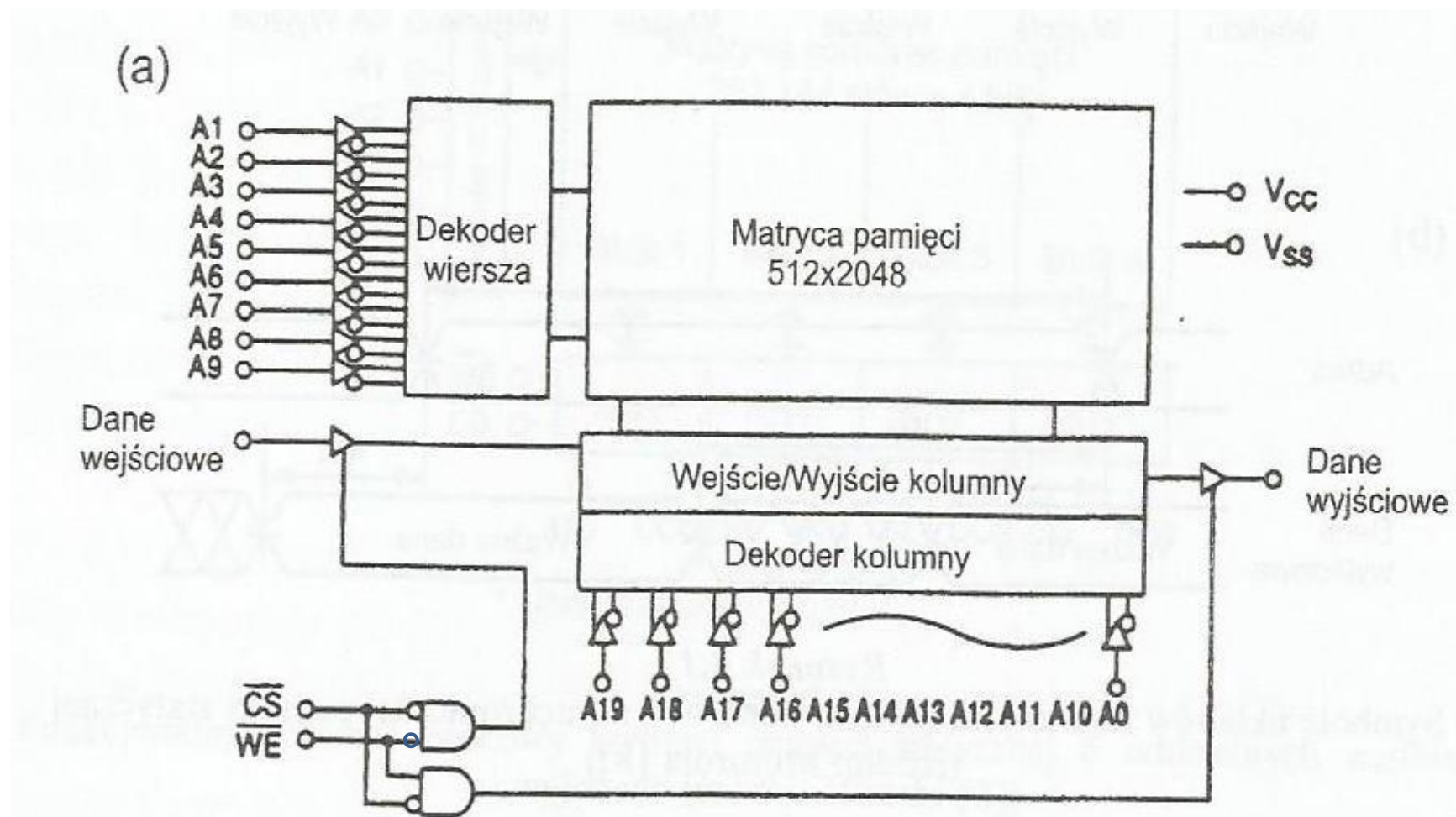


Pamięci statyczne budowane jako tablice komórek bitowych – 2 inwertetów i 2 tranzystorów dostępowych – rysunek obok.

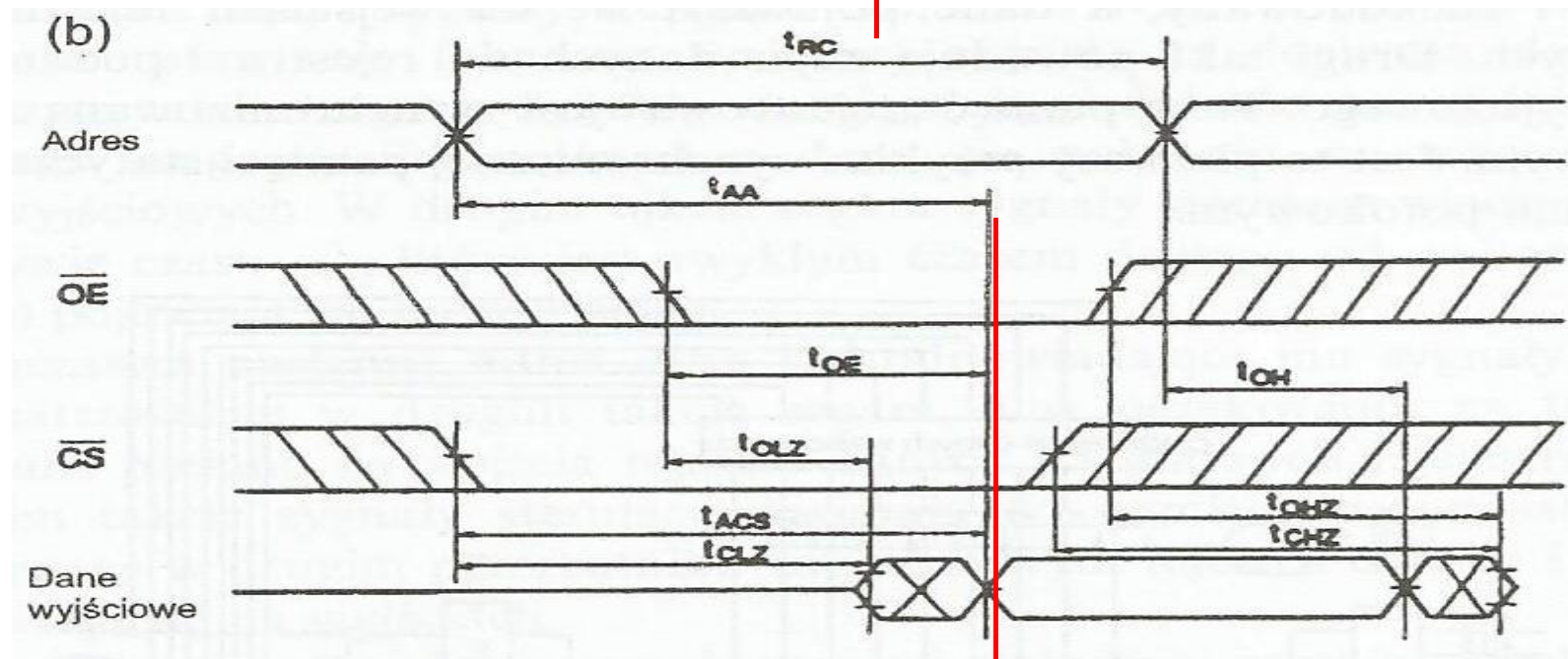
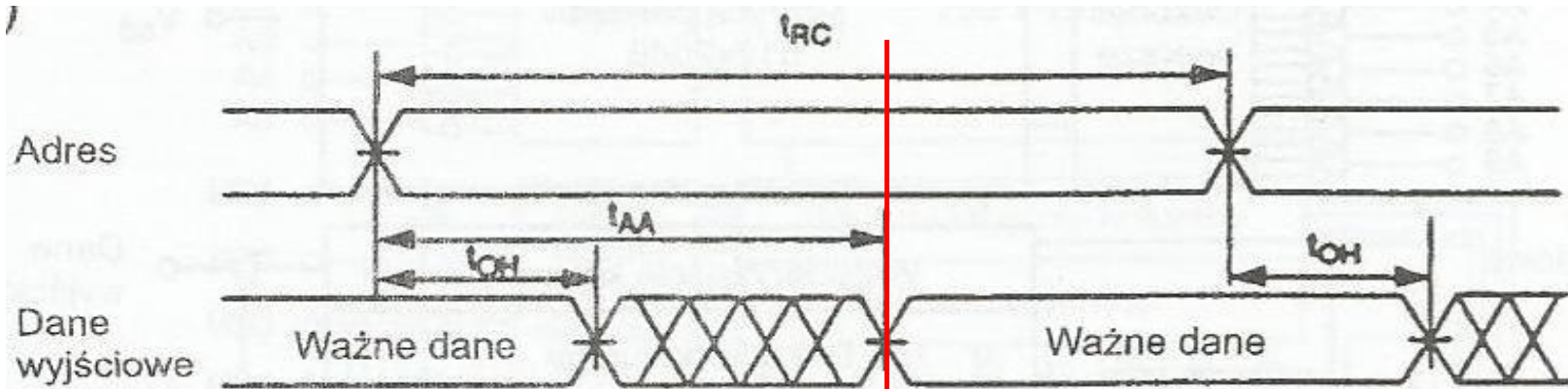


Schemat blokowy pamięci statycznej

pamięć **1 bit** x 1M – **20** linii adresowych

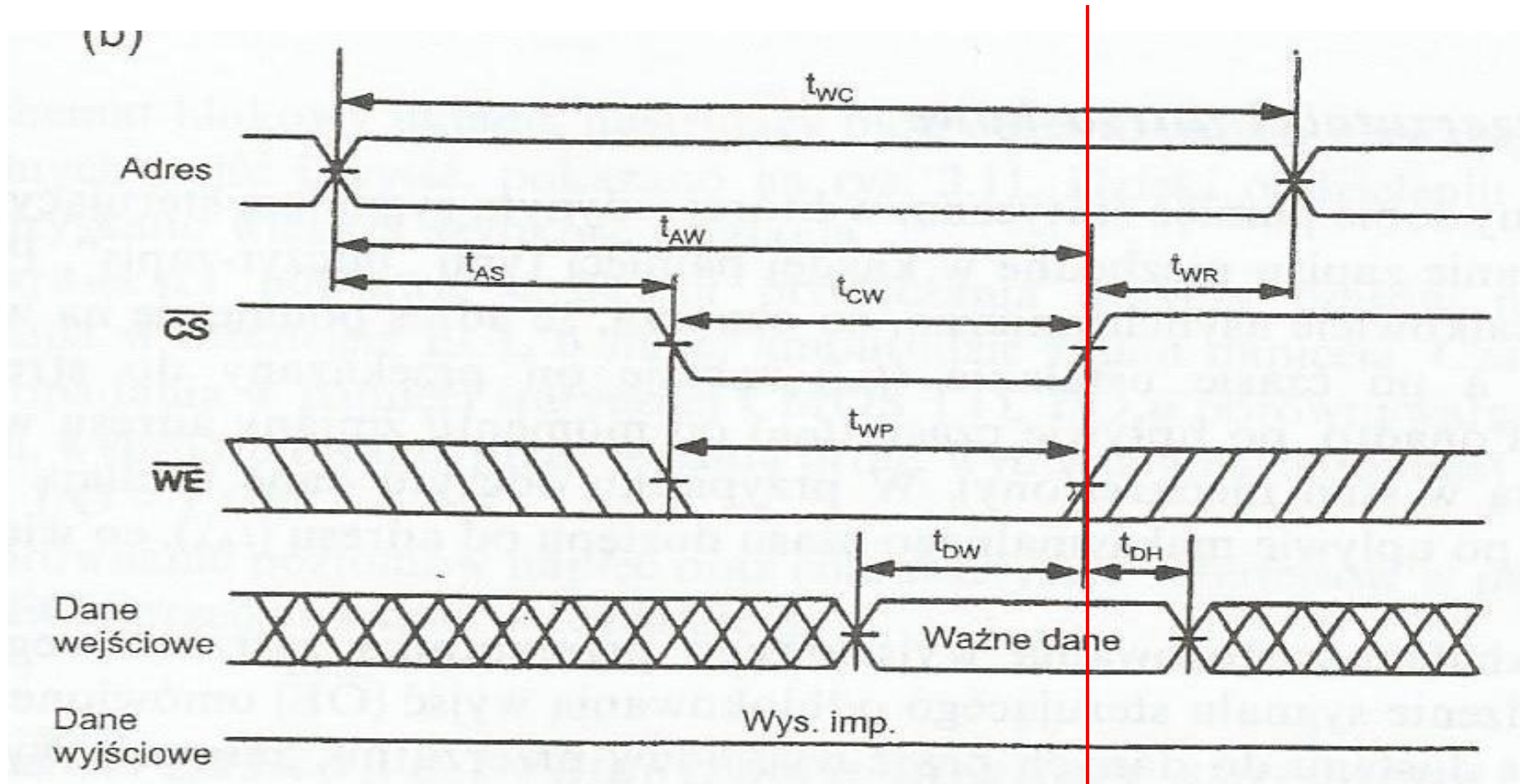


Cykle odczytu w pamięciach statycznych



Cykl zapisu w pamięciach statycznych

zapis - zbocze narastające sygnału - $cs' + we'$



Co przy braku spełnienia parametrów $t_{aw}, t_{cw}, t_{dw}, t_{wp}, t_{dh}$?

Parametry czasowe dostępu do pamięci statycznych

Par. zapisu	Znaczenie
tWC	Czas trwania cyklu zapisu
tAW	Wyprzedzenie adresu przed zapisem
tAS	Wyprzedzenie adresu przed CS'
tCW	Szerokość impulsu CS'
tWP	Szerokość impulsu WP'
tDW	Wyprzedzenie danych przed zapisem
tDH	Podtrzymanie danych po zapisie

Par. odczytu	Znaczenie
tRC	Czas trwania cyklu odczytu
tOH	Podtrzymanie starych danych na wyjściu
tAA	Czas dostępu do danych (od zmiany adresu) - dane ważne
tACS	Czas dostępu do danych (od CS') – dane ważne
tOE	Czas dostępu do danych (od OE') – dane ważne
tCLZ	Dane na wyjściu pamięci (od CS')
tOLZ	Dane na wyjściu pamięci (od OE')
tOHZ, tCHZ	Stan wysokiej impedancji na wyjściu (od OE,CS)

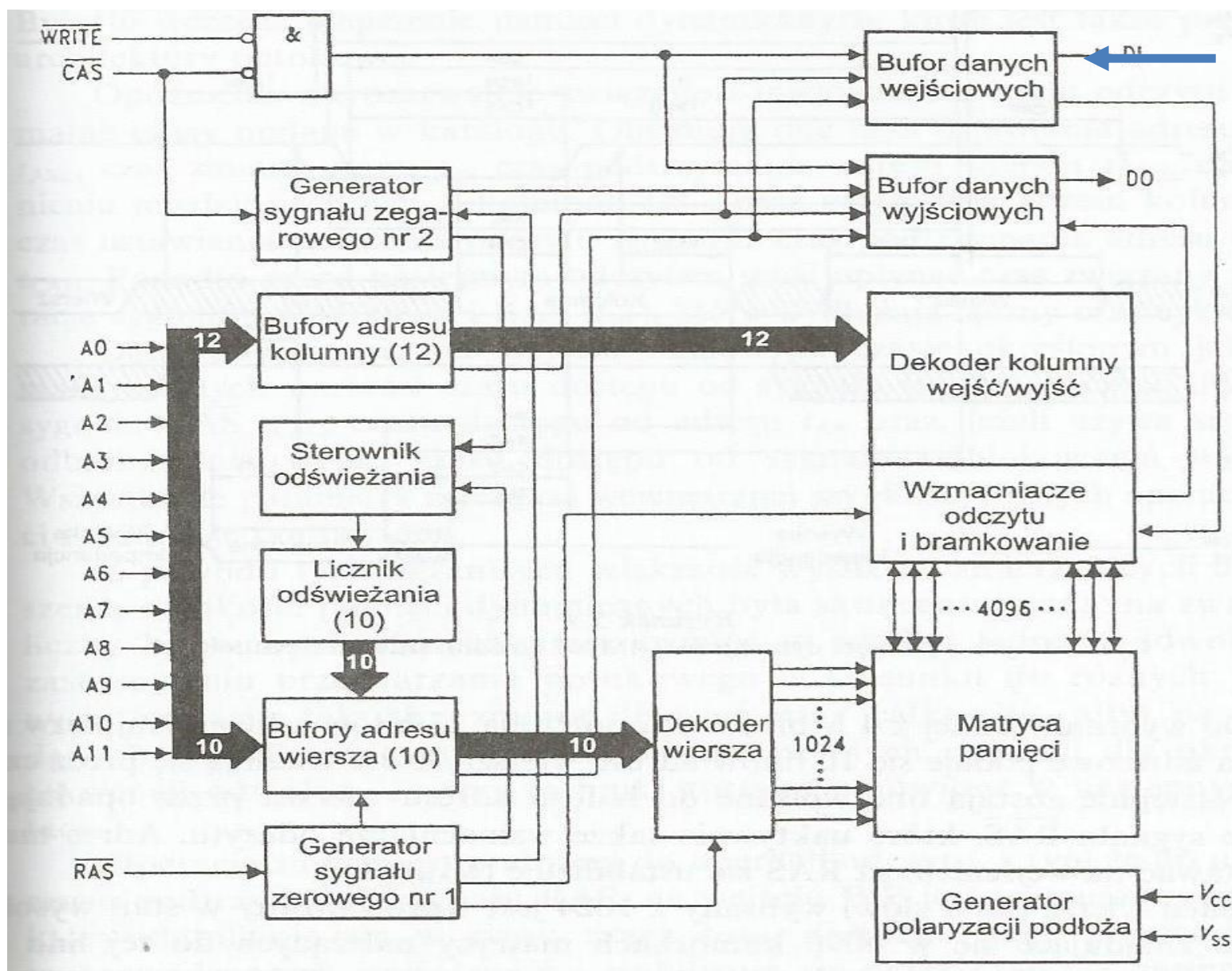
Półprzewodnikowe pamięci dynamiczne

- Pamięci dynamiczne DRAM
 - przechowywanie informacji w kondensatorze,
 - odczyt niszczy zapisaną informację i konieczne jest jej ponowne zapisanie ,
 - przed odczytem należy zapisaną informację zregenerować – uzupełnić ładunek zanikający ze względu na upływność kondensatora,
 - cykliczna regeneracja ładunku - odświeżanie zawartości pamięci,
 - dłuższe dla DRAM (niż w przypadku SRAM) cykle działania pamięci,
 - powierzchnia krzemu dla DRAM równa ok. $\frac{1}{4}$ powierzchni SRAM o tej samej pojemności - ok. 4 x tańsza,
 - wykorzystywane jako duże pamięci operacyjne systemu komputerowego.

Multipleksacja linii adresowych w pamięciach dynamicznych

- Liczba wejść adresowych układu pamięci ograniczona do połowy LA - liczby linii określonej liczbą słów pamięci – $2^{LA} = \text{liczba słów}$
- Na tych samych liniach wejściowych układu w kolejnych chwilach wyznaczonych sygnałami sterującym CAS i RAS (wybór wiersza i kolumny) podawane są fragmenty adresu – **adres wiersza** i **adres kolumny**.
- Duże pamięci – wiele linii adresowych – multipleksacja zmniejsza liczbę wyprowadzeń układu pamięci (zajmują one mniej miejsca na powierzchni płytki do montażu układów scalonych – **obwodu drukowanego**).

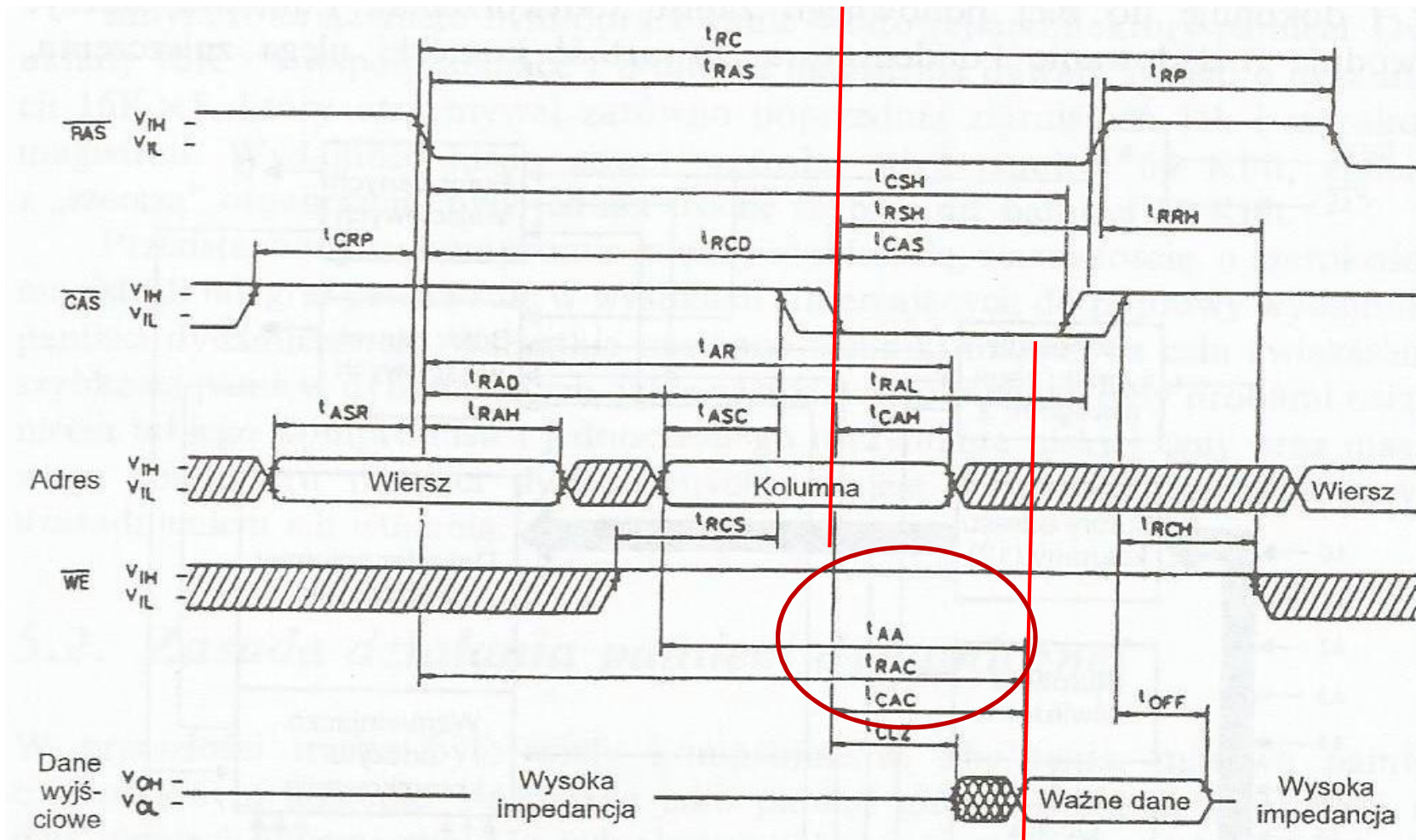
Schemat blokowy pamięci dynamicznej 4M bity



Cykle **odczytu** w pamięciach dynamicznych

- Cechy charakterystyczne:
 - **Wyprzedzenie i podtrzymanie adresu** wiersza i adresu kolumny względem zbocza opadającego odpowiednio sygnałów: wyboru wiersza RAS i wyboru kolumny CAS.
 - **Dane dostępne** po czasie dostępu mierzonym od:
 - RAS,
 - CAS i
 - adresu kolumny.
 - Poziom niski **CAS otwiera bufory wyjściowe** przy wejściu WE (write enable) nieaktywnym.

Cykle odczytu w pamięciach dynamicznych



Cykle **zapisu** w pamięciach dynamicznych

Cechy charakterystyczne:

- Wymagany WE' wyprzedzający **zbocze opadające wejścia CAS** którym odbywa się zapis.
- Dane wyprzedzające t_{DS} i podtrzymane t_{DH} względem zbocza opadającego wejścia CAS.
- Adresy wiersza i kolumny analogicznie jak w cyklu odczytu.

[illegible]

Parametry wielkości struktury pamięci

- **Określenie pojemności struktury pamięci:**
 - liczba bajtów
 - liczba słów
 - słowo to zbiór informacji w pamięci dostępny pod adresem zakodowanym na liniach adresowych układu
 - liczba linii adresowych wynikająca z kodowania binarnego adresów słów

Przykłady:

- 8 bajtów pamięci i słowo 1 bitowe to - ile słów i ile linii adresowych ?
 - $8 \times 8 = 64$ słowa i a 64 słowa to 6 linii adresowych, bo $64 = 2^6$
- 64 bajty pamięci i słowo 4 bitowe to
 - 64×2 (słowa w bajcie) = 128 słów , a 128 słów to 7 linii adresowych, bo $128 = 2^7$
- 1k bajt pamięci i słowo 1 bitowe to
 - $1 \text{ k} \times 8$ (słów w bajcie) = 8k słów pamięci, a 8k słów to 13 linii adresowych, bo $8 \text{ k} = 2^3 \times 2^{10}$
- 512 k bity i słowo 1 bajtowe to
 - $512 \text{ k} \times 1/8$ (słów w bicie) = 64 k słów pamięci, a 64k słów to 16 linii adresowych, bo $64 \text{ k} = 2^6 \times 2^{10}$
- 20 linii adresowych i słowo 4 bitowe to
 - $2^{20} = 1 \text{ M}$ słów – czyli $\frac{1}{2}$ M bajta

Łączenie pamięci - przykład 1

Dane wejściowe:

- Elementy składowe struktury pamięci – układy scalone pamięci: 1k bitów, słowo 4 bitowe – czyli 256 słów
- Rozmiar docelowej pamięci – 8 k bajtów, słowo 1 bajtowe – czyli 8k słów

Struktura wynikowa:

- Jeśli nowe słowo ma rozmiar będący wielokrotnością słowa układu wykorzystamy jego wszystkie bity - 2 x po 4 bity słowa daje słowo 8 bitowe
- Z wielkość pamięci i wykorzystania wszystkich bitów słów wynika że: rozmiar pamięci wzrasta zatem 64 x – potrzebne są 64 elementy o wielkości 128 bajtów.
- Adresowanie wynika z liczby słów w układzie:
 - Wyjściowym: 256 słów = 2^8 to 8 linii adresowych
 - Docelowym: 8 k słów = 2^{13} to 13 linii adresowych
 - **Do czego służy 5 dodatkowych linii adresowych?**
 - Do wybrania jednego z $2^5=32$ kolejnych modułów rozszerzających przestrzeń adresową.

Łączenie pamięci przykład 2

Zadanie:

- Zbudować w oparciu o podane elementy pamięci układ zadany parametrami:
 - Moduły składowe struktury pamięci – układy scalone pamięci: 1k bitów, słowo 4 bitowe
 - Układ - rozmiar docelowej pamięci – 1 k słów 8 bitowych

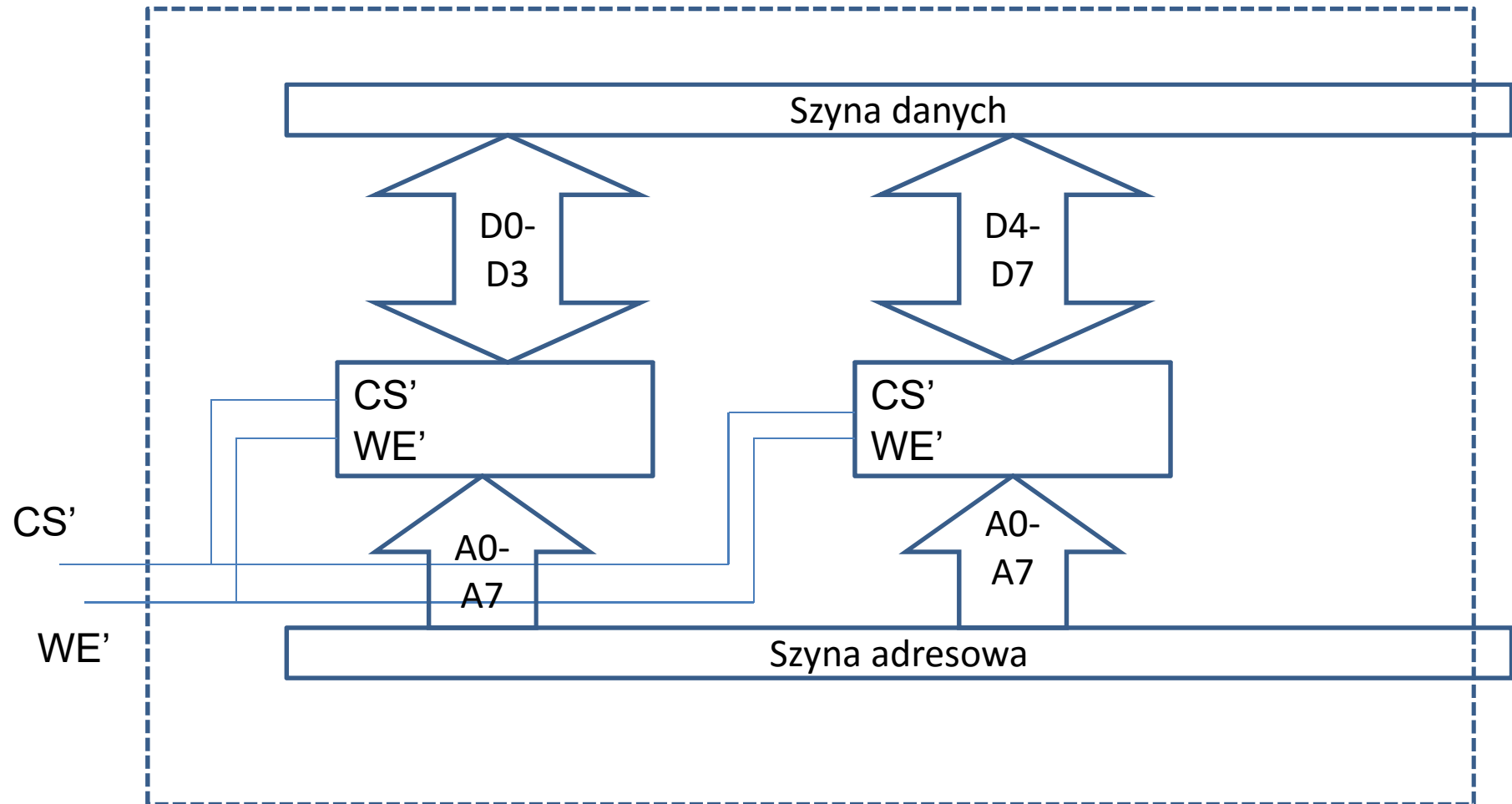
Rozwiązanie:

- Globalnie - rozmiar: Moduły mają pojemność 128 bajtów a pamięć docelowa – 1KBajt - potrzebnych jest zatem:
 - 8 układów.
- Przestrzeń adresowa: Moduły posiadają 256 słów. Układ ma 1 k słów.
- Aby zwiększyć liczbę słów potrzebne są :
 - 4 moduły.
- Zwiększenie długości słowa: Gdy mamy wystarczającą liczbę słów (4 moduły), aby długość słowa wzrosła 2 krotnie (z 4 na 8) potrzeba:
 - 2 x więcej modułów (czyli razem 8).

Połączenie struktury – rysunki na kolejnych stronach:

- Moduły mają 256 słów czyli 8 linii adresowych.
- Układ ma 1 k słów czyli 10 linii adresowych.

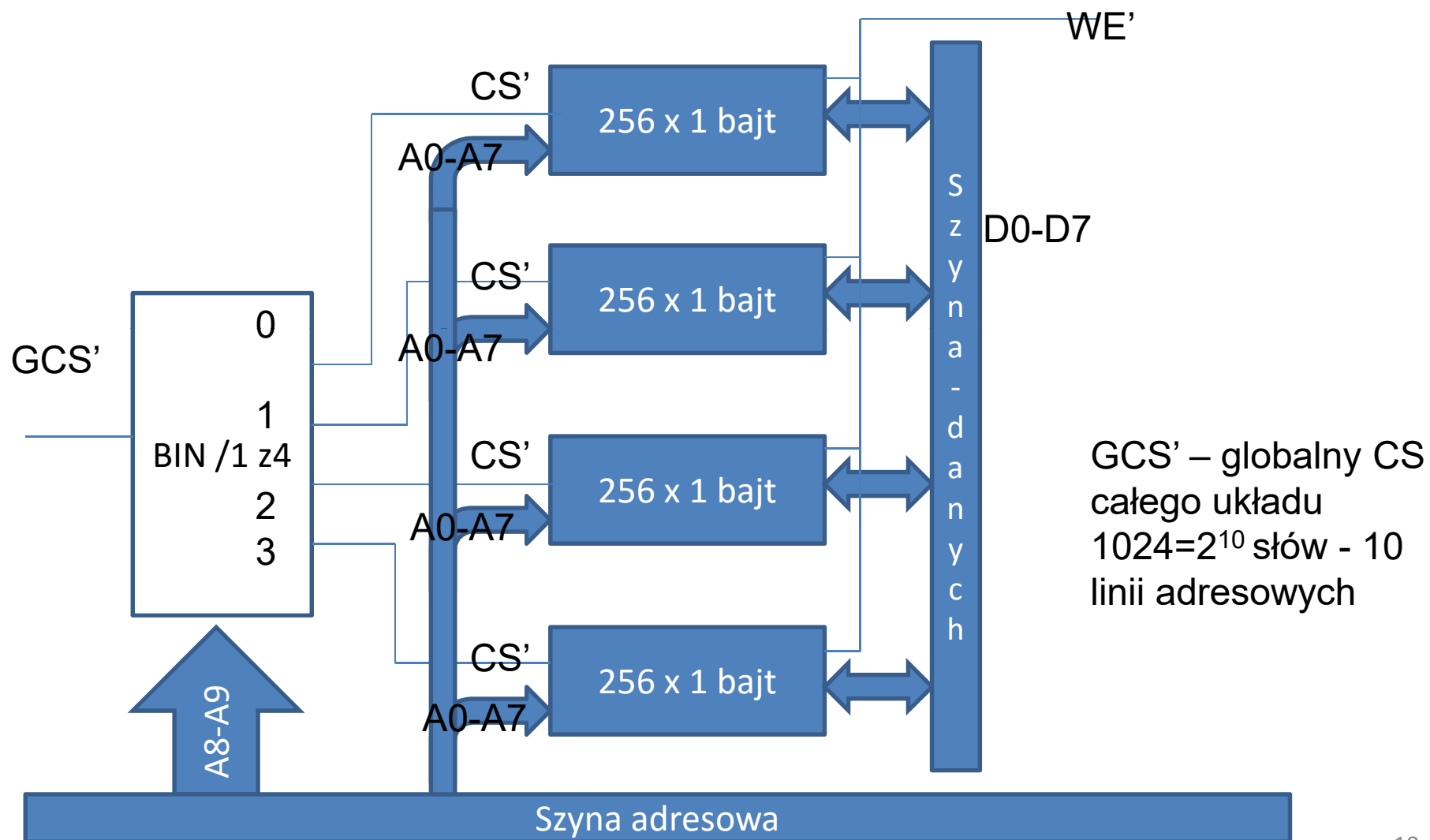
Łączenie pamięci przykład 2 wydłużenie słowa danych



Moduł 2x 256 po 4 bity daje 256 X 1 bajt
 $256 = 2^8$ słów adres 8 bitowy

Łączenie pamięci przykład 2

poszerzenie przestrzeni adresowej (1k X 1bajt)



Statyczne pamięci podręczne - przykład pamięci CAM

Pamięć podręczna (PP) służy do przechowywania danych bliżej procesora – dane aktualnie potrzebne, zawiera dane z PO (pamięci operacyjnej) powielone lub dane zmodyfikowane przez procesor (nie zapisane jeszcze w PO). Mniejszy rozmiar PP niż PO, inna organizacja i bliskość do jednostki wykonawczej (np. często w ramach układu scalonego procesora).

PP pamięta zarówno dane jak i adres tych danych (w PO). Informacje te są zapisane w sposób zależny od organizacji PP.

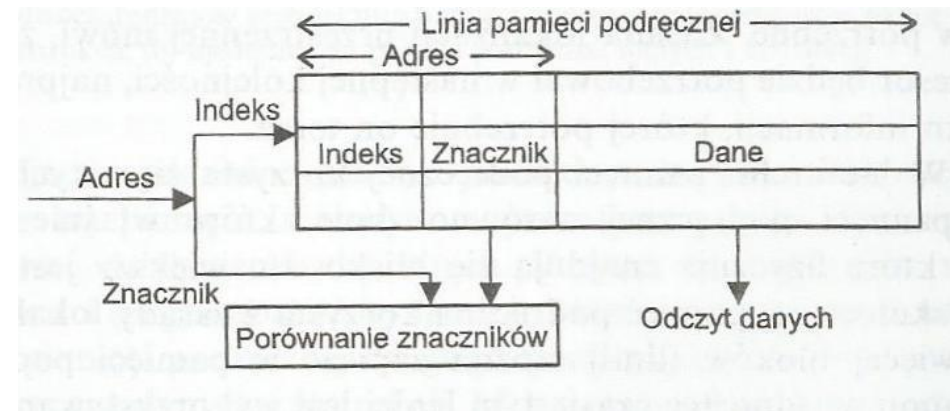
PP na podstawie adresu podanego przez procesor wystawia żądane dane (jeśli aktualnie są w pp – mamy trafienie – cache hit) lub informuje o braku danych (brak trafienia do PP – cache miss).

Pamięć działająca w ten sposób (w odróżnieniu od RAM) nazywa się **pamięcią adresowaną zawartością - CAM** (ang. **Content addressable memory**).

Przy odczycie **CAM** **dostarcza** informacji powiązanej z wartością podaną na wejściu pamięci:

- informacje czy i pod jakim adresem posiada szukane dane lub
- jak w przypadku PP - dane zapisane w PP odpowiadające podanej na wejściu PP informacji będącej adresem szukanych danych w PO.

Schemat koncepcyjny działania pamięci podręcznej

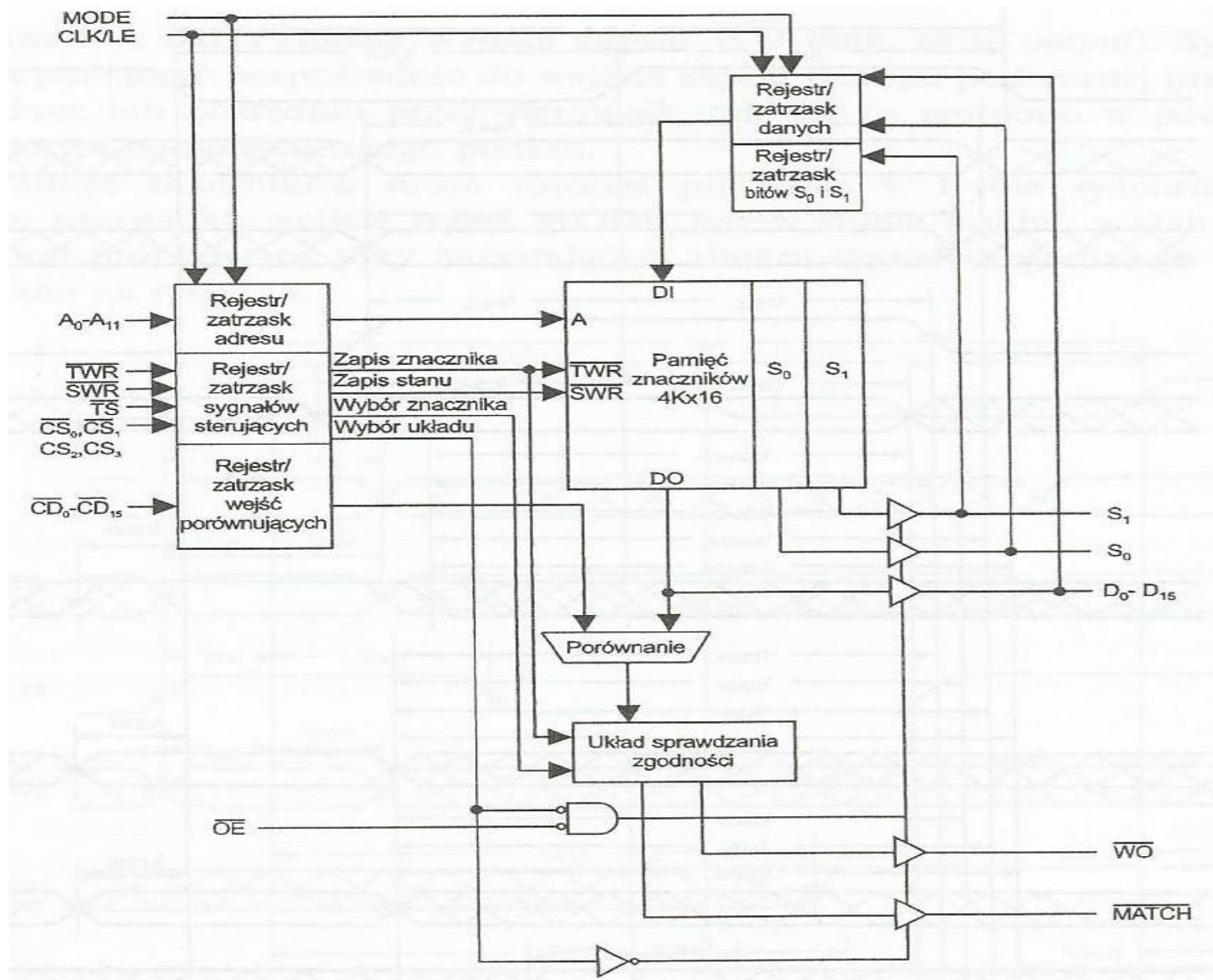


linia pamięci podręcznej (LP): ciąg słów z kolejnych adresów PO będący kwantem ilości informacji przechowywanej w pp, w tym przykładzie rozszerzona o adres początku tego ciągu bajtów w PO.

Adres danych przechowywanych w pp dzielimy na indeks (młodsze bity) i znacznik.

W trakcie operacji odczytu z pp sterownik pp dzieli adres żądanych danych na indeks i znacznik. Znacznik adresu jest porównywany ze znacznikiem zapisanym w lokacji odpowiadającej indeksowi. Jeśli znaczniki są równe to wystąpiło trafienie - dane są w pp i wtedy dane są wysyłane do procesora.

Schemat blokowy pamięci znaczników



Statyczne pamięci podręczne (2)

Pamięć podręczna zarządzana bezpośrednio

- Możliwe miejsce przechowywania danych z określonego adresu jest ściśle określone poprzez indeks.
- W takiej pamięci tylko **jeden blok** informacyjny (wskazany indeksem i zawierający linię danych i adres) w pamięci podręcznej **dostępny jest dla zapisu danych spod określonego adresu** p.o. (jeśli dana jest to wiadomo gdzie)
- Przy odczycie **jeden** komparator w układzie wystarczy do sprawdzenia zgodności podanego adresu dostępu (tj. znacznika t.j. starszej części adresu) z adresem w PP (znacznikiem), spod którego pochodzi wartość przechowywana w tej ściśle określonej (tj. określonej indeksem) pozycji pp.
- Zmniejszenie rozmiaru pp dla określonego adresu lecz również zmniejszenie kosztu (mniej komparatorów znaczników) dla tej samej prędkości dostępu.

Statyczne pamięci podręczne (3)

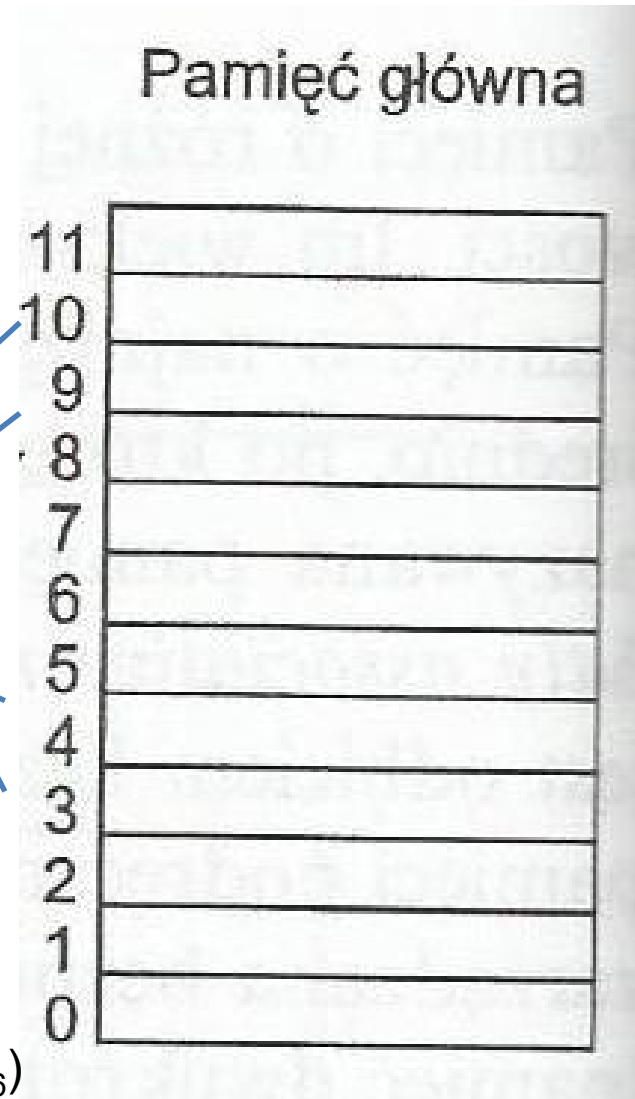
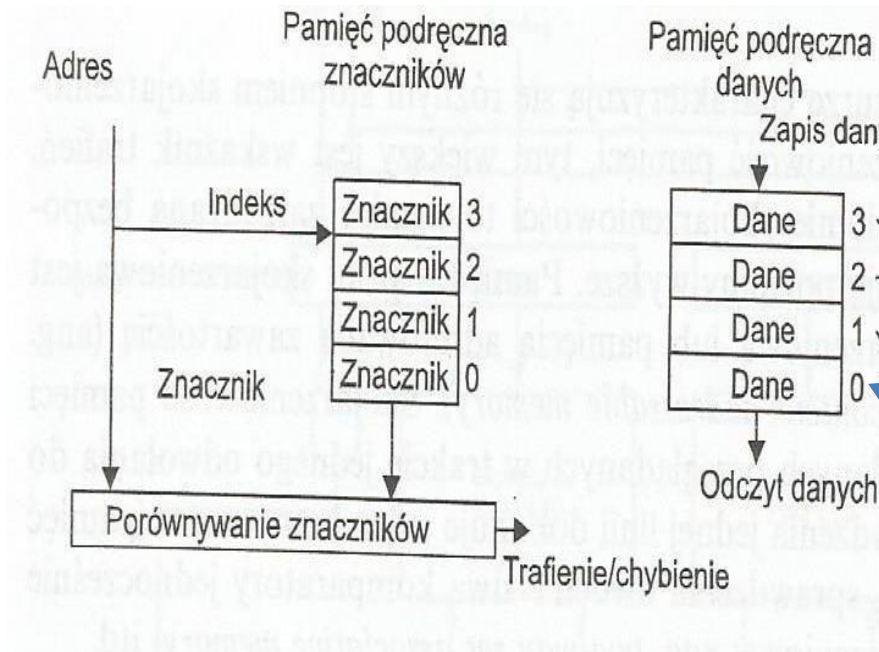
Pamięć podręczna zarządzana bezpośrednio

Przykład: jeśli PP **zarządzana bezpośrednio** zawiera 128 linii PP po 64 bajty (rozmiar linii PP) to jedno możliwe położenie danych w PP **jest zależne** od 13 młodszych linii adresowych:

- 6 bitów określa położenie słowa w linii PP o wielkości $64=2^6$ bajtów a
- 7 kolejnych bitów określa położenie tej linii w jednej z $128=2^7$ lokacji określonej tymi 7 bitami.

Pamięć podręczna zarządzana bezpośrednio

linia pp tylko w jednym miejscu

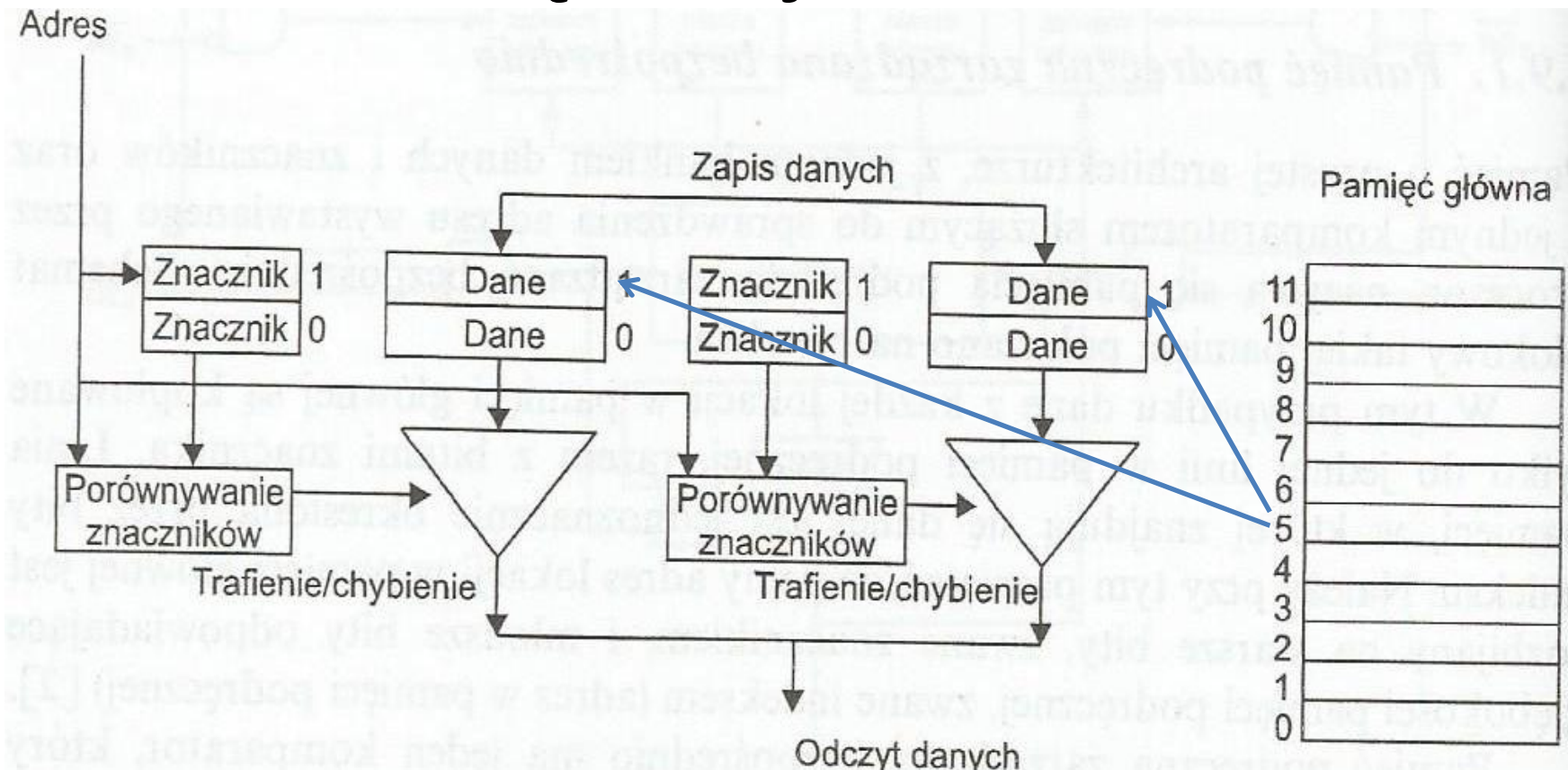


Na kolejnych pozycjach pp (indeksy 0,1,2,3) mogą być przechowywane wyłącznie dane pochodzące z obszarów adresowanych ($XXXA_6$) (0,4,8,...) (1,5,9,...) (2,6,10,...) ... (wartości tych adresów nie uwzględniają długości linii PP – A_0-A_5).

Statyczne pamięci podręczne (4)

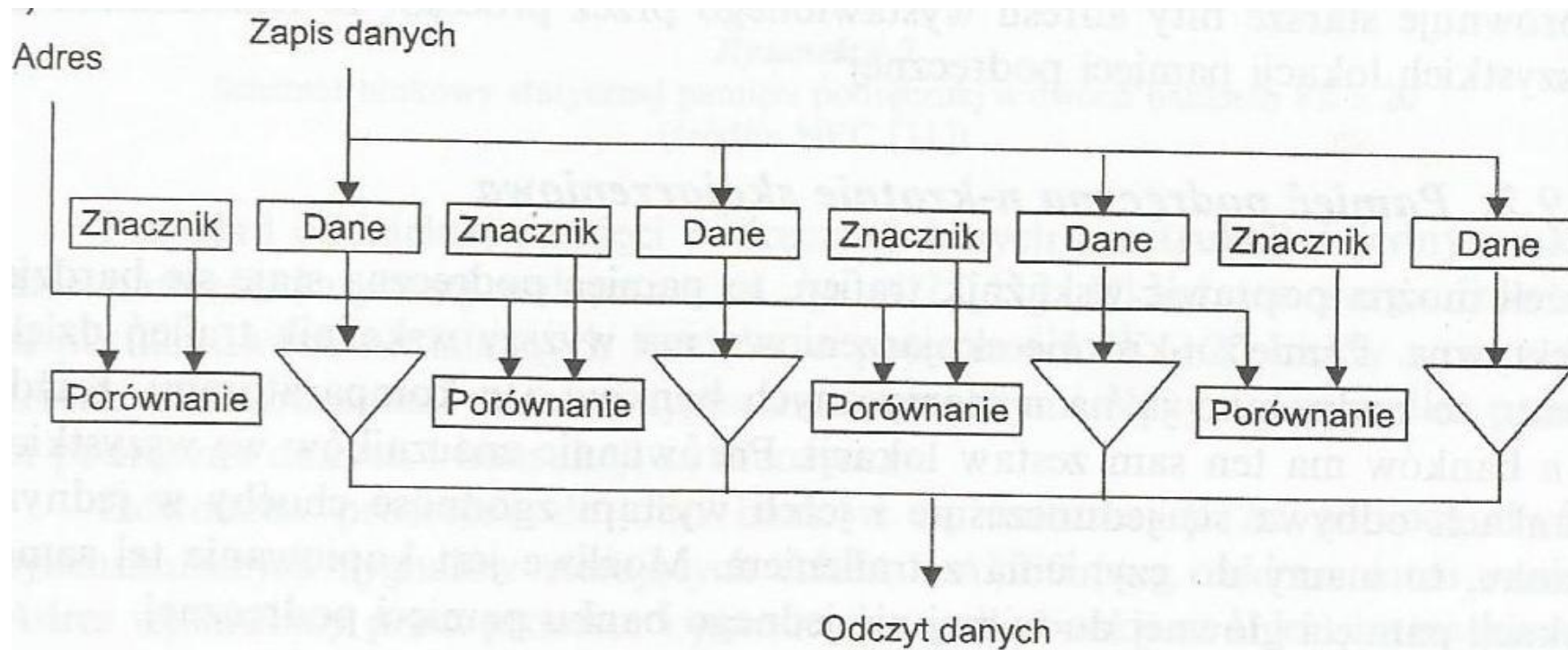
- **Pamięć podręczna skojarzeniowa** daje możliwość przechowywania wartości z określonego adresu na wielu pozycjach (w różnych bankach pp) i wymaga porównania znaczników w ramach każdego z banków (wielokrotnie skojarzeniowa pamięć podręczna).
- W **pełni skojarzeniowe pamięci podręczne** pozwalają na przechowywanie informacji w dowolnych komórkach pamięci.

Pamięć skojarzeniowa



Możliwe pozycje przechowywania w PP danych z „adresu 5” pamięci operacyjnej. Dane z określonego adresu mogą być przechowywane w jednym z 2 banków tj . na 2 z 4 pozycji pp.

Pamięć w pełni skojarzeniowa



Każdy blok pamięci umożliwia przechowywanie informacji spod dowolnego adresu. Brak indeksu dla określenia pozycji danych w ramach banku.